

# AWS Interview Preparation

---

Topic Name : CLOUD

Sub-topic Name : AWS

Course link : <https://ineuron.ai/course/AWS-Interview-Preparation>

## Course Description :-

Amazon Web Service (AWS) is one of the fastest-growing fields in the technology world. This course is designed to help you achieve your goals which cover interview questions. We cover a wide range of topics in this course. We have questions on Amazon Web Services (AWS) best practices, Security, Simple Storage Service S3, Elastic Compute Cloud EC2, Elastic Load Balancing ELB, CloudFront, DynamoDB, CloudWatch, ElastiCache and Lambda.

## Course Features :-

- => Self-Paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Aws interview questions
- => Learn about IAM
- => Learn about EC2

## Requirements :-

- => Understanding of AWS
- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

=> The Cloud :

- ~ Which of the following does not contribute significantly to the operational value of a large cloud provider like AWS?
- ~ Which of the following are signs of a highly available application? (Select TWO.)
- ~ How does the metered payment model make many benefits of cloud computing possible? (Select TWO.)
- ~ Which of the following are direct benefits of server virtualization? (Select TWO.)
- ~ What is a hypervisor? Preview
- ~ Which of the following best describes server virtualization?
- ~ Which of the following best describes Infrastructure as a Service products?
- ~ Which of the following best describes Platform as a Service products?
- ~ Which of the following best describes Software as a Service products?
- ~ Which of the following best describes scalability?
- ~ Which of the following best describes elasticity?
- ~ Which of the following characteristics most help AWS provide such scalable services? (Select TWO.)

=> Understanding Your AWS Account :

- ~ Which of the following EC2 services can be used without charge under the Free Tier?
- ~ You want to experiment with deploying a web server on an EC2 instance. Which two of the following resources can you include to make that work while remaining within the Free Tier? (Select TWO.)
- ~ Which of the following usage will always be cost-free even after your accounts Free Tier has expired? (Select TWO.)
- ~ Which of the following tools are available to ensure you won't accidentally run past your Free Tier limit and incur unwanted costs? (Select TWO.)
- ~ Which of the following is likely to be an accurate source of AWS pricing information?
- ~ Which of the following will probably not affect the pricing for an AWS service?
- ~ Which of the following is a limitation of the AWS Simple Monthly Calculator?
- ~ Which of the following Simple Monthly Calculator selections will likely have an impact on most other configuration choices on the page? (Select TWO.)
- ~ Which of the following is not an included parameter in the AWS Total Cost of Ownership Calculator?
- ~ Which of the following AWS Total Cost of Ownership Calculator parameters is likely to have the greatest impact on cost?
- ~ Which of the following AWS documentation URLs points to the page containing an up-to-date list of service limits?
- ~ Which of the following best describes one possible reason for AWS service limits?
- ~ Is it always possible to request service limit increases from AWS? Preview
- ~ Which is the best place to get a quick summary of this month's spend for your account?
- ~ What is the main goal for creating a Usage budget type (in AWS Budgets)?
- ~ Which of the following is not a setting you can configure in a Cost budget?
- ~ What is the main difference between the goals of Cost Explorer and of cost and usage reports?
- ~ What is the purpose of cost allocation tags?
- ~ Which of the following scenarios would be a good use case for AWS Organizations? (Select TWO.)
- ~ Which of these tools lets you design graphs within the browser interface to track your account spending?

## => Getting Support on AWS :

- ~ Your company is planning a major deployment on AWS. While the design and testing stages are still in progress, which of the following plans will provide the best blend of support and cost savings?
- ~ Your web development team is actively gearing up for deployment of an e-commerce site. During these early stages of the process, individual developers are running into frustrating conflicts and configuration problems that are highly specific to your situation. Which of the following plans will provide the best blend of support and cost savings?
- ~ Your corporate website was offline last week for more than two hours which caused serious consequences, including the early retirement of your CTO. Your engineers have been having a lot of trouble tracking down the source of the outage and admit that they need outside help. Which of the following will most likely meet that need?
- ~ For which of the following will AWS provide direct 24/7 support to all users even those on the Basic Support plan?
- ~ The primary purpose of an AWS technical account manager is to:
- ~ Your Linux-based EC2 instance requires a patch to a Linux kernel module. The problem is that patching the module will, for some reason, break the connection between your instance and data in an S3 bucket. Your team doesn't know if it's possible to work around this problem. Which is the most cost-effective AWS plan through which support professionals will try to help you?
- ~ Your company enrolled in the Developer Support plan and, through the course of one month, consumed \$4,000 USD of AWS services. How much will the support plan cost the company for the month?
- ~ Your company enrolled in the Business Support plan and, through the course of three months, consumed \$33,000 of AWS services (the consumption was equally divided across the months). How much will the support plan cost the company for the full three months?
- ~ Which of the following AWS support services does not offer free documentation of some sort?
- ~ What is the key difference between the roles of AWS Professional Services and a technical account manager (TAM)?
- ~ AWS documentation is available in a number of formats, including which of the following? (Select TWO.)
- ~ Which of the following documentation sites are most likely to contain code snippets for you to cut and (after making sure you understand exactly what they'll do) paste into your AWS operations? (Select TWO.)
- ~ What is the primary function of the content linked from the Knowledge Center?
- ~ On which of the following sites are you most likely to find information about encrypting your AWS resources?
- ~ When using AWS documentation pages, what is the best way to be sure the information you're reading is up-to-date?
- ~ Which of the following is not a Trusted Advisor category?
- ~ Data volumes that aren't properly backed up is an example of which of these Trusted Advisor categories?
- ~ Instances that are running (mostly) idle should be identified by which of these Trusted Advisor categories?
- ~ Within the context of Trusted Advisor, what is a false positive?
- ~ Which of the following Trusted Advisor alerts is available only for accounts on the Business or Enterprise Support plan? (Select TWO.)

## => Understanding the AWS Environment :

- ~ Which of the following designations would refer to the AWS US West (Oregon) region?
- ~ Which of the following is an AWS Region for which customer access is restricted?
- ~ When you request a new virtual machine instance in EC2, your instance will automatically launch into the currently selected value of which of the following?
- ~ Which of the following are not globally based AWS services? (Select TWO.)
- ~ Which of the following would be a valid endpoint your developers could use to access a particular Relational Database Service instance you're running in the Northern Virginia region?
- ~ What are the most significant architectural benefits of the way AWS designed its regions? (Select TWO.)
- ~ Why is it that most AWS resources are tied to a single region?
- ~ You want to improve the resilience of your EC2 web server. Which of the following is the most effective and efficient approach?
- ~ Which of the following is the most accurate description of an AWS Availability Zone?
- ~ Which of the following most accurately describes a subnet within the AWS ecosystem?
- ~ What determines the order by which subnets/AZ options are displayed in EC2 configuration dialogs?
- ~ What is the primary goal of autoscaling? Preview
- ~ Which of the following design strategies is most effective for maintaining the reliability of a cloud application?
- ~ Which of the following AWS services are not likely to benefit from Amazon edge locations? (Select TWO.)
- ~ Which of the following is the primary benefit of using CloudFront distributions?
- ~ What is the main purpose of Amazon Route 53?
- ~ According to the AWS Shared Responsibility Model, which of the following are responsibilities of AWS? (Select TWO.)
- ~ According to the AWS Shared Responsibility Model, what's the best way to define the status of the software driving an AWS managed service?
- ~ Which of the following is one of the first places you should look when troubleshooting a failing application?
- ~ Where will you find information on the limits AWS imposes on the ways you can use your account resources?

## => Securing Your AWS Resources :

- ~ What is the primary function of the AWS IAM service?
- ~ Which of the following are requirements you can include in an IAM password policy? (Select THREE.)
- ~ Which of the following should you do to secure your AWS root user? (Select TWO.)
- ~ How does multi-factor authentication work?
- ~ Which of the following SSH commands will successfully connect to an EC2 Amazon Linux instance with an IP address of 54.7.35.103 using a key named mykey.pem?
- ~ What's the most efficient method for managing permissions for multiple IAM users?
- ~ What is an IAM role?
- ~ How can federated identities be incorporated into AWS workflows? (Select TWO.)
- ~ Which of the following are valid third-party federated identity standards? (Select TWO.)
- ~ What information does the IAM credential report provide?
- ~ What text format does the credential report use?
- ~ Which of the following IAM policies is the best choice for the admin user you create in order to replace the root user for day-to-day administration tasks?
- ~ What will you need to provide for a new IAM user you're creating who will use programmatic access to AWS resources?
- ~ What will IAM users with AWS Management Console access need to successfully log in?
- ~ Which of the following will encrypt your data while in transit between your office and Amazon S3?
- ~ Which of the following AWS resources cannot be encrypted using KMS?
- ~ What does KMS use to encrypt objects stored on your AWS account?
- ~ Which of the following standards governs AWS-based applications processing credit card transactions?
- ~ What is the purpose of the Service Organization Controls (SOC) reports found on AWS Artifact?
- ~ What role can the documents provided by AWS Artifact play in your application planning? (Select TWO.) Preview

## => Working with Your AWS Resources :

- ~ Which of the following credentials can you use to log into the AWS Management Console?
- ~ How long will your session with the AWS Management Console remain active?
- ~ While looking at the EC2 service console in the AWS Management Console while logged in as the root user, you notice all of your instances are missing. What could be the reason?
- ~ Which of the following is true regarding a resource tag?
- ~ Which of the following is required to use the AWS Command Line Interface (CLI)?
- ~ Which of the following are options for installing the AWS CLI on Windows 10? (Select TWO.)
- ~ After installing the AWS Command Line Interface, what should you do before using it to securely manage your AWS resources?

~ Which output format does the AWS CLI support?

# Full Stack Data Science

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science>

## Course Description :-

This is a full stack data science course where you will learn all the stack required to work in data science, data analytics and big data industry including ML ops and cloud infrastructure .

## Course Features :-

- => Full stack Data Science masters certification
- => 56 + hands-on industry real-time projects.
- => 500 hours of recorded classes.
- => Lifetime Dashboard access
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => ML ops
- => Cloud
- => Data structure and algorithm
- => Architecture
- => Domain wise project
- => Databases
- => Negotiations skills
- => Mock interview
- => Interview preparation
- => Resume building after every module

## Requirements :-

- => Dedication
- => Computer with i3 and above configuration

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

## => Course Introduction :

- ~ *course overview and dashboard description*
- ~ *Introduction of data science and its application in day to day life*
- ~ *Programming language overview*
- ~ *Installation (tools: sublime, vscode, pycharm, anaconda, atom, jupyter notebook, kite)*
- ~ *Virtual environment*
- ~ *Why python*

## => Python Basic :

- ~ *Introduction of python and comparison with other programming language*
- ~ *Installation of anaconda distribution and other python ide*
- ~ *Python objects, number & Booleans, strings*
- ~ *Container objects, mutability of objects*
- ~ *Operators - arithmetic, bitwise, comparison and assignment operators, operators precedence and associativity*
- ~ *Conditions (if else, if-elif-else), loops (while, for)*
- ~ *Break and continue statement and range function*

## => String Objects :

- ~ *basic data structure in python*
- ~ *String object basics*
- ~ *String inbuilt methods*
- ~ *Splitting and joining strings*
- ~ *String format functions*

## => List Object Basics :

- ~ *List methods*
- ~ *List as stack and queues*
- ~ *List comprehensions*

## => Tuples, Sets, Dictionaries & its Function :

- ~ *Dictionary object methods*
- ~ *Dictionary comprehensions*
- ~ *Dictionary view objects*
- ~ *Functions basics, parameter passing, iterators*
- ~ *Generator functions*
- ~ *Lambda functions*
- ~ *Map, reduce, filter functions*

## => Memory Management :

- ~ *Multithreading*
- ~ *Multiprocessing*

## => OOPs Concepts :

- ~ *oops basic concepts.*
- ~ *Creating classes*
- ~ *Pillars of oops*
- ~ *Inheritance*
- ~ *Polymorphism*
- ~ *Encapsulation*
- ~ *Abstraction*
- ~ *Decorator*
- ~ *Class methods and static methods*
- ~ *Special (magic/dunder) methods*
- ~ *Property decorators - getters, setters, and deletes*

## => Files :

- ~ *Working with files*
- ~ *Reading and writing files*
- ~ *Buffered read and write*
- ~ *Other file methods*
- ~ *Logging, debugger*
- ~ *Modules and import statements*

## => Exception Handling and Difference between Exception and Error :

- ~ *Exceptions handling with try-except*
- ~ *Custom exception handling*
- ~ *List of general use exception*
- ~ *Best practice exception handling*

## => GUI Framework :

- ~ *What is desktop and standalone application*
- ~ *Use of desktop app*
- ~ *Examples of desktop app*
- ~ *Tinker*
- ~ *Kivy*

## => Database :

- ~ *SQLite*
- ~ *MySQL*
- ~ *Mongo dB*
- ~ *NoSQL - Cassandra*

## => Web API :

- ~ *What is web API*
- ~ *Difference b/w API and web API*
- ~ *Rest and soap architecture*
- ~ *Restful services*

## => Flask :

- ~ *Flask introduction*
- ~ *Flask application*
- ~ *Open link flask*

- ~ App routing flask
- ~ Url building flask
- ~ Http methods flask
- ~ Templates flask
- ~ Flask project: food app
- ~ Postman
- ~ Swagger

=> Django :

- ~ Django introduction
- ~ Django project: weather app
- ~ Django project: memes generator
- ~ Django project: blog app
- ~ Django project in cloud

=> Stream Lit :

- ~ Stream lit introduction
- ~ Stream lit project structure
- ~ Stream lit project in cloud

=> Pandas Basic :

- ~ Python pandas - series
- ~ Python pandas data frame
- ~ Python pandas panel
- ~ Python pandas - basic functionality
- ~ Reading data from different file system

=> Pandas Advance :

- ~ Python pandas re indexing python
- ~ Pandas iteration
- ~ Python pandas sorting.
- ~ Working with text data options & customization
- ~ Indexing & selecting
- ~ Data statistical functions
- ~ Python pandas - window functions
- ~ Python pandas - date functionality
- ~ Python pandas time delta
- ~ Python pandas - categorical data
- ~ Python pandas visualization
- ~ Python pandas - iotools

=> Dask :

- ~ Dask Array
- ~ Dask Bag
- ~ Dask DataFrame
- ~ Dask Delayed
- ~ Dask Futures
- ~ Dask API
- ~ Dask SCHEDULING
- ~ Dask Understanding Performance
- ~ Dask Visualize task graphs
- ~ Dask Diagnostics (local)
- ~ Dask Diagnostics (distributed)
- ~ Dask Debugging
- ~ Dask Ordering

=> Python Numpy :

- ~ Numpy - ND array object.
- ~ Numpy - data types.
- ~ Numpy - array attributes.
- ~ Numpy - array creation routines.
- ~ Numpy - array from existing.
- ~ Data array from numerical ranges.
- ~ Numpy - indexing & slicing.
- ~ Numpy advanced indexing.
- ~ Numpy broadcasting.
- ~ Numpy - iterating over array.
- ~ Numpy - array manipulation.
- ~ Numpy - binary operators.
- ~ Numpy - string functions.
- ~ Numpy - mathematical functions.
- ~ Numpy - arithmetic operations.
- ~ Numpy - statistical functions.
- ~ Sort, search & counting functions.
- ~ Numpy - byte swapping.
- ~ Numpy - copies & views.
- ~ Numpy - matrix library.
- ~ Numpy - linear algebra

=> Visualization :

- ~ Matplotlib
- ~ Seaborn
- ~ Cufflinks
- ~ Plotly
- ~ Bokeh

=> Statistics Basic :

- ~ Introduction to basic statistics terms
- ~ Types of statistics
- ~ Types of data
- ~ Levels of measurement

- ~ Measures of central tendency
- ~ Measures of dispersion
- ~ Random variables
- ~ Set
- ~ Skewness
- ~ Covariance and correlation

#### => Probability Distribution Function :

- ~ Probability density/distribution function
- ~ Types of the probability distribution
- ~ Binomial distribution
- ~ Poisson distribution
- ~ Normal distribution (Gaussian distribution)
- ~ Probability density function and mass function
- ~ Cumulative density function
- ~ Examples of normal distribution
- ~ Bernoulli distribution
- ~ Uniform distribution
- ~ Z stats
- ~ Central limit theorem
- ~ Estimation

#### => Statistics Advance :

- ~ a Hypothesis
- ~ Hypothesis testings mechanism
- ~ P-value
- ~ T-stats
- ~ Student t distribution
- ~ T-stats vs. Z-stats: overview
- ~ When to use a t-tests vs. Z-tests
- ~ Type 1 & type 2 error
- ~ Bayes statistics (Bayes theorem)
- ~ Confidence interval(ci)
- ~ Confidence intervals and the margin of error
- ~ Interpreting confidence levels and confidence intervals
- ~ Chi-square test
- ~ Chi-square distribution using python
- ~ Chi-square for goodness of fit test
- ~ When to use which statistical distribution?
- ~ Analysis of variance (anova)
- ~ Assumptions to use anova
- ~ Anova three type
- ~ Partitioning of variance in the anova
- ~ Calculating using python
- ~ F-distribution
- ~ F-test (variance ratio test)
- ~ Determining the values of f
- ~ F distribution using python

#### => Linear Algebra :

- ~ linear algebra
- ~ Vector
- ~ Scaler
- ~ Matrix
- ~ Matrix operations and manipulations
- ~ Dot product of two vectors
- ~ Transpose of a matrix
- ~ Linear independence of vectors
- ~ Rank of a matrix
- ~ Identity matrix or operator
- ~ Determinant of a matrix
- ~ Inverse of a matrix
- ~ Norm of a vector
- ~ Eigenvalues and eigenvectors
- ~ Calculus

#### => Solving Stats Problem with Python

#### => Stats Problem Implementation with Spicy

#### => Introduction to Machine Learning :

- ~ Ai vs ml vs dl vs ds
- ~ Supervised, unsupervised, semi-supervised, reinforcement learning
- ~ Train, test, validation split
- ~ Performance
- ~ Overfitting, under fitting
- ~ Bias vs variance

#### => Feature Engineering :

- ~ Handling missing data
- ~ Handling imbalanced data
- ~ Up-sampling
- ~ Down-sampling
- ~ Smote
- ~ Data interpolation
- ~ Handling outliers
- ~ Filter method
- ~ Wrapper method
- ~ Embedded methods
- ~ Feature scaling
- ~ Standardization

- ~ Mean normalization
- ~ Min-max scaling
- ~ Unit vector
- ~ Feature extraction
- ~ Pca (principle component analysis)
- ~ Data encoding
- ~ Nominal encoding
- ~ One hot encoding
- ~ One hot encoding with multiple categories
- ~ Mean encoding
- ~ Ordinal encoding
- ~ Label encoding
- ~ Target guided ordinal encoding
- ~ Covariance
- ~ Correlation check
- ~ Pearson correlation coefficient
- ~ Spearmans rank correlation
- ~ Vif

#### => Feature Selection :

- ~ Feature selection
- ~ Recursive feature elimination
- ~ Backward elimination
- ~ Forward elimination

#### => Exploratory Data Analysis :

- ~ Feature engineering and selection.
- ~ Analyzing bike sharing trends.
- ~ Analyzing movie reviews sentiment.
- ~ Customer segmentation and effective cross selling.
- ~ Analyzing wine types and quality.
- ~ Analyzing music trends and recommendations.
- ~ Forecasting stock and commodity prices

#### => Regression :

- ~ Linear regression
- ~ Gradient descent
- ~ Multiple linear regression
- ~ Polynomial regression
- ~ R square and adjusted r square
- ~ Rmse , mse, mae comparison
- ~ Regularized linear models
- ~ Ridge regression
- ~ Lasso regression
- ~ Elastic net
- ~ Complete end-to-end project with deployment on cloud and ui

#### => Logistics Regression :

- ~ Logistics regression in-depth intuition
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Hyper parameter tuning
- ~ Grid search cv
- ~ Randomize search cv
- ~ Data leakage
- ~ Confusion matrix
- ~ Precision, recall, f1 score , roc, auc
- ~ Best metric selection
- ~ Multiclass classification in lr
- ~ Complete end-to-end project with deployment in multi cloud platform

#### => Decision Tree :

- ~ Decision tree classifier
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Confusion matrix
- ~ Precision, recall, f1 score , roc, auc
- ~ Best metric selection
- ~ Decision tree repressor
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Performance metrics
- ~ Complete end-to-end project with deployment in multi cloud platform

#### => Support Vector Machines :

- ~ Linear svm classification
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Soft margin classification
- ~ Nonlinear svm classification
- ~ Polynomial kernel
- ~ Gaussian, rbf kernel
- ~ Data leakage
- ~ Confusion matrix
- ~ precision, recall, f1 score , roc, auc
- ~ Best metric selection
- ~ Svm regression
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Complete end-to-end project with deployment



## => Naive Bayes :

- ~ Bayes theorem
- ~ Multinomial naive Bayes
- ~ Gaussian naive Bayes
- ~ Various type of Bayes theorem and its intuition
- ~ Confusion matrix
- ~ precision , recall, f1 score , roc, auc
- ~ Best metric selection
- ~ Complete end-to-end project with deployment

## => Ensemble Technique and its Types :

- ~ Definition of ensemble techniques
- ~ Bagging technique
- ~ Bootstrap aggregation
- ~ Random forest (bagging technique)
- ~ Random forest regressor
- ~ Random forest classifier
- ~ Complete end-to-end project with deployment

## => Boosting :

- ~ Boosting technique
- ~ Ada boost
- ~ Gradient boost
- ~ Xgboost
- ~ Complete end-to-end project with deployment

## => Stacking :

- ~ Stacking technique
- ~ Complete end-to-end project with deployment

## => KNN :

- ~ Knn classifier
- ~ Knn regressor
- ~ Variants of knn
- ~ Brute force knn
- ~ K-dimension tree
- ~ Ball tree
- ~ Complete end-to-end project with deployment

## => Dimensionality Reduction :

- ~ The curse of dimensionality
- ~ Dimensionality reduction technique
- ~ Pca (principle component analysis)
- ~ Mathematics behind pca
- ~ Scree plots
- ~ Eigen-decomposition approach

## => Clustering :

- ~ Clustering and their types
- ~ K-means clustering
- ~ K-means++
- ~ Batch k-means
- ~ Hierarchical clustering
- ~ DbSCAN
- ~ Evaluation of clustering
- ~ Homogeneity, completeness and v-measure
- ~ Silhouette coefficient
- ~ Davies-bouldin index
- ~ Contingency matrix
- ~ Pair confusion matrix
- ~ Extrinsic measure
- ~ Intrinsic measure
- ~ Complete end-to-end project with deployment

## => Anomaly Detection :

- ~ Anomaly detection types
- ~ Anomaly detection applications
- ~ Isolation forest anomaly detection algorithm
- ~ Isolation forest anomaly detection algorithm
- ~ Support vector machine anomaly detection algorithm
- ~ DbSCAN algorithm for anomaly detection
- ~ Complete end-to-end project with deployment

## => Time-Series :

- ~ What is a time series?
- ~ Old techniques
- ~ Arima
- ~ ACF and PACF
- ~ Time-dependent seasonal components.
- ~ Autoregressive (AR),
- ~ Moving average (MA) and mixed ARMA- modeler.
- ~ The random walk model.
- ~ Box-jenkins methodology.
- ~ Forecasts with ARIMA and VAR models.
- ~ Dynamic models with time-shifted explanatory variables.
- ~ The Koyck transformation.
- ~ Partial adjustment and adaptive expectation models.
- ~ Granger's causality tests.
- ~ Stationarity, unit roots and integration
- ~ Time series model performance

- ~ Various approach to solve time series problem
- ~ Complete end-to-end project with deployment
- ~ Prediction of nifty stock price and deployment

=> NLP Basic :

- ~ Tokenization
- ~ Pos tags and chunking
- ~ Stop words
- ~ Stemming and lemmatization
- ~ Named entity recognition (ner)
- ~ Word vectorization (word embedding)
- ~ Tfidf
- ~ Complete end-to-end project with deployment

=> Machine Learning Pipeline :

- ~ Aws segmaker
- ~ Aure ml studio
- ~ Ml flow
- ~ Kube flow

=> Model Retraining Approach

=> Auto ML :

- ~ H2o
- ~ Pycaret
- ~ Auto sklearn
- ~ Auto time series
- ~ Auto vml
- ~ Auto gluon
- ~ Auto viz
- ~ Tpot
- ~ Auto neuro

=> Neural Network A Simple perception :

- ~ Detail mathematical explanation
- ~ Neural network overview and its use case.
- ~ Various neural network architect overview.
- ~ Use case of neural network in nlp and computer vision.
- ~ Activation function -all name
- ~ Multilayer network.
- ~ Loss functions. - all 10
- ~ The learning mechanism.
- ~ Optimizers. - all 10
- ~ Forward and backward propagation.
- ~ Weight initialization technique
- ~ Vanishing gradient problem
- ~ Exploding gradient problem
- ~ Visualization of nn

=> Hardware Setup - GPU :

- ~ Gpu introduction.
- ~ Various type of gpu configuration.
- ~ Gpu provider and its pricing.
- ~ Paper space gpu setup.
- ~ Running model in gpu

=> Tensor Flow Installation Environment Setup For Deep Learning :

- ~ Colab pro setup
- ~ Tensor flow installation 2.0 .
- ~ Tensor flow installation 1.6 with virtual environment.
- ~ Tensor flow 2.0 function.
- ~ Tensor flow 2.0 neural network creation.
- ~ Tensor flow 1.6 functions.
- ~ Tensor flow 1.6 neural network and its functions.
- ~ Keras introduction.
- ~ Keras in-depth with neural network creation.
- ~ Mini project in tensorflow.
- ~ Tensorspace
- ~ Tensorboard integration
- ~ Tensorflow playground
- ~ Netron

=> Pytorch :

- ~ pytorch installation.
- ~ Pytorch functional overview.
- ~ Pytorch neural network creation.

=> Mxnet :

- ~ Mxnet installation
- ~ Mxnet in depth function overview
- ~ Mxnet model creation and training

=> Keras Tuner :

- ~ Keras tuner installation and overview
- ~ Finding best parameter from keras tuner
- ~ Keras tuner application across various neural network

=> CNN Overview :

- ~ Cnn definition
- ~ Various cnn based architecture
- ~ Explanation end to end cnn network
- ~ Cnn explainer

- ~ *Training cnn*
- ~ *Deployment in azure cloud*
- ~ *Performance tuning of cnn network*

#### => Advance Computer Vision - Part 1 :

- ~ *Various cnn architecture with research paper and mathematics*
- ~ *Lenet-5 variants with research paper and practical*
- ~ *Alexnet variants with research paper and practical*
- ~ *Googlenet variants with research paper and practical*
- ~ *Transfer learning*
- ~ *Vggnet variants with research paper and practical*
- ~ *Resnet variants with research paper and practical*
- ~ *Inception net variants with research paper and practical*
- ~ *Darknet variants with research paper and practical*

#### => Advance Computer Vision - Part 2 :

- ~ *Object detection in-depth*
- ~ *Transfer learning*
- ~ *Rcnn with research paper and practical*
- ~ *Fast rcnn with research paper and practical*
- ~ *Faster r cnn with research paper and practical*
- ~ *Ssd with research paper and practical*
- ~ *Ssd lite with research paper and practical*

#### => Training of Custom Object Detection :

- ~ *Tfod introduction*
- ~ *Environment setup with tfod*
- ~ *Gpu vs tpu vs cpu*
- ~ *Various gpu comparison*

#### => Advance Computer Vision - Part 3 :

- ~ *Yolo v1 with research paper and practical*
- ~ *Yolo v2 with research paper and practical*
- ~ *Yolo v3 with research paper and practical*
- ~ *Yolo v4 with research paper and practical*
- ~ *Yolo v5 with research paper and practical*
- ~ *Retina net*
- ~ *Face net*
- ~ *Detectron2 with practical and live testing*

#### => Object Segmentation :

- ~ *Semantic segmentation*
- ~ *Panoptic segmentation*
- ~ *Masked rcnn*
- ~ *Practical with detectron*
- ~ *Practical with tfod*

#### => Object Tracking :

- ~ *Detail of object tracking*
- ~ *Kalman filtering*
- ~ *Sort*
- ~ *Deep sort*
- ~ *Object tracking live project with live camera testing*

#### => OCR :

- ~ *Introduction to ocr*
- ~ *Various framework and api for ocr*
- ~ *Practical implementation of ocr*
- ~ *Live project deployment for bill parsing*

#### => Image Captioning :

- ~ *Image captioning overview*
- ~ *Image captioning project with deployment*

#### => Tensorflow JS :

- ~ *Tensorflow js overview*
- ~ *Tfjs implementation*

#### => Model Conversion :

- ~ *Tfjs*
- ~ *Tflite*
- ~ *Tfvt*
- ~ *Torch to tf model*
- ~ *Mxnet to tf conversion*

#### => Advance NLP with Deep Learning :

- ~ *Overview computational linguistic.*
- ~ *History of nlp.*
- ~ *Why nlp*
- ~ *Use of nlp*

#### => Text Processing Importing Text :

- ~ *Web scrapping.*
- ~ *Text processing*
- ~ *Understanding regex.*
- ~ *Text normalization*
- ~ *Word count.*
- ~ *Frequency distribution.*
- ~ *Text annotation.*
- ~ *Use of annotator.*
- ~ *String tokenization*
- ~ *Annotator creation.*

- ~ Sentence processing.
- ~ Lemmatization in text processing
- ~ Pos
- ~ Named entity recognition
- ~ Dependency parsing in text.
- ~ Sentimental analysis

=> Spacy :

- ~ Spacy overview.
- ~ Spacy function
- ~ Spacy function implementation in text processing.
- ~ Pos tagging, challenges and accuracy.
- ~ Entities and named entry recognition
- ~ Interpolation, language models
- ~ Nltk
- ~ Text blob
- ~ Stanford nlp

=> RNN :

- ~ Recurrent neural networks.
- ~ Long short term memory (lstm)
- ~ Bi lstm.
- ~ Stacked lstm
- ~ Gru implementation.
- ~ Building a story writer using character level rnn.

=> Word Embedding :

- ~ Word embedding
- ~ Co-occurrence vectors
- ~ Word2vec
- ~ Doc2vec

=> Attention Based Model :

- ~ Seq 2 seq.
- ~ Encoders and decoders.
- ~ Attention mechanism.
- ~ Attention neural networks
- ~ Self-attention

=> Transfer Learning in NLP :

- ~ Introduction to transformers.
- ~ Bert model.
- ~ Elmo model.
- ~ Gpt1 model
- ~ Gpt2 model.
- ~ Albert model.
- ~ Distilbert model

=> Deployment of Model and Performance Tuning :

- ~ Deep learning model deployment strategies.
- ~ Deep learning project architecture
- ~ Deep learning model deployment phase.
- ~ Deep learning model retraining phase.
- ~ Deep learning model deployment in aws.
- ~ Deep learning model deployment in azure.
- ~ Deep learning model deployment in gcloud.

=> API for Speech and Vision :

- ~ AWS
- ~ Azure
- ~ GCP

=> Big Data Introduction :

- ~ What is big data?
- ~ Big data application
- ~ Big data pipeline

=> Hadoop :

- ~ Hadoop introduction
- ~ Hadoop setup and installation

=> Spark :

- ~ Spark
- ~ Spark overview.
- ~ Spark installation.
- ~ Spark rdd.
- ~ Spark data frame.
- ~ Spark architecture.
- ~ Spark ml lib
- ~ Spark NLP
- ~ Spark linear regression
- ~ Spark logistic regression
- ~ Spark decision tree
- ~ Spark naive bayes
- ~ Spark xg boost.
- ~ Spark time series
- ~ Spark deployment in local server
- ~ Spark job automation with
- ~ Scheduler

=> Kafka :

- ~ Kafka introduction

- ~ Kafka installation
- ~ Spark streaming
- ~ Spark with Kafka

=> ML Ops :

- ~ Jenkins
- ~ Kubernetes
- ~ Elasticsearch
- ~ Kibana
- ~ Git

=> SQL :

- ~ Introduction
- ~ ER Diagram
- ~ Schema Design
- ~ Normalization
- ~ SQL SELECT Statement
- ~ SQL SELECT Using common functions
- ~ SQL JOIN Overview
- ~ INNER JOIN
- ~ LEFT JOIN
- ~ RIGHT JOIN
- ~ FULL JOIN
- ~ SQL Best Practice
- ~ INNER JOIN - Advanced
- ~ INNER JOIN & LEFT JOIN Combo
- ~ SELF JOIN
- ~ Joins & Aggregation - Subqueries
- ~ Sorting
- ~ Independent Subqueries
- ~ Correlated Subqueries
- ~ Analytic Function
- ~ Set Operations
- ~ SQL Views
- ~ Create a view
- ~ Create a view using DDL
- ~ SQL Insert - Advanced Technique
- ~ INSERT to create a table
- ~ INSERT new data to an existing table-1
- ~ INSERT new data to an existing table-2
- ~ INSERT new data to an existing table-3
- ~ INSERT new data to an existing table-4
- ~ SQL Update - Advanced Technique and TCL
- ~ SQL DELETE and TCL
- ~ SQL Constraints
- ~ SQL Aggregations
- ~ SQL Programmability
- ~ SQL Query Performance
- ~ SQL Xtras

=> Advance Excel :

- ~ Microsoft Excel Fundamentals
- ~ Entering and Editing Text and Formulas
- ~ Working with Basic Excel Functions
- ~ Modifying an Excel Worksheet
- ~ Formatting Data in an Excel Worksheet
- ~ Inserting Images and Shapes into an Excel Worksheet
- ~ Creating Basic Charts in Excel
- ~ Printing an Excel Worksheet
- ~ Working with Excel Templates
- ~ Working with an Excel List
- ~ Excel List Functions
- ~ Excel Data Validation
- ~ Importing and Exporting Data
- ~ Excel PivotTables
- ~ Working with Excel's PowerPivot Tools
- ~ Working with Large Sets of Excel Data
- ~ Conditional Functions
- ~ Lookup Functions
- ~ Text Based Functions
- ~ Auditing an Excel Worksheet
- ~ Protecting Excel Worksheets and Workbooks
- ~ Mastering Excel "What If?" Tools
- ~ Automating Repetitive Tasks in Excel with Macros
- ~ Macro Recorder Tool
- ~ Excel VBA Concepts
- ~ Advance VBA
- ~ Preparing and Cleaning Up Data with VBA
- ~ VBA to Automate Excel Formulas
- ~ Preparing Weekly Report
- ~ Working with Excel VBA User Forms
- ~ Importing Data from Text Files

=> Tableau :

- ~ Talking about Business Intelligence
- ~ Tools and Methodologies used in BI
- ~ Why Visualization is getting more popular
- ~ Why Tableau?
- ~ Gartner Magic Quadrant of Market Leaders
- ~ Future business impact of BI

- ~ Tableau Products
- ~ Tableau Architecture
- ~ BI Project Execution
- ~ Tableau Installation in local system
- ~ Introduction to Tableau Prep
- ~ Tableau Prep Builder User Interface
- ~ Data Preparation techniques using Tableau Prep Builder tool
- ~ How to connect Tableau with different data source
- ~ Visual Segments
- ~ Visual Analytics in depth
- ~ Filters, Parameters & Sets
- ~ Tableau Calculations using functions
- ~ Tableau Joins
- ~ Working with multiple data source (Data Blending)
- ~ Building Predictive Models
- ~ Dynamic Dashboards and Stories
- ~ Sharing your Reports
- ~ Tableau Server
- ~ User Security
- ~ Scheduling
- ~ PDF File
- ~ JSON File
- ~ Spatial File
- ~ Statistical File
- ~ Microsoft SQL Server
- ~ Salesforce
- ~ AWS
- ~ Azure
- ~ Google Analytics
- ~ R
- ~ Python
- ~ Hadoop
- ~ OneDrive
- ~ Microsoft Access
- ~ SAP HANA
- ~ SharePoint
- ~ Snowflake
- ~ Subject
- ~ Planning
- ~ Pen & Paper approach
- ~ Tools
- ~ Color theme
- ~ Shapes
- ~ Fonts
- ~ Image Selection
- ~ text position
- ~ visual placing
- ~ Story layout & design
- ~ Dashboard planning

=> Power BI :

- ~ Power BI introduction and overview
- ~ Key Benefits of Power BI
- ~ Power BI Architecture
- ~ Power BI Process
- ~ Components of Power BI
- ~ Power BI - Building Blocks
- ~ Power BI vs other BI tools
- ~ Power Installation
- ~ Overview of Power BI Desktop
- ~ Data Sources in Power BI Desktop
- ~ Connecting to a data Sources
- ~ Query Editor in Power BI
- ~ Views in Power BI
- ~ Field Pane
- ~ Visual Pane
- ~ Custom Visual Option
- ~ Filters
- ~ Introduction to using Excel data in Power BI
- ~ Exploring live connections to data with Power BI
- ~ Connecting directly to SQL Azure, HD Spark, SQL Server Analysis Services/ My SQL
- ~ Import Power View and Power Pivot to Power BI
- ~ Power BI Publisher for Excel
- ~ Content packs
- ~ Introducing Power BI Mobile
- ~ Power Query Introduction
- ~ Query Editor Interface
- ~ Clean and Transform your data with Query Editor
- ~ Data Type
- ~ Column Transformations vs Adding Columns
- ~ Text Transformations
- ~ Cleaning irregularly formatted data -Transpose
- ~ Date and Time Calculations
- ~ Advance editor: Use Case
- ~ Query Level Parameters
- ~ Combining Data Merging and Appending
- ~ Data Modelling
- ~ Calculated Columns
- ~ Measures/New Quick Measures

- ~ Calculated Tables
- ~ Optimizing Data Models
- ~ Row Context vs Set Context
- ~ Cross Filter Direction
- ~ Manage Data Relationship
- ~ Why is DAX important?
- ~ Advanced calculations using Calculate functions
- ~ DAX queries
- ~ DAX Parameter Naming
- ~ Time Intelligence Functions
- ~ Types of visualization in a Power BI report
- ~ Custom visualization to a Power BI report
- ~ Matrixes and tables
- ~ Getting started with color formatting and axis properties
- ~ Change how a chart is sorted in a Power BI report
- ~ Move, resize, and pop out a visualization in a Power BI report
- ~ Drill down in a visualization in Power BI

=> GPT-3

=> GAN

=> Reinforcement Learning

## Project details :-

=> Python Project :

- ~ Weeding script
- ~ Image resizing
- ~ Jupyter notebook merging, reading etc.
- ~ Sending emails
- ~ Weather app
- ~ Memes generator
- ~ Food log app
- ~ Web scrapping
- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis.
- ~ Integration with web portal.
- ~ Integration with rest api, web portal and mongo db. on azure
- ~ Deployment on web portal on azure.
- ~ Text mining
- ~ Social media data churn
- ~ Mass copy, paste

=> Chatbot Projects :

- ~ Chatbot using Microsoft Luis
- ~ Chatbot using google dialog flow
- ~ Chatbot using amazon lex
- ~ Chatbot using rasa nlu
- ~ Deployment of Chabot with web , telegram , WhatsApp, skype

=> Major Projects :

- ~ Healthcare analytics prediction of medicines based on Fitbit band.
- ~ Revenue forecasting for startups.
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ anomaly detection in inventory packaged material.
- ~ Fault detection in wafers based on sensor data.
- ~ Demand forecasting for fmcg product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with zomato dataset.
- ~ Fault detection in wafers based on sensor data.
- ~ Cement strength reg.
- ~ Credit card fraud.
- ~ Forest cover classification.
- ~ Fraud detection.
- ~ Income prediction.
- ~ Mushroom classifier.
- ~ phishing classifier
- ~ Thyroid detection.
- ~ Visibility climate

=> Computer Vision Project :

- ~ Traffic surveillance system.
- ~ Object identification.
- ~ Object tracking.
- ~ Object classification.
- ~ Tensorflow object detection.
- ~ Image to text processing.
- ~ Speech to speech analysis.
- ~ Vision based attendance system

=> Mini NLP Project :

- ~ Machine translation.
- ~ Abstractive text summarization.
- ~ Keyword spotting.
- ~ Language modelling.
- ~ Document summarization

=> NLP Transfer Learning Project :

- ~ Deployment and integration with UI machine translation.
- ~ Question answering (like chat bot)

- ~ Sentiment analysis imdb.
- ~ Text search (with synonyms).
- ~ Text classifications.
- ~ Spelling corrector.
- ~ Entity (person, place or brand) recognition.
- ~ Text summarization.
- ~ Text similarity (paraphrase).
- ~ Topic detection.
- ~ Language identification.
- ~ Document ranking.
- ~ Fake news detection
- ~ Plagiarism checker
- ~ Text summarization extractive
- ~ Text summarization abstractive.

=> NLP End to End Project with Architecture and Deployment :

- ~ Movie review using bert
- ~ NER using Bert
- ~ Pos bert
- ~ Text generation gpt 2
- ~ Text summarization xlnet
- ~ Abstract bert
- ~ Machine translation
- ~ Nlp text summarization custom
- ~ Keras/tensorflow
- ~ Language identification
- ~ Text classification using fast bert
- ~ Neuralcore
- ~ Detecting fake text using gltr with bert and gpt2
- ~ Fake news detector using gpt2
- ~ Python plagiarism checker type a message
- ~ Question answering

=> NLP Project End to End with Deployment in Various Cloud and UI Integration :

- ~ Topic modeling.
- ~ Word sense disambiguation
- ~ Text to speech
- ~ Keyword spotting
- ~ Document ranking
- ~ Text search (with synonyms)
- ~ Language modeling
- ~ Spam detector
- ~ Image captioning

=> SQL Project :

- ~ Ecommerce Analysis - Tableau Integration
- ~ Sales Data Analysis - Tableau Integration

=> Tableau Project :

- ~ Human Resource - Tableau
- ~ Supply Chain - Tableau
- ~ Sale Return - Tableau
- ~ E-Commerce Customer Analysis
- ~ Project Management Dashboard
- ~ Sales Dashboard

=> Power BI Project :

- ~ Cost Insights - Power BI
- ~ Management Insights- Power BI
- ~ Retail Insights- Power BI



# Azure Data scientist Associate DP100

---

Topic Name : CLOUD

Sub-topic Name : AZURE

Course link : <https://ineuron.ai/course/Azure-Data-scientist-Associate-DP100>

## Course Description :-

Through this course, candidates for the Azure Data Scientist Associate certification should will be able to implement and execute machine learning workloads on Azure using data science and machine learning abilities.

## Course Features :-

- => Roadmap
- => Real-Time implementation
- => ML/DL model testing and monitoring
- => Scenario-based questions
- => Challenges
- => Downloadable resources
- => Quizzes
- => Completion Certificate

## What you will learn :-

- => Learn basics of Python programming language
- => How to make models and implement solutions for Azure ML Platform
- => Tackle the Microsoft DP-100 Microsoft Azure Machine Learning test
- => Be up-to-date on the latest updates of this ever-changing platform

## Requirements :-

- => No prior knowledge in programming as well as cloud
- => Zero-knowledge of Azure portal
- => Azure free or paid account
- => A system with internet connection
- => Your dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Course introduction :
  - ~ Introduction to Course Preview
  - ~ Create Your Free Azure Account Preview
- => Basics of Machine Learning :
  - ~ Machine Learning part 1
  - ~ Machine Learning part 2
  - ~ Types of Machine Learning
  - ~ Walkthrough of Azure ML
- => Basic statistics :
  - ~ Statistics part 1
  - ~ Statistics part 2
  - ~ Statistics part 3
  - ~ Statistics part 4
  - ~ Statistics part 5
- => Getting started with Azure ML :
  - ~ What is Azure ML studio
  - ~ Overview of Azure ML studio
  - ~ Azure ML experiment workflow
- => Data processing :
  - ~ How to upload data
  - ~ How to import data
  - ~ Add rows/columns and remove duplicates
  - ~ Add Rows/Columns and Remove duplicates
  - ~ Apply SQL Transformation,Clean Missing Data
  - ~ Sample and data partition
  - ~ Split data
- => Classification :

- ~ Different classification algorithms
- ~ What is logistic regression
- ~ Hands-On - Logistic regression part 1
- ~ Hands-On - Logistic regression part 2
- ~ Logistic regression - Understand parameters and their impact
- ~ Confusion matrix, AUC, accuracy etc
- ~ Logistic regression model selection and impact analysis
- ~ Demo on Logistic regression part 1
- ~ Demo on Logistic regression part 2
- ~ What is Decision tree
- ~ What is Bagging and Boosting ?
- ~ Two class boosted Decision tree
- ~ Demo on two class boosted Decision tree
- ~ Decision forest parameters explained
- ~ Demo on two class Decision forest
- ~ Demo multi-class Decision forest IRIS data
- ~ What is SVM ?
- ~ Demo on SVM part 1
- ~ Demo on SVM part 2

#### => Hyperparameter tuning :

- ~ What is hyperparameter tuning
- ~ Demo on hyperparameter tuning

#### => Deploy webservice :

- ~ Azure ML webservice-prepare the experiment for webservice
- ~ Demo Deploy Machine Learning Model as a web service
- ~ Demo - use the web service example of excel

#### => Regression Analysis :

- ~ What is Linear regression
- ~ Regression analysis comma metrics
- ~ Demo linear regression using OLS
- ~ R squared
- ~ Gradient descent
- ~ Online gradient descent
- ~ Demo online gradient
- ~ What is a regression tree
- ~ What is boosted decision tree
- ~ Demo boosted decision tree part 1
- ~ Demo boosted decision tree part 2

#### => Clustering :

- ~ What is cluster analysis theory
- ~ What is cluster analysis
- ~ Demo on cluster analysis part 1
- ~ Demo on cluster analysis part 2

#### => Hands-on data processing :

- ~ How to Summarize Data
- ~ Demo on summarizing data
- ~ What is outliers and outlier treatment
- ~ Demo on outliers
- ~ Cleaning missing data with MICE
- ~ Hands-on cleaning missing data with MICE
- ~ SMOTE- create new synthetic observations
- ~ Demo on SMOTE
- ~ Data normalization - scale and reduce
- ~ Demo on data normalization
- ~ What is PCA
- ~ Demo on PCA
- ~ Join data
- ~ Demo on join data

#### => Feature selection :

- ~ Feature selection
- ~ Pearson correlation coefficient
- ~ Chi-Square test of independence
- ~ Kendall correlation coefficient
- ~ Spearman rank correlation
- ~ Demo on filter based selection
- ~ Fisher based LDA
- ~ Demo on LDA

#### => Recommendation system :

- ~ What is recommendation
- ~ Data preparation using recommender split
- ~ What is matchbox recommender
- ~ How to score the matchbox recommender
- ~ Restaurant recommendation experiment
- ~ Understanding the matchbox recommendation results

#### => Text analytics and Natural language Processing :

- ~ What is text analytics and Natural Language Processing
- ~ Text pre-processing
- ~ Bag of words and n-gram models for text features
- ~ Feature hashing
- ~ Demo on text analytics

#### => About DP certification exam :

- ~ Exam curriculum discussion

=> Azure Machine learning with Studio Designer :

- ~ What this section covers for topics

=> Set up Azure Machine Learning Workspace :

- ~ Azure ML service architecture
- ~ Create the Azure ML workspace
- ~ View and manage workspace
- ~ Overview of new Azure ML studio
- ~ What is Azure ML datastore and dataset
- ~ Create and register a datastore
- ~ How to create dataset
- ~ Understanding the Azure ML compute resources
- ~ Create a compute cluster and compute instance

=> Train models and Azure pipeline :

- ~ What is the Azure ML pipeline
- ~ Create a pipeline using Azure ML designer
- ~ Submit the designer pipeline run
- ~ Create an inference pipeline

=> Deploy and consume the models :

- ~ Deploy a real-time endpoint using designer
- ~ Create a batch inference pipeline
- ~ Run a batch inference pipeline from designer

=> Pandas and scikit learn in designer/classic studio :

- ~ Pandas - import data from experiments
- ~ Selecting columns using pandas
- ~ Clean missing data
- ~ Edit metadata of columns using pandas
- ~ Summary statistics of data
- ~ Remove outliers
- ~ Covert and save a delimited file using pandas
- ~ Data normalization
- ~ Label encoding of categorical string data
- ~ What is encoding
- ~ Hot encoding using pandas get\_dummies
- ~ Split the data for training and testing
- ~ Build logistic regression using python part 1
- ~ Build logistic regression using python part 2

=> Azure machine learning with Azure ML SDK :

- ~ Introduction to Azure ML SDK

=> DP 100 set up Azure Machine Learning workspace :

- ~ Create Azure ML workspace using SDK
- ~ Verify the workspace and write the workspace config file
- ~ Create and register a datastore using Azure ML SDK
- ~ Create and register a dataset using SDK
- ~ Access workspace, datastore and datasets using SDK
- ~ Pandas dataframe and Azure ML dataset conversions
- ~ Upload local data to storage account via datastore

=> DP 100 run and experiments and train models :

- ~ Problem statement - run a sample experiment and log values
- ~ Run a sample experiment using Azure ML SDK - part 1
- ~ Run a sample experiment using Azure ML SDK - part 2
- ~ Run a script in Azure ML environment - part 1
- ~ Run a script in Azure ML environment - part 2
- ~ Run a script in Azure ML environment - part 3
- ~ Run a script in Azure ML environment - part 4
- ~ Run a script in Azure ML environment - part 5
- ~ Train and run a model script in Azure ML part 1
- ~ Train and run a model script in Azure ML part 2
- ~ Train and run a model script in Azure ML part 3
- ~ Train and run a model script in Azure ML part 4
- ~ Train and run a model script in Azure ML part 5
- ~ Provisioning compute cluster using SDK
- ~ Automate model training using Azure ML SDK
- ~ Automate model training - define pipeline steps
- ~ Automate model training - define run configuration
- ~ Automate model training - define build and run
- ~ Detour - command line arguments
- ~ Automate model training - create dataprep step
- ~ Automate model training - create training step
- ~ Run the pipeline and see the results

=> Python scripts in Azure ML designer :

- ~ Simple python script in designer
- ~ Execute python script using zip bundle
- ~ Execute python script using zip bundle - hands-on

=> Azure AutoML :

- ~ What is Azure AutoML?
- ~ Use the automated ML interface in Azure Machine Learning studio
- ~ View the AutoML run result
- ~ Use automated ML from the Azure Machine Learning SDK
- ~ Retrieve the best model and view results

=> Azure hyperdrive :

- ~ Introduction to Azure hyperdrive

- ~ Define the hyperparameter search space
- ~ Select a sampling method
- ~ Define early termination options
- ~ Configure the hyperdrive run
- ~ Create the training script for hyperdrive run
- ~ Retrieve the best model

=> Model explainers to interpret models :

- ~ Why is model explanation necessary?
- ~ Understanding shapley value
- ~ Interpretability techniques in Azure
- ~ Implement interpretability - initial set-up
- ~ Implement interpretability - global explanations
- ~ Implement interpretability - local explanations part 1
- ~ Implement interpretability - local explanations part 2
- ~ Run interpret model script in Azure workspace
- ~ Visualize explanations in Azure ML studio
- ~ Retrieve/Download feature importance values.

=> Model registration and deployment using Azure ML SDK :

- ~ Model deployment steps
- ~ Understanding model/object serialization
- ~ Hands-on - serialization using joblib
- ~ Handling onehotencoding/dummy values in production
- ~ Hands-on - dummy variables in production
- ~ Train the model for web service deployment
- ~ Register the model using run\_id
- ~ Register the model using a local.pkl file
- ~ Retrieve all the registered models from the workspace
- ~ Provisioning AKS production cluster using SDK
- ~ Create the inference and deployment configuration for webservice
- ~ Entry script - init function
- ~ Understanding data processing using JSON, dictionary and dataframe
- ~ Entry script - run function
- ~ Create web services deployment object
- ~ Deploy a real-time endpoint using SDK
- ~ Consume the web service from Python program
- ~ Consume the web service as an end point.

=> Databricks with Azure ML :

- ~ Databricks with Azure ML
- ~ Databricks update to DP-100
- ~ (Optional) what is Big data?
- ~ (Optional) what is Hadoop?
- ~ Create an Azure databricks workspace
- ~ Note on deleting databricks resource in Azure portal
- ~ Note on increasing vCPU quota limits
- ~ Create an Azure databricks cluster
- ~ Create an Azure databricks cluster
- ~ Link Azure ML workspace with the databricks workspace
- ~ Create and run notebooks in Azure databricks part-1
- ~ Create and run notebooks in Azure databricks part-2
- ~ Mount blob storage to databricks using duties part-1
- ~ Mount blob storage to databricks using duties part-2
- ~ Run a sklearn experiment with databricks notebook
- ~ Overview to run a training script using databricksstep in a pipeline
- ~ Saving data to Azure blob storage from databricks
- ~ Passing parameters between Azure databricks notebooks
- ~ Attach a databricks cluster as an attached computer target
- ~ Verify databricks cluster as attached compute
- ~ Databricks pipeline - initial set-up
- ~ Databricks pipeline - build databricksstep
- ~ Databricks pipeline - databricks and Python notebook
- ~ Databricks pipeline - submit the pipeline and verify the output

=> Azure fundamentals :

- ~ Azure storage services
- ~ Azure virtual machine
- ~ Azure network resources

=> Product overview :

- ~ Product overview

=> Python basics :

- ~ An important note
- ~ Install Anaconda
- ~ Hello world and know your environment
- ~ Python identifiers and reserved words
- ~ What are variable and variable types
- ~ Basic operators
- ~ Decision making
- ~ Python loops
- ~ Python numbers
- ~ Python list
- ~ Python tuple
- ~ Python string
- ~ Python sets
- ~ Python dictionary
- ~ Python functions
- ~ Python arguments

- ~ *Object-Oriented programming*
- ~ *Packages and modules in python*

# Class 10th Physics

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10th-Physics>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life.

Class 10th is crucial and is the foundation for higher education of students. It will make the foundation strong for the higher classes.

Physics is the Study of Physical World. It is the natural science that studies matter, its fundamental constituents, its motion and behavior through space and time, and the related entities of energy and force. Main goal of Physics is to understand how the universe behaves.

## Course Features :-

=> Self Paced Videos

=> Completion Certificate

## What you will learn :-

=> Light Reflection and Refraction

=> The Human Eye and the Colourful World

=> Magnetic effects of electric current

=> Sources of Energy

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Jawala Prakash :

~

## Curriculum details :-

=> Light Reflection and Refraction :

- ~ Lecture 1 : Introduction Preview
- ~ Lecture 2 : Details of Topics that are covered Preview
- ~ Lecture 3 : Definition of Light, Laws of Reflection, Image Formation in Plane Mirror, Characteristics of Image formed by Plane Mirror, Number of Images formed by two Plane Mirrors Placed at some angle
- ~ Lecture 4 : Spherical\_Mirror\_Terminologies, Concave Mirror , Convex Mirror, Pole, Centre of Curvature, Radius of Curvature, Principal Axis, Aperture, Principal Focus, Focus, Relation between Radius of Curvature and Focal Length.
- ~ Lecture 5 : Real Image Vs Virtual Image Preview
- ~ Lecture 6 : Ray Diagram Rules
- ~ Lecture 7 : Image formation by Concave Mirror
- ~ Lecture 8 : Uses of Concave Mirror
- ~ Lecture 9 : Sign Convention for Reflection by Plane Mirror, Mirror Formula and Magnification
- ~ Lecture 10 : NCERT Problems
- ~ Lecture 11 : Refraction Introduction
- ~ Lecture 12 : Laws of Refraction, The Refractive Index,
- ~ Lecture 13 : Refraction through Rectangular Glass Slab,
- ~ Lecture 14 : Introduction of Lenses and related Terminologies
- ~ Lecture 15 : Lens Image Formation
- ~ Lecture 16 : Lens Formula Magnification
- ~ Lecture 17 : NCERT Lens Problems
- ~ Lecture 18 : Two thin Lenses in Contact, Power of Lens
- ~ Lecture 19 : NCERT Exemplar Problems

=> The Human Eye and the Colourful World :

- ~ Lecture 1 : Introduction and Topics
- ~ Lecture 2 : The Human Eye , Parts of an Eye
- ~ Lecture 3 : Power of Accommodation
- ~ Lecture 4 : Defects of Vision and their Correction, Myopia, Hypermetropia, Presbyopia
- ~ Lecture 5 : Refraction of Light through Prism
- ~ Lecture 6 : Dispersion of White Light by Glass Prism
- ~ Lecture 7 : Atmospheric Refraction, Twinkling of Stars, Advance
- ~ Lecture 8 : Scattering Light Phenomenon
- ~ Lecture 9 : NCERT Questions
- ~ Lecture 10 : NCERT Exemplar Questions

=> Magnetic effects of electric current :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Oersted Experiment showing magnetic effect of Electric current, Magnetic Field and Magnetic Field Lines, Drawing magnetic field lines using Compass needle

- ~ Lecture 3 : Magnetic Field due to Current Carrying Conductor, Maxwell Right Hand Thumb Rule / Corkscrew Rule
- ~ Lecture 4 : Magnetic Field due to Current through a Circular Loop, Clock Face Rule
- ~ Lecture 5 : Magnetic Field due to a Current in a Solenoid, Electromagnet
- ~ Lecture 6 : Force on a Current Carrying Conductor in a Magnetic Field, Fleming's Left Hand Rule
- ~ Lecture 7 : Applying Fleming's Left Hand Rule to find the direction of Force on Current Current conductor in a Magnetic Field.
- ~ Lecture 8 : Electric Motor
- ~ Lecture 9 : Electromagnetic Induction, Fleming's Right Hand Rule
- ~ Lecture 10 : Electric Generator, DC Generator
- ~ Lecture 11 : Domestic Electric Circuit, Advantage of Parallel Connection in Domestic Electric Circuit

=> Sources of Energy :

- ~ Lecture 1: Introduction & Course Topics
- ~ Lecture 2 : Characteristics Good Fuel
- ~ Lecture 3 : Conventional Vs NonConventional Energy Sources
- ~ Lecture 4: All About Fossil Fuels
- ~ Lecture 5 : Thermal & Hydel Power Plants
- ~ Lecture 6 : Bio Gas & WindEnergy\_Discussion
- ~ Lecture 7 : Solar Energy
- ~ Lecture 8 : Solar Cells
- ~ Lecture 9 : Energy From Sea
- ~ Lecture 10 : Geothermal Energy
- ~ Lecture 11 : Nuclear Energy
- ~ Lecture 12 : NCERT Intext Questions Discussion
- ~ Lecture 13 : Environmental Consequences
- ~ Lecture 14 : NCERT Exercise & Exemplar Problems Discussion

# Mern Stack

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Mern-Stack>

## Course Description :-

Learn how to build complete online applications with MongoDB, Express.js, React.js, and Nodejs. Mern stack is one of the most versatile tech stacks available. Learn front-end and hybrid mobile development, as well as server-side support, to build a multi-platform solution.

## Course Features :-

- => Challenges
- => Assignments in each module
- => Quizzes
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => HTML
- => CSS
- => Bootstrap
- => JAVASCRIPT
- => Bootstrap
- => Node js & Express js
- => Database
- => Validation
- => JWT
- => React js
- => GraphQL

## Requirements :-

- => No Prior knowledge is required
- => A System with Internet Connection
- => Your dedication

## Instructors :-

- => Syed Ashraf :
  - ~ Full Stack Engineer at TensorGo Technologies

## Curriculum details :-

- => HTML :
  - ~ How do websites work? Preview
  - ~ HTML vs CSS vs Javascript
  - ~ HTML files
  - ~ Doctype & HTML Boilerplate
  - ~ Spaces & Line Breaks
  - ~ Heading Tag
  - ~ Paragraph & Pre Tag
  - ~ Difference between Elements, Attributes & Tags
  - ~ Comments
  - ~ Useful Tags
  - ~ Nesting of Tags
  - ~ Extensions in HTML
  - ~ Live Server in VSCode
  - ~ Formatting Tags
  - ~ Article in HTML
  - ~ Time & Address Tag
  - ~ Quote & Cite
  - ~ Strike
  - ~ Progress Bar
  - ~ Anchor Tag Styling
  - ~ Image Tag
  - ~ HTML Table
  - ~ List
  - ~ Input Tags,iframe



- ~ Forms
- ~ Video & Audio
- ~ iframe
- ~ Embed pdf
- ~ Maps
- ~ Symbols
- ~ Meta Tags
- ~ SVG
- ~ Emoji
- ~ Validate your HTML Code

=> CSS :

- ~ CSS Introduction Preview
- ~ Inline vs Internal vs External
- ~ Priority between Inline, Internal & External
- ~ Multiple Properties in Single Element
- ~ Types of Selectors
- ~ Priority between Id, Class & Element
- ~ Comments
- ~ Colors
- ~ Background
- ~ Border
- ~ Height & Width
- ~ Padding
- ~ Margin
- ~ Box Model
- ~ Text Properties
- ~ Anchor Tag Styling
- ~ Fonts
- ~ Cursor
- ~ !Important in CSS
- ~ Box Shadow
- ~ Opacity
- ~ Filter
- ~ Gradient
- ~ Overflow
- ~ List
- ~ Tables
- ~ Box Sizing
- ~ Inherit & Initial
- ~ Object Fit
- ~ Pseudo Classes
- ~ Pseudo Elements
- ~ Display
- ~ Position
- ~ Z-Index
- ~ Floats
- ~ 2D Transform
- ~ Transitions
- ~ Flex
- ~ Flex Direction & Wrap
- ~ Justify & Align in Flex
- ~ Order in Flex
- ~ Grow & Basis in Flex
- ~ Align Items in Flex
- ~ Grids
- ~ Rows, Columns & Gap in Grids
- ~ Justify & Align in Grids
- ~ CSS Validator (Final Video)

=> JAVASCRIPT :

- ~ Introduction
- ~ Running Javascript in Browser
- ~ Console
- ~ Strings & Numbers
- ~ var, let & const
- ~ Data Types
- ~ Type Conversions
- ~ Arithmetic Operators
- ~ Assignment Operator
- ~ Comparison Operator
- ~ Logical Not, Or and And
- ~ Swap Numbers
- ~ String Handling
- ~ String Searching
- ~ Arrays
- ~ Objects
- ~ Dates
- ~ Maths
- ~ If & Else
- ~ Challenge - If & Else
- ~ Switch Case
- ~ Challenge - Switch Case
- ~ JS Loops
- ~ For Loops
- ~ Nested Loops
- ~ Break & Continue
- ~ Arrays, Strings & Objects
- ~ For-in

- ~ For-of
- ~ While Loops
- ~ Do while Loops
- ~ Loops Exercises
- ~ Functions
- ~ Variable Scopes in Functions
- ~ Nested Functions
- ~ Parameters & Arguments
- ~ How function is useful?
- ~ Return in Function
- ~ Anonymous Functions
- ~ Calculator Exercise
- ~ Arrow Functions
- ~ forEach
- ~ maps
- ~ String Literals
- ~ Filter, Reduce & Every
- ~ Spread Operator
- ~ Challenge
- ~ Window & Document
- ~ Document Access
- ~ innerText & innerHTML
- ~ HTML Calculator
- ~ Query Selector
- ~ Styling in JS
- ~ Advance DOM Manipulation
- ~ Events
- ~ Basic Events
- ~ Time Events
- ~ Pop-up Boxes
- ~ Error Handling
- ~ Form Validation
- ~ Asynchronous JS
- ~ this keyword
- ~ useStrict
- ~ Hoisting
- ~ Local Storage
- ~ Session Storage
- ~ Cookies
- ~ Cookies vs Local Storage vs Session Storage
- ~ JSON vs Object literals
- ~ API
- ~ Fetching
- ~ Methods & Status Codes
- ~ Post Method
- ~ Put Method
- ~ Guess the Number
- ~ Generators
- ~ Regex

#### => Bootstrap :

- ~ Introduction
- ~ Bootstrap in Project
- ~ Containers
- ~ Buttons
- ~ Alerts
- ~ Badges
- ~ Button Groups
- ~ Cards in Bootstrap
- ~ Grids
- ~ Advance Column Properties
- ~ Image Slider
- ~ Dropdowns
- ~ Modal in Bootstrap
- ~ OffCanvas
- ~ Popovers
- ~ Spinners
- ~ Toast
- ~ Accordion
- ~ Bootstrap Navs
- ~ NavBars in Bootstrap
- ~ Forms
- ~ Helper Classes
- ~ Utilities Classes
- ~ Flex
- ~ Interactions
- ~ Utilities Properties
- ~ Typography in Bootstrap
- ~ Handling in Images & Tables
- ~ Build a Bootstrap Website

#### => Nodejs :

- ~ Introduction & Installation
- ~ Global Objects
- ~ Modules
- ~ OS Module
- ~ Path Module
- ~ Fs Module
- ~ Advance Fs

- ~ *NPM*
- ~ *Http Server*
- ~ *Events*
- ~ *Streams*
- ~ *Express*
- ~ *Serving Files*
- ~ *Router*
- ~ *Post ,Query & Parameters*
- ~ *Adding routes & Validation*
- ~ *Middlewares*
- ~ *Controllers*
- ~ *Serving FTP & Compression*
- ~ *Async Express Route*
- ~ *Save API's from DDoS Attack*
- ~ *Uploading & Downloading*
- ~ *Nodemailer*
- ~ *Error Handling*
- ~ *Embedded Javascript Templates*

#### => DataBase :

- ~ *SQL*
- ~ *SQL Basic Query with Nodejs*
- ~ *Mongo Shell*
- ~ *MongoDB Compass*
- ~ *MongoDB with Express*
- ~ *Mongoose Intro*
- ~ *ToDo API\_01*
- ~ *ToDo API\_02*
- ~ *ToDo API\_03*
- ~ *ToDo API\_04*
- ~ *ToDo API\_05*
- ~ *ToDo API\_06*
- ~ *ToDo API\_07*
- ~ *MongoDB Atlas*
- ~ *Sequelize*
- ~ *Redis*
- ~ *Redis with Nodejs*

#### => Validation :

- ~ *Joi Basics*
- ~ *Joi In-depth*
- ~ *Joi with Express*

#### => JWT :

- ~ *Introduction*
- ~ *Using JWT*
- ~ *Basics*
- ~ *Setup*
- ~ *Database & Schema*
- ~ *Registering Users to Database*
- ~ *Validating Data*
- ~ *Hashing the password*
- ~ *Signin with JWT*
- ~ *Login Implementation*
- ~ *Verifying JWT Token*
- ~ *Generating Refresh Tokens*
- ~ *Generating pair of Refresh & Access Token*
- ~ *Saving Refresh Token in Redis*
- ~ *JWT Logging Out User*
- ~ *Implementing Controllers*

#### => TYPESCRIPT :

- ~ *Introduction*
- ~ *Strings, Number & Booleans*
- ~ *Objects, Arrays & Enums*
- ~ *Unions, Literals & Alias*
- ~ *Functions*
- ~ *Configurations*
- ~ *Express App*

#### => Swagger :

- ~ *Introduction*
- ~ *Setting up Swagger in Express*
- ~ *Get the Data*

#### => React :

- ~ *Introduction*
- ~ *Folder Structure*
- ~ *JSX*
- ~ *Expressions and Literals*
- ~ *CSS in React*
- ~ *Nested Components*
- ~ *Greeting App*
- ~ *Props*
- ~ *Conditional Rendering*
- ~ *useState*
- ~ *Arrays & Objects in useState*
- ~ *Forms*
- ~ *ToDo app with useState*
- ~ *useEffect*

- ~ *useRef*
- ~ *React Router Dom*
- ~ *Context*
- ~ *fetch API*
- ~ *Axios*
- ~ *React-Hooks-Forms*
- ~ *Memo*
- ~ *Callback*
- ~ *Sockets*
- ~ *Charts with Sockets*
- ~ *Custom Hooks*
- ~ *React Redux Introduction*
- ~ *Redux*
- ~ *Redux-thunk*
- ~ *Ecommerce with Redux*
- ~ *React Bootstrap Introduction*
- ~ *React Bootstrap*
- ~ *Material-ui Introduction*
- ~ *Material-ui Buttons*
- ~ *Material-ui Slider*
- ~ *Material-ui Typography*
- ~ *Material-ui Forms*
- ~ *Material-ui Grids*
- ~ *Material-ui Cards*

=> GraphQL :

- ~ *Introduction*
- ~ *GraphQL with Node*
- ~ *Apollo Server with GraphQL*
- ~ *Create, Read and Delete in MERNG 1*
- ~ *Create, Read and Delete in MERNG 2*

# Node JS Foundation

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : NODE JS

Course link : <https://ineuron.ai/course/Node-JS-Foundation>

## Course Description :-

The entire course focuses on transforming you into a professional Node developer capable of creating, testing, and deploying real-world production apps. You'll be programming every project from the beginning and working through challenges that I've made to reinforce what you've learned. This will provide you with the practical experience you'll need to design and launch your project once you're finished.

## Course Features :-

- => Online live Classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => NPM
- => Router
- => Middlewares
- => Controllers
- => Serving FTP & Compression
- => Async Express Route
- => Save API's from DDoS Attack
- => Nodemailer
- => Error Handling
- => Embedded Javascript Templates

## Requirements :-

- => No Prior knowledge is required
- => A System with Internet Connection
- => Your dedication

## Instructors :-

- => Syed Ashraf :  
~ Full Stack Engineer at TensorGo Technologies

## Curriculum details :-

- => JAVASCRIPT :
  - ~ JAVASCRIPT Introduction
  - ~ JAVASCRIPT Running Javascript in Browser
  - ~ JAVASCRIPT Console
  - ~ JAVASCRIPT Strings & Numbers
  - ~ JAVASCRIPT var, let & const
  - ~ JAVASCRIPT Data Types
  - ~ JAVASCRIPT Type Conversions
  - ~ JAVASCRIPT Arithmetic Operators
  - ~ JAVASCRIPT Assignment Operator
  - ~ JAVASCRIPT Comparison Operators
  - ~ JAVASCRIPT Logical Not, Or and And
  - ~ JAVASCRIPT Swap Numbers
  - ~ JAVASCRIPT String Handling
  - ~ JAVASCRIPT String Searching
  - ~ JAVASCRIPT Arrays
  - ~ JAVASCRIPT Objects
  - ~ JAVASCRIPT Dates
  - ~ JAVASCRIPT Maths
  - ~ JAVASCRIPT If & Else

- ~ JAVASCRIPT Challenge - If & Else
- ~ JAVASCRIPT Switch Case
- ~ JAVASCRIPT Challenge - Switch Case
- ~ JAVASCRIPT JS Loops
- ~ JAVASCRIPT For Loops
- ~ JAVASCRIPT Nested Loops
- ~ JAVASCRIPT Break & Continue
- ~ JAVASCRIPT Arrays, Strings & Objects
- ~ JAVASCRIPT For-in
- ~ JAVASCRIPT For-of
- ~ JAVASCRIPT While Loops
- ~ JAVASCRIPT Do while Loops
- ~ JAVASCRIPT Loops Exercises
- ~ JAVASCRIPT Functions
- ~ JAVASCRIPT Variable Scopes in Function
- ~ JAVASCRIPT Nested Functions
- ~ JAVASCRIPT Parameters & Arguments
- ~ JAVASCRIPT How function is useful
- ~ JAVASCRIPT Return in Function
- ~ JAVASCRIPT Anonymous Functions
- ~ JAVASCRIPT Calculator
- ~ JAVASCRIPT Arrow Functions

=> NODEJS :

- ~ NODEJS Introduction & Installation
- ~ NODEJS Global Objects
- ~ NODEJS Modules
- ~ NODEJS OS Module
- ~ NODEJS Path Module
- ~ NODEJS Fs Module
- ~ NODEJS Advance FS
- ~ NODEJS npm
- ~ NODEJS Http Server
- ~ NODEJS CRUD API
- ~ NODEJS Events
- ~ NODEJS Streams
- ~ NODEJS Weather API
- ~ NODEJS Express Introduction
- ~ NODEJS Serving Files
- ~ NODEJS Router
- ~ NODEJS Post, Query & Params
- ~ NODEJS Validation
- ~ NODEJS Adding routes & Validation
- ~ NODEJS Middlewares
- ~ NODEJS Controllers
- ~ NODEJS Serving FTP & Compression
- ~ NODEJS Async Express Route
- ~ NODEJS Headers & Cookies
- ~ NODEJS Saving API's from DDoS Attack
- ~ NODEJS Uploading & Downloading
- ~ NODEJS Handling Errors
- ~ NODEJS Embedded Javascript Templates
- ~ NODEJS Validation Joi Basics
- ~ NODEJS Validation Joi In-depth
- ~ NODEJS Validation Joi with Express
- ~ NODEJS DATABASE SQL
- ~ NODEJS DATABASE SQL Basic Query with Nodejs
- ~ NODEJS Sequelize
- ~ NODEJS DATABASE MongoDB Compass
- ~ NODEJS DATABASE MongoDB with Express
- ~ NODEJS DATABASE Mongoose Intro
- ~ NODEJS DATABASE ToDo API
- ~ NODEJS DATABASE MongoDB Atlas
- ~ NODEJS DATABASE Sequelize
- ~ NODEJS JWT

# Class 10 Mathematics

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10-Mathematics>

## Course Description :-

Maths is a subject that needs a lot of practice, memorisation of formulas and a clear understanding of concepts. It is a scoring subject for students of Class 10 if they practice it on a regular basis. CBSE syllabus for Class 10 Maths for the academic year 2020-2021 includes topics such as number systems, algebra, coordinate geometry, etc. All the units mentioned in the CBSE 10th Maths syllabus is important because the final question paper is prepared as per the syllabus. It also provides information about marks weightage, practicals, assignments, projects, etc.

## Course Features :-

- => Self paced video session
- => Covered entire class 10th Mathematics syllabus
- => Solved questions chapter wise
- => Notes
- => Previous year solved questions

## What you will learn :-

- => Entire NCERT Class 10th Mathematics Syllabus
- => Chapter wise solution with detailed explanation

## Requirements :-

- => Computer with Internet Connectivity

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => Real Numbers :
  - ~ 1.1 Introduction
  - ~ 1.2 Euclids Division Lemma
  - ~ 1.3 The Fundamental Theorem of Arithmetic
  - ~ 1.4 Revisiting Irrational Numbers
  - ~ 1.5 Revisiting Rational Numbers and Their Decimal Expansions
  - ~ 1.6 Summary
- => Polynomials :
  - ~ 2.1 Introduction
  - ~ 2.2 Geometrical Meaning of the Zeroes of a Polynomial
  - ~ 2.3 Relationship between Zeroes and Coefficients of a Polynomial
  - ~ 2.4 Division Algorithm for Polynomials
  - ~ 2.5 Summary
- => Pair of Linear Equations in Two Variables :
  - ~ 3.1 Introduction
  - ~ 3.2 Pair of Linear Equations in Two Variables
  - ~ 3.3 Graphical Method of Solution of a Pair of Linear Equations
  - ~ 3.4 Algebraic Methods of Solving a Pair of Linear Equations
  - ~ 3.5 Equations Reducible to a Pair of Linear Equations in Two Variables
  - ~ 3.6 Summary
- => Quadratic Equations :
  - ~ 4.1 Introduction
  - ~ 4.2 Quadratic Equations
  - ~ 4.3 Solution of a Quadratic Equation by Factorisation
  - ~ 4.4 Solution of a Quadratic Equation by Completing the Square
  - ~ 4.5 Nature of Roots
  - ~ 4.6 Summary
- => Arithmetic Progressions :
  - ~ 5.1 Introduction
  - ~ 5.2 Arithmetic Progressions
  - ~ 5.3  $n$ th Term of an AP
  - ~ 5.4 Sum of First  $n$  Terms of an AP
  - ~ 5.5 Summary
- => Triangles :
  - ~ 6.1 Introduction
  - ~ 6.2 Similar Figures

- ~ 6.3 Similarity of Triangles
- ~ 6.4 Criteria for Similarity of Triangles
- ~ 6.5 Areas of Similar Triangles
- ~ 6.6 Pythagoras Theorem
- ~ 6.7 Summary

#### => Coordinate Geometry :

- ~ 7.1 Introduction
- ~ 7.2 Distance Formula
- ~ 7.3 Section Formula
- ~ 7.4 Area of a Triangle
- ~ 7.5 Summary

#### => Introduction to Trigonometry :

- ~ 8.1 Introduction
- ~ 8.2 Trigonometric Ratios
- ~ 8.3 Trigonometric Ratios of Some Specific Angles
- ~ 8.4 Trigonometric Ratios of Complementary Angles
- ~ 8.5 Trigonometric Identities
- ~ 8.6 Summary

#### => Some Applications of Trigonometry :

- ~ 9.1 Introduction
- ~ 9.2 Heights and Distances
- ~ 9.3 Summary

#### => Circles :

- ~ 10.1 Introduction
- ~ 10.2 Tangent to a Circle
- ~ 10.3 Number of Tangents from a Point on a Circle
- ~ 10.4 Summary

#### => Constructions :

- ~ 11.1 Introduction
- ~ 11.2 Division of a Line Segment
- ~ 11.3 Construction of Tangents to a Circle
- ~ 11.4 Summary

#### => Areas Related to Circles :

- ~ 12.1 Introduction
- ~ 12.2 Perimeter and Area of a Circle A Review
- ~ 12.3 Areas of Sector and Segment of a Circle
- ~ 12.4 Areas of Combinations of Plane Figures
- ~ 12.5 Summary

#### => Surface Areas and Volumes :

- ~ 13.1 Introduction
- ~ 13.2 Surface Area of a Combination of Solids
- ~ 13.3 Volume of a Combination of Solids
- ~ 13.4 Conversion of Solid from One Shape to Another
- ~ 13.5 Frustum of a Cone
- ~ 13.6 Summary

#### => Statistics :

- ~ 14.1 Introduction
- ~ 14.2 Mean of Grouped Data
- ~ 14.3 Mode of Grouped Data
- ~ 14.4 Median of Grouped Data
- ~ 14.5 Graphical Representation of Cumulative Frequency Distribution
- ~ 14.6 Summary

#### => Probability :

- ~ 15.1 Introduction
- ~ 15.2 Probability A Theoretical Approach
- ~ 15.3 Summary



# MERN Stack Bootcamp

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/MERN-Stack-Bootcamp>

## Course Description :-

Learn complete web stack programming with React and Node by doing it the way a full-stack professional would do it. Learn how to create whole web apps from start to finish with one of the most trending tech stacks available. You can learn how to create data-driven applications, as well as how to test, protect, and deploy your code, in this thorough study path. This course necessitates a basic understanding of HTML, CSS, and JavaScript.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => File structure and response codes
- => Express Server & Database Modelling
- => Preparing Database Models
- => Database Connectivity

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

=> Introduction to Full Stack :

- ~ Getting started as full stack developer
- ~ Installing node and mongo
- ~ File structure and response codes
- ~ Let me help you to think about a project
- ~ Designing the architecture of our project

=> Express Server & Database Modelling :

- ~ Getting started with express
- ~ Reading express documentation and assignment
- ~ Why everyone loves nodemon ?

=> Preparing Database Models :

- ~ Installing from exercise files
- ~ What is mongoose ?
- ~ Creating a basic schema for User
- ~ Your first virtual
- ~ What the UUID ?
- ~ Create schema methods
- ~ Writing complete User schema

=> Creating Schema with Mongoose :

- ~ Creating category schema
- ~ Creating product - Tshirt Schema
- ~ Order page is complex

=> Database Connectivity :

- ~ MongoDB connection and other DB talk
- ~ Environment variables in process

=> Middleware and Initial Routes :

- ~ What is a middleware ?

- ~ Most common middlewares
- ~ Router in express
- ~ How to use controllers ?
- ~ What is RoboMongo ?
- ~ Use Postman for post routes request
- ~ Fixing models bug and postman
- ~ Saving a user to database
- ~ What about data validation ?
- ~ Setting up validation in routes
- ~ Setting up validation results with custom messages

#### => Authentication Routes & Tokens :

- ~ What are tokens and how to create them ?
- ~ Creating signin route
- ~ Sending tokens in cookie for user
- ~ Learn to do testing with Postman
- ~ What is Bearer and signout route ?
- ~ How to protect a route ?
- ~ How to write custom middleware ?
- ~ Where we are so far

#### => Parameter & User Controllers :

- ~ What are PARAMS
- ~ Get ID from param
- ~ Handling user route
- ~ Fixing a major bug
- ~ An assignment for USERS
- ~ Get all users assignment solution
- ~ Learn to update the user info
- ~ Testing the PUT route for user
- ~ Using populate from other collection
- ~ One more middleware to update purchases

#### => Summer & Winter Collection - Category :

- ~ Moving further for collections
- ~ Get category from param
- ~ Creating category and saving it
- ~ Get all categories at once
- ~ Update the collection
- ~ Perform a delete operation
- ~ Testing all routes with postman

#### => Adding Tshirt to our backend :

- ~ JSON vs FORM DATA
- ~ Get product by ID
- ~ Saving photo in mongo and tshirt assets
- ~ Add restriction on product fields
- ~ Long debugging at postman
- ~ Get compressed image assets here
- ~ Some optimization for binary data
- ~ Update and delete of Tshirts
- ~ Get all products
- ~ Update your inventory
- ~ This is an alternative to get all distinct categories

#### => Placing the Order :

- ~ Getting started with order
- ~ Get Order by ID
- ~ create an order
- ~ Why ENUMS are important
- ~ Update the order status

#### => Moving to Front End :

- ~ Bootstrap crash course
- ~ React Router crash course

#### => Connecting to Frontend to Backend :

- ~ Create a react application
- ~ Prepare the folder structure
- ~ Remove all errors first
- ~ How to add CSS in react app
- ~ Inject backend into frontend env
- ~ Take full advantage of base component

#### => Navigation & Authentication :

- ~ Inject the navigation bar
- ~ Adding styling to navbar
- ~ Routes for signin and signup
- ~ Signup and Signin form
- ~ Authentication helper for react
- ~ Signup component talks to backend
- ~ Testing the signup route
- ~ Sign in logic
- ~ Sign in testing part 1
- ~ Sign in testing part 2

#### => Restricted Routes & Profile :

- ~ Conditional rendering for signout
- ~ Restriction on Routes
- ~ Admin and user routes in front end
- ~ Protect the admin dashboard

- ~ Admin left side
- ~ Admin Right side

=> Design an Admin Panel :

- ~ Add category routes
- ~ Making category request to backend
- ~ finish creating categories
- ~ CRUD calls for product
- ~ Assignment and add product front end
- ~ UseEffect and preload data
- ~ Fixing categories bug
- ~ Create a product and assignment
- ~ Delete a product from admin
- ~ update the Tshirt part 1
- ~ update the tshirt part 2
- ~ SMALL HELP IN ASSIGNMENT

=> Purchase Card & Cart :

- ~ Creating a reusable card
- ~ Conditional rendering of card elements
- ~ Bring everything to home page
- ~ Home page loads all tshirts
- ~ Add tshirts to cart
- ~ UI for Cart page
- ~ Design a force component reload method
- ~ Order helpers

=> Integrating Stripe - Track1 - Optional :

- ~ Get the final amount for user
- ~ Get a stripe in front end
- ~ Writing backend for stripe payment
- ~ Writing front end for stripe
- ~ Add key and payment gateway is all set

=> Integrating Paypal with card :

- ~ Read the payment gateway docs
- ~ Setup backend from docs
- ~ Some bug fixes and payment helpers
- ~ we are finally processing all payments

=> Empty card bugs & Optional Orders :

- ~ Bug hunting and placing orders

# OpenCV

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/OpenCV>

## Course Description :-

OpenCV (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. This course will guide you through your first steps in studying computer vision and artificial intelligence (AI) using OpenCV. You'll learn about Image & Video Manipulation, Image Enhancement, Filtering, Edge Detection, Object Detection, and Tracking, among other topics.

## Course Features :-

- => Source Code
- => Downloadable resources
- => Assignments
- => Quizzes
- => Completion certificate
- => Detailed discussion on every topic

## What you will learn :-

- => Basics to advance level of OpenCV
- => Image annotation
- => Mouse click events
- => Image processing
- => Feature matching
- => Corner detection and many more

## Requirements :-

- => No prior knowledge in OpenCV
- => Basic knowledge in Python programming
- => A system with a decent internet connection
- => Dedication

## Instructors :-

=> Ashish Kushwaha :

~ Worked in various Machine Learning, Deep Learning, Data Science and Image Processing projects. he has expertise in Python Programming. Currently he is working as a freelancer & tutor & teaching many students from different regions across the globe.

## Curriculum details :-

=> OpenCV basics :

- ~ Installation of OpenCV Preview
- ~ Read and display images
- ~ Pycharm IDE installation
- ~ Read the live video feed from webcam and display
- ~ Saving an image file
- ~ Saving a video file
- ~ Image resizing and rescaling

=> Image annotation :

- ~ Drawing a line on the image Preview
- ~ Drawing a circle on the image
- ~ Draw geometric shapes on images
- ~ Write text on image
- ~ Display the FPS on image

=> Mouse click events :

- ~ What is mouse click events?
- ~ How to use mouse click events?
- ~ Getting the coordinates of the mouse click events
- ~ Use mouse as a paint brush
- ~ Using mouse to change the colors

=> Image processing :

- ~ Changing color spaces (BGR2RGB, Grey scale, HSV and etc.)
- ~ Geometric transformation of the images
- ~ Scaling
- ~ Translation

- ~ *Warning*
- ~ *Rotation*
- ~ *Affine transformation*
- ~ *Perspective transformation*
- ~ *Image threshold*
- ~ *Smoothing images*
- ~ *Image gradients*
- ~ *Canny edge detection*
- ~ *Contours in OpenCv*
- ~ *Histograms*
- ~ *Template matching*
- ~ *Hough line transform*
- ~ *Hough circle transform*
- ~ *Cascades*
- ~ *Image segmentation with Watershed algorithm*

=> Advanced OpenCv :

- ~ *Corner detection*
- ~ *SIFT, SURF, FAST, BRIEF, ORB*
- ~ *Feature matching*
- ~ *Feature Matching + homograph*
- ~ *Image denoising*
- ~ *Image inpainting*

# Pro Operating Systems

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Operating-Systems>

## Course Description :-

This course is designed mostly for computer science subject OPERATING SYSTEM test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> OS Theoretical Test

=> OS Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Operating System Test :

- ~ Operating System Test 1
- ~ Operating System Test 2
- ~ Operating System Test 3
- ~ Operating System Test 4
- ~ Operating System Test 5
- ~ Operating System Test 6
- ~ Operating System Test 7
- ~ Operating System Test 8
- ~ Operating System Test 9
- ~ Operating System Test 10
- ~ Operating System Test 11
- ~ Operating System Test 12
- ~ Operating System Test 13
- ~ Operating System Test 14

# Google Dialogflow

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Google-Dialogflow>

## Course Description :-

Wants to become an expert in Dialogflow? You've come to the right spot. This course is specially designed for developing learning and comprehending Dialogflow from the ground up. This course doesn't require any prior knowledge of Chatbot. It will give you a complete knowledge on Dialogflow in-depth.

## Course Features :-

- => Completion Certificate
- => Source code
- => Challenges
- => Assignments
- => Quizzes
- => Downloadable Resources

## What you will learn :-

- => Create and deploy a conversational chatbot from scratch
- => Deploy Chatbot to cloud platforms
- => Integrate with third-party apps like Facebook, Slack, Skype, Whatsapp

## Requirements :-

- => No Prior experience of any Chatbot is required.
- => Slack, Facebook, Telegram & more Accounts
- => A system with internet connection
- => Your Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Introduction :

- ~ What is Chatbot? Preview
- ~ Why Chatbot?
- ~ Types of Chatbot
- ~ Use of Chatbot
- ~ Examples of Chatbot
- ~ Chatbot Architecture
- ~ What is Google Dialogflow?

=> Create Account :

- ~ Google Account Preview
- ~ Dialogue Console quick review

=> Dialogflow Concepts :

- ~ Dialogflow - Agents
- ~ Dialogflow - Create and manage agents
- ~ Dialogflow - Prebuilt Agents
- ~ Dialogflow - Multilingual agents
- ~ Dialogflow - Mega agents
- ~ Dialogflow - Intents
- ~ Dialogflow - Create and manage intents
- ~ Dialogflow - Training Phrases
- ~ Dialogflow - Actions and parameters
- ~ Dialogflow - Responses
- ~ Dialogflow - Rich response messages
- ~ Dialogflow - Default intents
- ~ Dialogflow - Entities
- ~ Dialogflow - Entity options
- ~ Dialogflow - System entities
- ~ Dialogflow - Custom entities
- ~ Dialogflow - Contexts
- ~ Dialogflow - Input and Output contexts
- ~ Dialogflow - Follow-up intents

- ~ *Dialogflow - Follow-up intents creation*
- ~ *Dialogflow - Events*
- ~ *Dialogflow - Fulfillment*
- ~ *Dialogflow - Inline editor*
- ~ *Dialogflow - Webhook service*

=> Building Chatbot :

- ~ *Overview*
- ~ *Create Agent in Dialogflow*
- ~ *Create Intent and Entities*
- ~ *Food Order Intent*
- ~ *Why is integration required?*
- ~ *Telegram Integration*
- ~ *Facebook Integration*
- ~ *Facebook Integration Test*
- ~ *Slack Integration*

=> COVID19 Chatbot :

- ~ *Overview*
- ~ *Agent & Intent Creation*
- ~ *World Stats Info Intent*
- ~ *Webhook Code for Welcome Intent*
- ~ *Get Stats Covid Code*
- ~ *World Covid Code*
- ~ *Deployment*
- ~ *Enable Webhook*

=> Course Summary :

- ~ *Summary*



# Apache Druid

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Apache-Druid>

## Course Description :-

Apache Druid is a real-time database to power modern analytics applications. It is a modern technology which has made itself indispensable in the world of big data and data analytics. It comes up with cutting-edge features like easy integration in data pipelines, fast consistent queries, high concurrency support, support in both on-prem and cloud infrastructure, and many more.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduces Apache Druid
- => Apache Druid Installation
- => Walkthrough of Apache Druid Console
- => Data Transformation while ingestion
- => Data filtering while ingestion
- => Nested data parsing while ingestion
- => Data Rollup while ingestion
- => Introduction to Data Segments
- => Data Deletion in Druid
- => Other ways of loading data into Druid
- => Querying data in Druid
- => Druid Internals
- => Druid integration with Superset
- => Druid at Netflix

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shruti Mantri :

~ Shruti Mantri is a well-known software architect, instructor and mentor in the industry. She has 10+ years of experience in the software industry, and has worked with different organizations like Oracle, Flipkart, Amazon, Myntra and Twitter. She is known for her expertise in the data engineering field, and has a sound knowledge on the latest technologies in this domain. She has helped develop data platform at organizations, and guided several mentees in understanding data engineering and how to get better at it.

## Curriculum details :-

- => Introduces Apache Druid :
  - ~ An Introduction to Apache Druid
  - ~ Course Objectives
  - ~ Where to use & where not to use Druid
- => Apache Druid Installation :
  - ~ Single Server Installation
  - ~ Overview of the Druid Console
- => Understanding Data Load :
  - ~ Load data from file using console
  - ~ Other ways to load data from file
  - ~ Transformations over data
  - ~ Applying filters over data

- ~ *Parsing nested data*
- ~ *Rollup Data*

=> Other Data Loads :

- ~ *Load data from Kafka*
- ~ *Load data from Https*
- ~ *Load ORC data format*

=> Querying :

- ~ *Druid SQL Queries*
- ~ *Native Queries*
- ~ *Querying Data via Http*

=> Druid Internals :

- ~ *Druid Architecture*

=> Use Cases :

- ~ *Integrating Superset with Druid*
- ~ *Industry Use-case: Druid at Netflix*

# Data Science Interview Tech Neuron

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/Data-Science-Interview-Tech-Neuron>

## Course Description :-

This course is designed for an individual trying to transition towards various data science careers in the industry. Keeping all the hurdles in mind that we generally face during your transition so that your journey will be smooth and without losing any opportunity, you will be able to transition in the industry. Discuss, Collaborate, Participate and Win the Race.

## Course Features :-

- => Online Instructor-led learning
- => Meet with Achiever
- => Proper Roadmap
- => One-One Resume Building
- => Lifetime Dashboard access
- => Doubt clearing
- => Quiz in every module
- => Career Counselling
- => Assessments
- => Mock Interview
- => Certificate
- => 850 + interview question live discussion
- => AI leader talk(Panasonic, EY, Verizon, Apple, and many)

## What you will learn :-

- => Profile Building
- => System Designing
- => Domain Understanding
- => Common Mistakes
- => Project Management
- => ML Interview Questions
- => DL Interview Questions
- => NLP Interview Questions
- => Stats Interview Questions
- => Python Interview Questions
- => Computer Vision Interview Questions
- => Mock Interview

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Basic Understanding of Python
- => Basic Understanding of ML
- => Basic Understanding of DL

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Introduction about Data science industry and does and don't in your profile and public profile building with iNeuron team. :

- ~ *Induction & Course Introduction*
- ~ *Impact of Data Science in today's world & Roles in Data Science*

=> Python Interview Questions :

- ~ *50 Interview Questions Day1*
- ~ *50 Interview Questions Day2*

=> Stats Interview Questions :

- ~ *60 Interview Questions Day1*
- ~ *60 Interview Questions Day2*

=> Machine Learning Interview Question and Solution Design :

- ~ *40 Interview Questions Day1*
- ~ *40 Interview Questions Day2*
- ~ *40 Interview Questions Day3*
- ~ *40 Interview Questions Day4*
- ~ *40 Interview Questions Day5*
- ~ *40 Interview Questions Day6*
- ~ *40 Interview Questions Day7*

=> Deep Learning & Computer Vision Interview Questions :

- ~ *40 Interview Questions Day1*
- ~ *40 Interview Questions Day2*
- ~ *40 Interview Questions Day3*
- ~ *40 Interview Questions Day4*
- ~ *40 Interview Questions Day5*

=> NLP Interview Questions :

- ~ *40 Interview Questions Day1*
- ~ *40 Interview Questions Day2*

=> Project deployment & Solution Design Life Cycle Interview Questions :

- ~ *50 Interview Questions Day1*
- ~ *50 Interview Questions Day2*

=> Meet with multiple people who has made recent transition :

- ~ *Ask anything*
- ~ *Get suggestion and roadmap*

=> Generic Project architecture design for interview :

- ~ *How Project start in Industry?*
- ~ *Business Expectation*
- ~ *Data Sharing Agreement*
- ~ *Proof of Concept*
- ~ *Master Data Management*
- ~ *High Level Architecture Design*
- ~ *Low Level Architecture Design*
- ~ *Project Wireframe*
- ~ *Data Accusation*
- ~ *Code Level Architecture*
- ~ *Tech Identification*
- ~ *Team Building*
- ~ *Project Delivery Methodology*
- ~ *Project Timeline Calculation*
- ~ *Infrastructure Setup*
- ~ *Project Cost Estimation*
- ~ *Project Kickoff*

=> Resume Design and projects by iNeuron one to one resume building :

- ~ *Resume Template Selection*
- ~ *Tech Stack Involvement*
- ~ *Project Selection & Alignment as per your Experience*
- ~ *Project Details*
- ~ *Your Involvement in Project*
- ~ *Tech Stack for Project*
- ~ *Fine Tuning of your Resume*
- ~ *Proof Reading*
- ~ *LinkedIn & GitHub Update*
- ~ *Applying for Job*
- ~ *Resume Finalization based on Job Description*
- ~ *One to One Discussion with iNeuron Team*

=> Mock interview with Krish and Sudhanshu one to one live/Offline :

- ~ *Fact Check*
- ~ *All round Interview*
- ~ *Review*
- ~ *Feedback*
- ~ *Suggestions*

=> Interaction with many achievers who has done a recent transition in data science on all level :

- ~ *Interaction with Ineuron Achievers of all Ages*
- ~ *Ask Anything*
- ~ *Expert Advice*
- ~ *Doubt Clarification*

=> Final touch of everything for next journey and launch :

- ~ *Check the JOSH!!!*

# Pro Java Programming Language

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Java-Programming-Language>

## Course Description :-

This course is designed mostly for Java test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Java Theoretical Test

=> Java Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Java Test :

~ *Java Test 1*

~ *Java Test 2*

~ *Java Test 3*

~ *Java Test 4*

~ *Java Test 5*

~ *Java Test 6*

~ *Java Test 7*

~ *Java Test 8*

# Data Structure and Algorithm Foundation

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/Data-Structure-and-Algorithm-Foundation>

## Course Description :-

A computer program is a collection of instructions to perform a specific task. For this, a computer program may need to store data, retrieve data, and perform computations on the data. A data structure is a named location that can be used to store and organize data and an algorithm is a collection of steps to solve a particular problem. Learning data structures and algorithms allow us to write efficient and optimized computer programs. Data Structure is a way of collecting and organizing data in such a way that we can perform operations on these data in an effective way.

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Certificate
- => Assignment
- => Quiz

## What you will learn :-

- => Data structure and algorithm
- => Use of data structure
- => Practical implementation
- => Logical ability

## Requirements :-

- => Computer with Internet Connectivity
- => Basic programming understanding

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving an analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

- => Introduction about Data Structure and Algorithms (Hindi) :
  - ~ Introduction Preview
- => Analysis in Data Structure Algorithms (Hindi)
- => Introduction to DS Algo Analysis Part 2 (Hindi)
- => Asymptotic Notation : Discussion about theta Notation
- => Big O Notation in DS&Algo (Hindi)
- => Omega Notation in DS&ALGO (Hindi)
- => Recurrence Relation Solving : Master's Theorem
- => Recurrence Relation Solving-Substitution method
- => Recursive Tree Method DSA - (Hindi)
- => Introduction to Divide and Conquer DSA - (Hindi)
- => Binary Search Part 1 - Data Structure and Algorithm Hindi
- => Binary Search Part 2 Data Structure and Algorithm - Hindi)
- => Mergesort Part 1 - Data Structure and Algorithm | Hindi
- => Mergesort Part 2 Data Structure and Algorithm (Hindi)
- => Mergesort Part 3 Data Structure and Algorithm (Hindi)
- => Introduction to Quicksort Algorithm | Data Structure and Algorithm | Hindi
- => Implementation of QuickSort | Data Structure and Algorithm (Hindi)
- => Recurrence Relation of Quicksort Algorithm | Data Structure and Algorithm | Hindi
- => Problem1 based on Quicksort | Data Structure and Algorithm | Hindi
- => Problem2 based on Quicksort | Data Structure and Algorithm

=> Selection Procedure Algorithm

=> Recurrence Relation of selection procedure | Data Structure and Algorithm | Hindi

=> Finding of Maxima and Minima Using DAC | Data Structure and Algorithm | Hindi

# Android Chatapp with FireBase

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : ANDROID

Course link : <https://ineuron.ai/course/Android-Chatapp-with-FireBase>

## Course Description :-

If you are an Android developer, but you have not yet created an Android chat app, then you're losing out on the actual excitement of development. Real-time messaging, which is at the heart of every real-time application, is another feature of chat applications. You may have seen and used a variety of different Android chat applications and longed to have your own. Take a closer look at how Firebase is used to develop an excellent full-stack Android chat app. This course will walk you through the whole process of creating a feature-rich Android chat app.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Screen Designing
- => Firebase data authentication
- => Email and password validation
- => Kill switch
- => Chat activity layout
- => Custom adapters for chat applications
- => Firebase event listeners
- => Debugging
- => Handling Gradles
- => Phone verification integration

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Let's write some code directly :

- ~ Get a pen and paper
- ~ Assets and reading materials
- ~ Tricky start and icons
- ~ Login screen design and assignment
- ~ Designing tips for login Activity
- ~ Designing Register Activity
- ~ Adding firebase to our project- auth and database
- ~ Getting Firebase instance and all fields
- ~ Setting ErrorBox for failures
- ~ Saving your usernames

=> Let's nail down the login process :

- ~ Creating a firebase user and errors
- ~ Email and Password Validation - Regex
- ~ Having a Kill switch
- ~ Code before me
- ~ Solving Registration errors at StackOverflow



- ~ Copying code to login from registerActivity
- ~ Finishing up our login process

=> Customized Phone book app :

- ~ Take a break and understand list view customization
- ~ Custom listview and adapters
- ~ Finishing customied phoneBook App

=> Back to code chat app :

- ~ Chat Activity Layout
- ~ Single chat row design
- ~ Custom Adapter for chat
- ~ Child event listner from firebase
- ~ Setting chats in listView- Toughest video so far
- ~ Finishing up Adapter class and memory saving

=> Let's finish this app - A lot of errors are here to face :

- ~ Setting up username for chat in chatActivity
- ~ Pushing chats to firebase
- ~ Falling into problems
- ~ After intense debugging

# Business Analytics Crash Course

---

Topic Name : DATA ANALYTICS

Sub-topic Name : BUSINESS ANALYTICS MASTERS

Course link : <https://ineuron.ai/course/Business-Analytics-Crash-Course>

## Course Description :-

Learn the power of using powerful visualization tools such as PowerBi and Tableau alongside advanced excel coupled with the most important fundamentals of Python

## Course Features :-

- => Business Analytics Certification
- => Online Instructor-led learning: Live teaching by instructors
- => Hands-on project implementation
- => 100+ hours of live interactive classes
- => Every week doubt clearing session after the live classes
- => Lifetime Dashboard access
- => Assignments in all the module
- => Live class recordings and materials
- => Interview Questions

## What you will learn :-

- => Python
- => PowerBI
- => Tableau
- => Advanced Excel
- => Statistics

## Requirements :-

- => Laptop
- => Stable internet connection
- => Your Dedication

## Curriculum details :-

- => Introduction to Analytics
- => Python for Data Analytics :
  - ~ Install setup and overview Preview
  - ~ Ipython/Jupyter Notebook overview Preview
  - ~ Intro to NUMPY
  - ~ Creating Arrays.
  - ~ Using Arrays and Scalar
  - ~ Indexing Arrays
  - ~ Arrays transposition
  - ~ Universal arrays function
  - ~ Arrays processing
  - ~ Array input and output
  - ~ Series
  - ~ DataFrames
  - ~ Index Objects
  - ~ Re-index
  - ~ Drop entry
  - ~ Selecting entries
  - ~ Data alignment
  - ~ Rank and Sort
  - ~ Summary statistics
  - ~ Missing data
  - ~ Index Hierarchy
  - ~ Reading and writing text files
  - ~ JSON with Python
  - ~ HTML with Python
  - ~ Microsoft Excel files with Python
  - ~ Merge
  - ~ Merge on Index
  - ~ Concatenate
  - ~ Combining Data Frames
  - ~ Reshaping
  - ~ Pivoting
  - ~ Duplicates in DataFrames

- ~ Mapping
- ~ Replace
- ~ Rename index
- ~ Binning
- ~ Outliners
- ~ Permutation
- ~ GroupBy on DataFrames
- ~ GroupBy on Dict and Series
- ~ Aggregation
- ~ Splitting, Applying and combining.
- ~ Cross Tabulation
- ~ Installing Seaborn
- ~ Histograms
- ~ Kernel Density estimate plots
- ~ Combining plot styles
- ~ Box and Violin plots
- ~ Regression Plots
- ~ Heat maps and clustered matrices
- ~ Introduction to SQL with Python
- ~ SQL - SELECT, DISTINCT, WHERE, AND & OR
- ~ SQL WILDCARDS, ORDER BY, GROUP BY, and Aggregate Functions

#### => SQL FOR DATA ANALYTICS :

- ~ Introduction.
- ~ ER Diagram.
- ~ Schema Design.
- ~ Normalization.
- ~ SQL SELECT statement.
- ~ SQL SELECT using common functions.
- ~ SQL JOIN overview.
- ~ INNER JOIN.
- ~ LEFT JOIN.
- ~ RIGHT JOIN.
- ~ FULL JOIN.
- ~ SQL best practice.
- ~ INNER JOIN Advanced.
- ~ INNER JOIN and LEFT JOIN combo.
- ~ SELF JOIN.
- ~ JOINS and AGGREGATION Subqueries.
- ~ Sorting.
- ~ Independent Subqueries.
- ~ Co related Subqueries.
- ~ Analytic function.
- ~ Set operations.
- ~ SQL views.
- ~ Create a view.
- ~ Create a view using DDL.
- ~ SQL insert Advanced Technique.
- ~ Insert to create table.
- ~ INSERT to new data on existing table 1.
- ~ INSERT to new data on existing table 2.
- ~ INSERT to new data on existing table 3
- ~ INSERT to new data on existing table 4.
- ~ SQL update Advance technique and TCL.
- ~ SQL delete and TCL.
- ~ SQL constraints.
- ~ SQL aggregations.
- ~ SQL programmability.
- ~ SQL query performance.
- ~ SQL Extras.

#### => Advance Excel

#### => Data wrangling with Excel :

- ~ Microsoft Excel fundamentals.
- ~ Entering and editing texts and formulae.
- ~ Working with basic Excel functions.
- ~ Modifying an Excel worksheet.
- ~ Formatting data in an excel worksheet.
- ~ Inserting images and shapes into an Excel worksheet.
- ~ Creating Basic charts in Excel.
- ~ Printing an Excel worksheet.
- ~ Working with an Excel template.
- ~ Working with an excel list.
- ~ Excel list function.
- ~ Excel data validation.
- ~ Importing and exporting data.
- ~ Excel pivot tables.
- ~ Working with excels PowerPivot tools.
- ~ Working with large sets of Excel data.
- ~ Conditional function.
- ~ Lookup functions.
- ~ Text based functions.
- ~ Auditing and Excel worksheet.
- ~ Protecting Excel worksheets and workbooks.
- ~ Mastering Excel "What if?" Tools?
- ~ Automating Repetitive Tasks in Excel with Macros.
- ~ Macro Recorder Tool.
- ~ Excel VBA Concepts.

- ~ Advance VBA.
- ~ Preparing and Cleaning Up Data with VBA.
- ~ VBA to Automate Excel Formulas.
- ~ Preparing Weekly Report.
- ~ Working with Excel VBA User Forms.
- ~ Importing Data from Text Files.

#### => Business Statistics :

- ~ Descriptive Analytics.
- ~ Inferential Statistics.
- ~ Hypothesis Test 1 & 2.
- ~ Covariance.
- ~ Correlation.
- ~ Regression.
- ~ Conjoint & Discriminant Analysis.
- ~ Discrete Uniform Distribution.
- ~ Continuous Uniform Distribution.
- ~ Binomial Distribution.
- ~ Poisson Distribution.
- ~ Normal Distribution.
- ~ Sampling Techniques.
- ~ T Distribution.
- ~ Hypothesis Testing and Confidence Intervals.
- ~ Chi Square Test and Distribution.
- ~ Bayes Theorem.

#### => Visual Analyst :

- ~ Talking about Business Intelligence.
- ~ Tools and Methodologies used in BI.
- ~ Why Visualization is getting more popular.
- ~ Why Tableau?
- ~ Gartner Magic Quadrant of Market Leaders.
- ~ Future business impact of BI.
- ~ Let's Explore
- ~ Tableau Products.
- ~ Tableau Architecture.
- ~ BI Project Execution.
- ~ Tableau Installation in local system.
- ~ Introduction to Tableau Prep.
- ~ Tableau Prep Builder User Interface.
- ~ Data Preparation techniques using Tableau Prep Builder tool.
- ~ How to connect Tableau with different data source.
- ~ Visual Segments.
- ~ Visual Analytics in depth.
- ~ Filters, Parameters & Sets.
- ~ Tableau Calculations using functions.
- ~ Tableau Joins.
- ~ Working with multiple data source (Data Blending).
- ~ Building Predictive Models.
- ~ Dynamic Dashboards and Stories.
- ~ Sharing your Reports.
- ~ Tableau Server.
- ~ User Security.
- ~ Scheduling.
- ~ PDF File.
- ~ JSON File.
- ~ Spatial File.
- ~ Statistical File.
- ~ Microsoft SQL Server.
- ~ Salesforce.
- ~ AWS.
- ~ Azure.
- ~ Google Analytics.
- ~ R.
- ~ Python.
- ~ Hadoop.
- ~ OneDrive.
- ~ Microsoft Access.
- ~ SAP HANA.
- ~ SharePoint.
- ~ Snowflake.
- ~ Subject.
- ~ Planning.
- ~ Pen & Paper approach.
- ~ Tools.
- ~ Color theme.
- ~ Shapes.
- ~ Fonts.
- ~ image Selection.
- ~ text position.
- ~ visual placing.
- ~ Story layout & design.
- ~ Dashboard planning.
- ~ Power BI introduction and overview.
- ~ Key Benefits of Power BI.
- ~ Power BI Architecture.
- ~ Power BI Process.
- ~ Components of Power BI.
- ~ Power BI Building Blocks.

- ~ Power BI vs other BI tools.
- ~ Power Installation.
- ~ Overview of Power BI Desktop.
- ~ Data Sources in Power BI Desktop.
- ~ Connecting to a data Sources.
- ~ Query Editor in Power BI.
- ~ Views in Power BI.
- ~ Field Pane.
- ~ Visual Pane.
- ~ Custom Visual Option.
- ~ Filters.
- ~ Introduction to using Excel data in Power BI.
- ~ Exploring live connections to data with Power BI.
- ~ Connecting directly to SQL Azure, HD Spark, SQL Server Analysis Services/ My SQL.
- ~ Introduction to Power BI Development API.
- ~ Import Power View and Power Pivot to Power BI.
- ~ Power BI Publisher for Excel.
- ~ Content packs.
- ~ Introducing Power BI Mobile.
- ~ Power Query Introduction.
- ~ Query Editor Interface.
- ~ Clean and Transform your data with Query Editor.
- ~ Data Type.
- ~ Column Transformations vs Adding Columns.
- ~ Text Transformations.
- ~ Cleaning irregularly formatted data Transpose.
- ~ Date and Time Calculations.
- ~ Advance editor: Use Case.
- ~ Query Level Parameters.
- ~ Combining Data Merging and Appending.
- ~ Data Modelling.
- ~ Calculated Columns.
- ~ Measures/New Quick Measures.
- ~ Calculated Tables.
- ~ Optimizing Data Models.
- ~ Row Context vs Set Context.
- ~ Cross Filter Direction.
- ~ Manage Data Relationship.
- ~ Why is DAX important?
- ~ Advanced calculations using Calculate functions
- ~ DAX queries.
- ~ DAX Parameter Naming.
- ~ Time Intelligence Functions.
- ~ Types of visualization in a Power BI report.
- ~ Custom visualization to a Power BI report.
- ~ Matrixes and tables.
- ~ Getting started with color formatting and axis properties.
- ~ Change how a chart is sorted in a Power BI report.
- ~ Move, resize, and pop out a visualization in a Power BI report.
- ~ Drill down in a visualization in Power BI.
- ~ Drill Through.
- ~ Histograms.
- ~ Basic Area chart.
- ~ Combo Chart in Power BI.
- ~ Customize visualization title, background, and legend.
- ~ Doughnut charts in Power BI.
- ~ Scatter Charts in Power BI.
- ~ Funnel charts in Power BI.
- ~ KPI Visuals.
- ~ Radial Gauge charts in Power BI.
- ~ Bookmarks in Power BI.
- ~ Slicers in Power BI.
- ~ Filters.
- ~ Report Level Parameters.
- ~ Z Order.
- ~ Waterfall charts in Power BI.
- ~ Create a Power BI dashboard.
- ~ Dashboard tiles in Power BI.
- ~ Pin a tile to a Power BI dashboard from a report.
- ~ Pin an entire report page to a Power BI dashboard.
- ~ Data alerts in Power BI service.
- ~ Add an image, text box, video, hyperlink or web code to your dashboard.
- ~ Configuring a Dashboard.
- ~ Power BI Q&A.
- ~ Display a dashboard tile in Focus mode.
- ~ Power BI embedded.
- ~ Row Level Security in Power BI.
- ~ Report Server Basics.
- ~ Refresh a dataset created from a Power BI Desktop file local.
- ~ Refresh a dataset created from a Power BI Desktop file cloud.
- ~ Web Portal.
- ~ Paginated Reports.
- ~ Data Gateways.
- ~ Scheduled Refresh.
- ~ Resources (Rest API/ SOAP APIs/ URL Access).
- ~ R Integration in Power BI Desktop.
- ~ R Powered Custom Visuals.
- ~ Creating R visuals in Power BI.

- ~ *R Visuals in Power BI Service.*
- ~ *R Scripts Security.*
- ~ *Creating visual using Python.*

=> Predictive Analytics :

- ~ *Machine Learning*
- ~ *Deep Learning*

=> Descriptive Analytics :

- ~ *EDA*

### Project details :-

=> Python for Data Analytics :

- ~ *Stock Market Analysis.*
- ~ *House prices : Advanced Regression Techniques.*
- ~ *Election Analysis.*

=> SQL FOR DATA ANALYTICS :

- ~ *Ecommerce analysis Tableau integration.*
- ~ *Sales Data Analysis Tableau integration.*

=> Data wrangling with Excel :

- ~ *E Commerce Customer Analysis.*
- ~ *Project Management Dashboard.*
- ~ *Sales Dashboard.*

# Blockchain Foundations

---

Topic Name : BLOCKCHAIN

Sub-topic Name : BLOCKCHAIN MASTERS

Course link : <https://ineuron.ai/course/Blockchain-Foundations>

## Course Description :-

Presenting the blockchain community session where students will learn the fundamentals of Blockchain Technology along with Solidity programming fundamentals with hands-on practical problems. Learners will learn to build their own cryptocurrency after completion of this community session

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to blockchain
- => Ethereum and Solidity
- => Solidity
- => Create your Cryptocurrency Project

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Sanjeevan Thorat :

~ Data Scientist and Blockchain developer, with experience in developing and managing end to end solutions. I have hands-on experience in Python Programming Language, Machine Learning Deep Learning and Natural language processing. Blockchain development experience in smart contracts, Decentralised Finance applications, DAOs, NFTs and Oracles running on Ethereum and Polygon blockchains. I have worked with various clients for different industry projects in the blockchain space. I specialize in building smart contracts on the Ethereum blockchain along with JavaScript integration for enhancing user experience to generate maximum returns on investment.

## Curriculum details :-

=> Introduction to blockchain :

- ~ What is Blockchain
- ~ History of Blockchain
- ~ Bitcoin Blockchain in depth

=> Ethereum and Solidity :

- ~ History Ethereum Blockchain
- ~ Ethereum Blockchain in depth
- ~ Creating a blockchain from scratch in Javascript

=> Solidity :

- ~ What is Solidity
- ~ Solidity basics
- ~ Smart contract fundamentals
- ~ Payable functions
- ~ Fallback functions
- ~ View functions
- ~ Pure functions
- ~ Function overloading
- ~ Function overriding
- ~ Solidity Events
- ~ Block and Transaction details
- ~ Solidity Inheritance
- ~ Single Inheritance
- ~ Multiple Inheritance
- ~ Heirarchical Inheritance
- ~ Multilevel Inheritance
- ~ Abstract Contracts
- ~ Solidity Interfaces
- ~ Solidity Libraries

=> Project :

~ *Creating a cryptocurrency with ICO in Solidity from scratch*



# C++ Coding Interview Preparation

---

Topic Name : PROGRAMMING

Sub-topic Name : C++

Course link : <https://ineuron.ai/course/C++-Coding-Interview-Preparation>

## Course Description :-

This course is designed mostly for C++ test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> C++ Theoretical Test

=> C++ Practical Test

=> C++ Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> C++ Coding Test :

- ~ C++ Test 1
- ~ C++ Test 2
- ~ C++ Test 3
- ~ C++ Test 4
- ~ C++ Test 5
- ~ C++ Test 6
- ~ C++ Test 7
- ~ C++ Test 8
- ~ C++ Test 9
- ~ C++ Test 10
- ~ C++ Test 11
- ~ C++ Test 12
- ~ C++ Test 13
- ~ C++ Test 14
- ~ C++ Test 15
- ~ C++ Test 16
- ~ C++ Test 17
- ~ C++ Test 18
- ~ C++ Test 19
- ~ C++ Test 20

# Complete VueJS Development

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : VUE JS

Course link : <https://ineuron.ai/course/Complete-VueJS-Development>

## Course Description :-

VueJS is the shooting star in the world of JavaScript frameworks, regardless of whatever measure you choose (Google Trends, Tweets, etc.). This course covers the most recent version of Vue in great depth and from the ground up. In this course, we will go over all of the fundamentals of VueJs. Vue JS and other frontend frameworks are incredibly popular because they provide the same dynamic, fantastic user experience that we have come to expect from mobile applications - but now in the browsers as well. And it is no surprise that positions requiring frontend framework expertise such as VueJS are among the highest-paying in the business!

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => VueJs project structure
- => VueJs data types and methods
- => Passing data to props
- => Adding editable forms in todo
- => Passing methods in vueJs

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting Started With VueJS :

- ~ Introduction to Vue JS
- ~ Important note on Vue docs
- ~ Vue web page via CDN
- ~ Injecting Vue on web page
- ~ Another method to add app

=> Basics of VueJS :

- ~ A nice card in Vue
- ~ Directives in VueJS
- ~ Handling Arrays in VueJS
- ~ loops and assignment in VueJS
- ~ Handling Booleans and conditionals in VueJS
- ~ Login and logout in VueJS
- ~ Why people avoid v-show

=> 2 way binding in VueJS :

- ~ Getting the values from html in VUEJS

=> 3 way binding in VueJS :

- ~ Model the data in VueJS

=> 4 way binding in VueJS :

- ~ Computed and methods in VueJS

=> 5 way binding in VueJS :

- ~ Handling computed in VueJS

=> 6 way binding in VueJS :

- ~ Assignment time in VueJS

=> 7 way binding in VueJS :

- ~ Life Cycle hooks in VueJS

=> Moving to Vue cli :

- ~ Vue cli and GUI

- ~ Redo the project in VueJS

- ~ Setup you HTML for counter app

- ~ Counter app and assignment

=> Conditionals in VueJS :

- ~ bulding logics for Rating app

- ~ Finishing up rating app in VueJS

- ~ Word generator project in VueJS

- ~ Word generator methods

- ~ A nasty bug to find in VueJS

=> Components and third part

libraries :

- ~ Adding third party libraries

- ~ Your first component

- ~ Watcher in VueJS

- ~ craft a winning login in tictacToe VueJS

- ~ Making our game functional in VueJS

- ~ Reload the game in Vue JS

=> Handling local storage in

VueJS :

- ~ Building a local storage app in VueJS

- ~ Bring in Moment and UUID

- ~ A reuseable header in Vue JS

- ~ Input form component in VueJS

- ~ Movie card component in VueJS

- ~ Handling local storage in VueJS

- ~ Bring all components together and bug assignment VueJS

- ~ LifeCycle events in action VueJS

=> Handling API in VueJS :

- ~ Introducing the API in VueJS

- ~ Setting up API project in VueJS

- ~ Axios to fire request on web VueJS

- ~ Handling response with check Vuejs

- ~ Testing the response VueJS

- ~ Summing up user card Vue JS

=> Routing and state management :

- ~ A new router app in vuejs

- ~ Basics of routing middleware

- ~ router link in vue js

- ~ All about routing in Vuejs

- ~ Getting started with Github app in vuejs

- ~ Firebase config settings in vue

- ~ Creating lots of files for vue git project

- ~ Store in vuejs

- ~ Signup gitapp in vuejs

- ~ map getters in vuex

- ~ map actions in vuex

- ~ handling user card in vuex

- ~ preparing repo table in vuex

- ~ handling home component with store in vuex

- ~ Auth Guard in vue router

- ~ debugging session

# Advanced Python

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON

Course link : <https://ineuron.ai/course/Advanced-Python>

## Course Description :-

The purpose of this course is to teach students about Python's advanced modules. Using various assigned challenges, we'll go through several advanced python modules in order to create real-time apps. The information in this course should be understood by anybody with a basic familiarity of the Python programming language. Students will receive hands-on practical experience in producing industrial projects after successfully completing the course. You might begin applying for freelancing employment in order to make a fortune.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Python Flow control
- => List
- => Tuple
- => sets
- => Dictionary
- => Python Functions
- => Python Programs

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shubham Sharma :

~ Having 3+ years of DataScience and Web Development expertise, proficient in data modelling, data preprocessing as well as scripting languages Python and PHP. I've also worked as a mentor and a freelancer. Machine Learning and Natural Language Processing (NLP) are two of my areas of expertise.

## Curriculum details :-

- => Introduction to the course :
  - ~ Course Introduction
  - ~ Who is this course for?
  - ~ Course Overview & Course outcome
  - ~ Course Pre-requisite
- => Python Flow Control :
  - ~ What are control statements?
  - ~ Explanation of If-else
  - ~ Explanation of for loop
  - ~ Explanation of While Loop
  - ~ What is Break & Continue Pass
  - ~ Practical: Guess the Number
  - ~ Practical: Write a program to print a Right-Angle Triangle?
  - ~ Practical: Check the Leap Year
  - ~ Practical: Write a Program to Print Pyramid, Left /right triangle/ diamond?
- => Assignment 1 : :
  - ~ How to convert one unit to another?

=> Python Data Structure - List :

- ~ What is a List?
- ~ Creating a List
- ~ Accessing The List Elements
- ~ Adding New Data in the List
- ~ The Slice Operator With List
- ~ Modifying a List
- ~ Deletion in a List
- ~ Appending Items In a List
- ~ Multiplying a List
- ~ Membership Operators On List
- ~ Built In Functions For List
- ~ Methods Of List
- ~ Practical: Python program to interchange first and last elements in a list.
- ~ Practical: Count occurrences of an element in a list

=> Assignment 2: :

- ~ Multiply all numbers in the list

=> Python Data Structure-Tuple :

- ~ What is a Tuple?
- ~ Difference between list and tuple.
- ~ Benefits Of Tuple
- ~ Creating Tuple
- ~ Packing/Unpacking a Tuple
- ~ Accessing a Tuple
- ~ Changing the Tuple
- ~ Deleting the Tuple
- ~ Functions used with Tuple
- ~ Methods used with Tuple
- ~ Operations allowed on Tuple
- ~ Practical: Create a list of tuples from a given list having a number and its cube in each tuple.

=> Assignment 3: :

- ~ Sum of tuple elements

=> Python Sets :

- ~ What are Sets?
- ~ Practical: Iterate over a set in Python

=> Assignment 4: :

- ~ Find lost element from a duplicated array using Set difference

=> Python Dictionary :

- ~ What Is a Dictionary ?
- ~ What Is a Key-Value Pair ?
- ~ Creating a Dictionary
- ~ Important Characteristics of a Dictionary
- ~ Different ways to access a Dictionary
- ~ Updating elements In Dictionary
- ~ Removing elements from Dictionary
- ~ Functions used In Dictionary
- ~ Dictionary Methods
- ~ Removing elements from Dictionary
- ~ Functions used In Dictionary
- ~ Practical: Python program to find the sum of all items in a dictionary

=> Assignment 5: :

- ~ Extract Keys Value, if Key Present in List and Dictionary

=> Python Functions :

- ~ Global & Local Variables
- ~ Function Argument
- ~ Function Recursion
- ~ Lambda Function
- ~ The map( ) Function
- ~ The filter( ) Function
- ~ Using map( ) and filter( ) with Lambda Expressions
- ~ Practical: Creating a calculator

=> Assignment 6: :

- ~ Create a scientific calculator

=> Projects :

- ~ Create a currency convertor using python.
- ~ Create an alarm clock.
- ~ Create a mail sending program.

=> Final Task :

- ~ Build an app that will track the number of calories that you eat every day.

# Data Engineering with Google Cloud

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA ON CLOUD

Course link : <https://ineuron.ai/course/Data-Engineering-with-Google-Cloud>

## Course Description :-

By gathering, transforming, and releasing data, Data Engineers enable data-driven decision making. A Data Engineer should be able to design, implement, operationalize, secure, and monitor data processing systems, with a focus on security and compliance, scalability and efficiency, fidelity and dependability, and flexibility and portability. This course will teach you how to use Cloud Computing to learn the principles of data engineering.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Necessity of Compute Power for ML Workloads
- => Storage, networking & security importance
- => Big Data & Big query tools
- => BigQuery ML
- => BigQuery Keypoints
- => BigQuery Features
- => Bigquery GIS Example
- => AI Platform
- => DataPrep Tool
- => Understanding ML Workflow
- => What is vertex AI
- => Image classification application overview
- => Collection of yoga pose image dataset
- => Build image classification model with vertex AI

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Introduction :
  - ~ Syllabus Overview
- => Big Data & ML Fundamentals :
  - ~ Necessity of Compute Power for ML Workloads
  - ~ Storage, networking & security importance
  - ~ Big Data & Big query tools
- => BigQuery ML :
  - ~ BigQuery ML
  - ~ BigQuery Keypoints
  - ~ BigQuery Features
  - ~ Bigquery GIS Example

- ~ ML Flow
- ~ ML Process
- ~ BigQueryML create Model
- ~ BigQueryML evaluate model
- ~ BigQueryML predict model

=> PubSub/Dataflow/Pipeline :

- ~ Overview of pipeline
- ~ PubSub Features
- ~ Core concepts
- ~ Hands-on send/receive messages with gcloud
- ~ Dataflow

=> ML :

- ~ Overview
- ~ AI Platform
- ~ DataPrep Tool
- ~ Understanding ML Workflow
- ~ What is vertex AI
- ~ Pricing
- ~ Managing ML Datasets with vertex AI
- ~ Managed dataset hands-on
- ~ Build & Train ML model with vertex AI
- ~ Training a model using AutoML Vertex AI
- ~ AutoML training & deployment
- ~ AutoML prediction
- ~ Clean up resources
- ~ Overview of custom model
- ~ Enable API
- ~ Create Notebook instance
- ~ Create container image
- ~ custom ML model training with vertex AI
- ~ Deployment
- ~ Prediction
- ~ Image classification application overview
- ~ Collection of yoga pose image dataset
- ~ Build image classification model with vertex AI
- ~ Deployment & prediction
- ~ Vertex AI SDK
- ~ Vertex AI SDK code walkthrough
- ~ Vertex AI SDK code walkthrough -2
- ~ Vertex AI SDK hands-on-1
- ~ Vertex AI SDK hands-on-2
- ~ Vertex AI SDK hands-on-3
- ~ Vertex AI SDK hands-on-4
- ~ Vertex AI SDK hands-on-5
- ~ Vertex AI SDK hands-on-6
- ~ What is Hyperparameter tuning
- ~ Hyperparameter tuning
- ~ Hyperparameter tuning mechanism
- ~ Hyperparameter tuning hands-on-1
- ~ Hyperparameter tuning hands-on-2
- ~ Hyperparameter tuning hands-on-3
- ~ Hyperparameter tuning hands-on-4

=> Practical :

- ~ Creating data pipeline using dataprep and bigquery
- ~ Run data transformation pipeline
- ~ Data exploration of ecommerce dataset using Bigquery
- ~ Prediction using BQML

# NLP Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP PROJECT

Course link : <https://ineuron.ai/course/NLP-Projects>

## Course Description :-

Natural Language Processing (NLP) is a technique for deciphering and manipulating human language using algorithms. One of the most widely used fields of machine learning is this technique. You will gain knowledge of various NLP frameworks like SpaCy, NLTK, Torchtext, TensorFlow, Keras, hugging face, etc., and their project implementation. Study transfer learning in NLP - BERT family and many more.

## Course Features :-

- => Quizzes
- => Assignments
- => Hands-on practicals
- => Downloadable resources
- => Completion certificate
- => Hands-on practicals

## What you will learn :-

- => Creating rest API's
- => Project building
- => Real-time projects
- => Application workflow

## Requirements :-

- => Prior understanding of NLP
- => A system with internet connection
- => Your dedication

## Instructors :-

- => Sourangshu Pal :
  - ~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

- => Introduction to course :
  - ~ Introduction to the course Preview
  - ~ Course curriculum
  - ~ Installing software and applications
  - ~ Working with Anaconda environments
  - ~ Pycharm introduction
  - ~ Pycharm with conda
  - ~ Pycharm with venv
  - ~ Pycharm with pipenv
- => Covering Python basics :
  - ~ Building a calculator Preview
  - ~ Working with command-line arguments
  - ~ Building the Flask application
  - ~ Testing our app in Postman
  - ~ Learn to debug with Pycharm
  - ~ Adding a UI to our web app
- => Text-to-speech :
  - ~ Introduction
  - ~ Project setup text-to-speech
  - ~ Project demo
  - ~ Code explanation
  - ~ Project workflow
  - ~ Prediction with Postman
  - ~ Debugging application
- => Speech-to-text :
  - ~ Introduction
  - ~ Project setup speech-to-text
  - ~ Project demo



- ~ *Code explanation*
- ~ *Project workflow*
- ~ *Prediction with Postman*
- ~ *Debugging application*

=> Spell Corrector :

- ~ *Introduction*
- ~ *Project Setup Spell Corrector*
- ~ *Project demo*
- ~ *Code explanation*
- ~ *Project workflow*
- ~ *Prediction with Postman*
- ~ *Debugging application*

=> Named Entity Recognition :

- ~ *NER using BERT*

=> Machine Translation & Keyword Spotting :

- ~ *Machine translation*
- ~ *Keyword spotting*

=> Keyword Extracter & Summarization :

- ~ *Keyword extraction*
- ~ *Extractive text summarization*

=> Brand Measures :

- ~ *Brand measures project*

=> Paraphrasing :

- ~ *Rephrase project*

# Angular JS Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : ANGULAR JS

Course link : <https://ineuron.ai/course/Angular-JS-Course>

## Course Description :-

This course will help you to learn the fundamentals and the practical implementations of Angular JS.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Getting started with angular
- => TypeScript that you need for Angular
- => Let's Build counter
- => Core foundation of angular apps
- => Generator - user input
- => Game- passing info to parent
- => RxJs fundamentals
- => Project:- Form service and pipe in angular project
- => Project :- SignUp reactive form in Angular
- => Project:- Web request and API in Angular
- => Project:- Fire base login and github searcher

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with angular :

- ~ Angular Section 1 Intro
- ~ Tools and installation for Angular
- ~ NG command line utility
- ~ Create your first angular application
- ~ Angular file structure
- ~ Official Hello to Angular app
- ~ Online editors -A word of caution

=> TypeScript that you need for Angular :

- ~ Angular section 2 intro
- ~ Types in TypeScript
- ~ Conditionals in TypeScript
- ~ Looping through array
- ~ Functions and Parameters in TypeScript
- ~ Interface in TypeScript
- ~ Class constructor and Interface
- ~ Decorators- Best explanation you will see

=> Let's Build counter :

- ~ Angular section 3 intro
- ~ Counter app assets and naming
- ~ Creating HTML interface for the counter app
- ~ Handling counter logic
- ~ Injecting class into template

~ Event binding and interpolation

=> Core foundation of angular apps :

- ~ Angular section 4 intro
- ~ Delete everything in project
- ~ What is main ts file
- ~ Creating a module in Angular
- ~ Inject decorator in Module
- ~ Inject decorator in Component
- ~ Polyfills and running the app
- ~ File separation for component
- ~ Injecting second component

=> Generator - user input :

- ~ Angular section 5 intro
- ~ Basics about user input and routing
- ~ Load CSS, Bootstrap and Custom assets
- ~ Logic part of word application
- ~ Finish word generator and assignment

=> Game- passing info to parent :

- ~ Angular section 6 intro
- ~ Prototype and reading docs
- ~ Creating a mistake and custom CSS
- ~ Winning logic and package config
- ~ Input decorator in Angular
- ~ Angular switch and case
- ~ Winning and reset logic in Angular
- ~ Detect clicks and custom messages in Angular
- ~ Reset the game
- ~ Ng For loop and property binding in Angular
- ~ Small CSS fix - optional

=> RxJs fundamentals :

- ~ Angular section 7 intro
- ~ Problem that RxJs is trying to solve
- ~ Comparing regular Js and RxJs
- ~ Understand the flow in RxJs
- ~ What is observable in RxJs
- ~ What are observers in RxJs
- ~ Subscribe and Unsubscribe to events
- ~ Pipe and operators in RxJs

=> Project:- Form service and pipe in angular project :

- ~ Angular section 8 intro
- ~ Building todo - form and service in angular
- ~ Creating app structure for todo in Angular
- ~ Creating model for todo in Angular
- ~ Angular pipe in Action
- ~ Service - Business logic of Angular app
- ~ Life cycle hooks in Angular
- ~ NgClass and NgFor todo Angular
- ~ Reactive form and template driven form
- ~ Adding form in module Angular
- ~ 2 way binding syntax in Angular
- ~ Angular wrapper elements

=> Project :- SignUp reactive form in Angular :

- ~ Angular section 9 intro
- ~ Building a signup form and validations
- ~ Adding reactive form and bootstrap
- ~ Brain part of reactive form in Angular
- ~ Building custom form validators
- ~ Understand the basic signup form template
- ~ Connect form with validators
- ~ Render error messages to users

=> Project:- Web request and API in Angular :

- ~ Angular section 10 Intro
- ~ HTTP module in Angular
- ~ Generating components and services for users
- ~ Adding HttpClientModule to app
- ~ Injecting HttpClient in Angular
- ~ Make a web request in Angular
- ~ Accept data from parent as user
- ~ Getting API response and displaying it
- ~ One more thing about ngContent

=> Project:- Fire base login and github searcher :

- ~ Angular section 11 intro
- ~ reading routing docs and layout in Angular
- ~ Understand the project structure
- ~ Reading fire base docs
- ~ Creating a new firebase project
- ~ Config project to firebase
- ~ Generating file structure for github app
- ~ Bring everything in Module for Angular
- ~ signup and sign in and getUser from firebase
- ~ Working with GitHub service API
- ~ Footer for github Angular
- ~ Fixing bugs and header brain

- ~ Header template with router
- ~ Detect changes in grand child
- ~ User card for git
- ~ Finishing home component
- ~ Page not found
- ~ Signup with firebase for git
- ~ Protecting routes and routing
- ~ Sign In with firebase for git
- ~ Minor debugging and Final github searcher

=> Project :- Social Media and - Insta Inspired :

- ~ Angular section 12 intro
- ~ Social media mockup -intro
- ~ Understand database and storage
- ~ Understand database and architecture
- ~ Installing tools that we need
- ~ Generating all components for travelgram
- ~ Building firebase services for travelgram
- ~ Header and footer of the application
- ~ Signup with DB entry in travelgram
- ~ How to upload images or any resources in database
- ~ Conditional rendering of signup template
- ~ Setting up routing for travelgram
- ~ Router and lots of debugging
- ~ Signs in is easy now
- ~ Add post by user
- ~ Home component with a BUG
- ~ Adding list of users
- ~ Like and dislike the post
- ~ Like and dislike with changes

# Auto Sklearn and Auto TimeSeries

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Auto-Sklearn-and-Auto-TimeSeries>

## Course Description :-

This course will help you to get started with the machine learning libraries auto sklearn and auto timeseries.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Auto Sklearn
- => Auto Timeseries

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Rishav Dash :

~ This is Rishav Dash. I am a Jr. Data Scientist and mentor at INeuron.ai with working experience in computer vision, natural language processing, Machine Learning, and Alops. Hands-on experience leveraging machine learning, deep learning, transfer learning models to challenging real-world problems, and building products to solve peoples problems.

## Curriculum details :-

=> Auto Sklearn :

- ~ What is Auto-Sklearn
- ~ Install and Using Auto-Sklearn
- ~ Auto-Sklearn for Classification
- ~ Auto-Sklearn for Regression
- ~ Advanced Examples
- ~ The END

=> Auto Timeseries :

- ~ What is auto-ts
- ~ Setup and installation auto ts
- ~ Auto-ts implementation
- ~ The END

# DP 900 Microsoft Azure Data Fundamentals

---

Topic Name : CLOUD

Sub-topic Name : AZURE

Course link : <https://ineuron.ai/course/DP-900-Microsoft-Azure-Data-Fundamentals>

## Course Description :-

To get ready for other Azure role-based certifications like Azure Database Administrator Associate or Azure Data Engineer Associate, take the DP-900: Azure Data Fundamentals exam. Candidates who are just starting to work with cloud data are the target audience for this exam.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Data Concepts
- => Azure Synapse and Data Lake
- => Account Storage
- => Power BI
- => Relational Databases
- => MS SQL SERVER
- => Database Security
- => Azure Tables Cosmos DB
- => Hadoop Systems
- => Azure and Databricks
- => ELT and SQL Tools
- => Demos

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Data Concepts :
  - ~ Introduction to DP 900
  - ~ Sample Questions
  - ~ Azure Core Data Related Services
  - ~ Types of Cloud Computing
  - ~ Azure Data Related Roles
  - ~ Database Administrator Common tools
  - ~ Data Engineering Common tools
  - ~ Data Analyst Common tools
  - ~ Data Overview
  - ~ Introduction to Data
  - ~ Data documents
  - ~ Data sets
  - ~ Data Types
  - ~ Schema vs Schemaless
  - ~ Query and Querying
  - ~ Batch vs Stream processing
  - ~ Relational data
  - ~ Relational data Relationships
  - ~ Row store vs Column store

- ~ Database Index
- ~ Data Integrity vs Data Corruption
- ~ Normalized vs Denormalized data
- ~ Pivot table
- ~ Strongly consistent vs Eventually consistent
- ~ Synchronous vs Asynchronous
- ~ Non relational data
- ~ Data source
- ~ Data store
- ~ Database
- ~ Data warehouse
- ~ Data mart
- ~ Data lakes
- ~ Data lakehouse
- ~ Data structures
- ~ Unstructured data
- ~ Semi structured
- ~ Semi structured Data Structures
- ~ Semi structure JSON
- ~ semi structure ORC
- ~ semi structure Parquets
- ~ semi structure AVRO
- ~ Structured Data
- ~ Data mining
- ~ Data mining methods
- ~ Data wrangling
- ~ Data modeling
- ~ ETL vs ELT
- ~ Data analytics
- ~ Key performance indicators KPI
- ~ Data Analytic Techniques
- ~ Microsoft One Drive
- ~ Microsoft Sharepoint

=> Azure Synapse and Data Lake :

- ~ Azure Synapse Analytics
- ~ Synapse SQL and pools
- ~ Synapse Key Features
- ~ Azure Data Lake Gen 2
- ~ Polybase
- ~ Synapse ELT
- ~ Azure Data Lake Analytics

=> Account Storage :

- ~ Azure Blob Storage
- ~ Azure Files

=> Power BI :

- ~ Business Intelligence
- ~ Power BI
- ~ Power BI Visualizations
- ~ Power BI Embedded
- ~ Power BI Interactive Reports
- ~ Power BI Service and Dashboards
- ~ Reports vs Dashboards
- ~ Paginated Reports

=> Relational Databases :

- ~ Structured Query Language
- ~ OLAP vs OLAP
- ~ Open Source Relational Databases
- ~ Read Replicas
- ~ Citus Postgres Hyperscale
- ~ Azure SQL Family
- ~ Elastic pools

=> MS SQL SERVER :

- ~ What is DBMS
- ~ Types of DBMS
- ~ What is SQL
- ~ What is SQL Server
- ~ Features of SQL Server
- ~ sql server installation
- ~ SQL Server Architecture
- ~ SQL Server Command Categories
- ~ SQL Server Data Types
- ~ DDL Commands
- ~ Keys in Database
- ~ Constraints in Database
- ~ DML Commands
- ~ Operators
- ~ Nested Queries
- ~ Joins
- ~ Stored Procedure in sql server part 1
- ~ Stored Procedure in sql server part 2
- ~ Stored Procedure in sql server part 3
- ~ Creating and executing stored procedures with output parameters part 1
- ~ Creating and executing stored procedures with output parameters part 2
- ~ Advantages of stored procedure
- ~ DCL Commands

- ~ *TCL Commands*
- ~ *Exception Handling*
- ~ *Difference between sql and T sql*

=> Database Security :

- ~ *Connectivity Architecture*
- ~ *Database Authentication*
- ~ *Network Connectivity*
- ~ *Azure Defender for SQL*
- ~ *Azure Database Server Firewalls*
- ~ *Always Encrypted*
- ~ *Role Based Access Controls*
- ~ *Transparent Data Encryption*
- ~ *Dynamic Data Masking*
- ~ *Private Links*

=> Azure Tables Cosmos DB :

- ~ *Key Value Store*
- ~ *Document Store*
- ~ *Mongo DB*
- ~ *Graph Database*
- ~ *Apache Tinkerpop and Gremlin*
- ~ *Azure Tables*
- ~ *Azure Cosmos DB*
- ~ *Azure Table Account Storage vs Cosmos DB*

=> Hadoop Systems :

- ~ *Apache Hadoop*
- ~ *Apache Kafka*
- ~ *HDInsights*

=> Azure and Databricks :

- ~ *Spark Basics*
- ~ *Why spark is difficult*
- ~ *Why databricks in cloud*
- ~ *How to save databricks demo cost*

=> ELT and SQL Tools :

- ~ *SQL Server Management Studio*
- ~ *SQL Server Data Tools*
- ~ *Azure Data Studio*
- ~ *Azure Data Factory*
- ~ *SQL Server Integration Services*

=> Demos :

- ~ *Install and Use Power BI*
- ~ *Launch Azure SQL and Use Data Studio part 1*
- ~ *Launch Azure SQL and Use Data Studio part 2*
- ~ *Use Azure SQL as data source in Power BI*
- ~ *Use SSMS to perform a query on Azure SQL*



# Complete Bootstrap - 5 Projects

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : BOOTSTRAP

Course link : <https://ineuron.ai/course/Complete-Bootstrap---5-Projects>

## Course Description :-

This course will take you from having no prior knowledge of Bootstrap to mastering all of the utilities, components, widgets, and grids, as well as designing real-world themes and websites. This project-oriented course does not need prior knowledge of Bootstrap. Upon successful completion of this course, you will be able to build responsive and interactive websites and beautiful static pages using the bootstrap framework. So hurry up and enrol now to start a successful career as a front-end web developer.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Bootstrap Integration and typography
- => Buttons, breakpoints and utilities
- => Team pages
- => Navbars
- => Flexboxes
- => Forms
- => Modals
- => Custom cards

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with bootstrap :

- ~ Introduction to Bootstrap4
- ~ Tools to be used in this course
- ~ File structure for learning
- ~ Emmet quick start part 1
- ~ Emmet quick start part 2

=> Bootstrap integration and typography :

- ~ Bootstrap integration
- ~ Bootstrap typography basics
- ~ Bootstrap typography for testimonials
- ~ Embed responsive YouTube videos

=> Video Landing Page :

- ~ Getting assets and preparing html
- ~ Beautiful landing page
- ~ Customized fonts

=> Buttons Breakpoints and utilities :

- ~ Get started with bootstrap buttons
- ~ Button size and backgrounds
- ~ Border utilities in Bootstrap
- ~ Grid system basics in Bootstrap

- ~ Mobile first concept of bootstrap
- ~ Breakpoints in grid

=> Project team-page :

- ~ Getting assets and basic setup of project
- ~ Logo and display utilities
- ~ Heading section
- ~ Team person one content
- ~ Custom styling for team section
- ~ Some fix and assignments

=> Navbar, flexbox, forms and modals :

- ~ Get started with navs
- ~ Flexbox utilities
- ~ Nav panels and assignment
- ~ Basics of navbars
- ~ Toggles and colors in navbars
- ~ Forms in bootstrap
- ~ input groups in Bootstrap
- ~ Modals in bootstrap

=> Project- App launch website :

- ~ Device mockups
- ~ Getting resources
- ~ Navbar part 1
- ~ center menu of navbar
- ~ Customized navbars
- ~ Login Modal
- ~ Feature section with custom font
- ~ Background svg image
- ~ Device mockups usage
- ~ Subscription form customization
- ~ App store icons
- ~ app store CSS
- ~ Building feature section
- ~ feature column section
- ~ Customized CSS for features
- ~ fixing bugs and gradients
- ~ Just fun - unplanned video

=> Project - Build 4 Custom Cards :

- ~ Introduction to cards
- ~ Introduction to cards part 2
- ~ Downloading project 4 files
- ~ preparing HTML for Card 1
- ~ Card 1 custom CSS part 1
- ~ Card 1 custom CSS part 2 and assignment
- ~ preparing HTML for Card 2
- ~ Card 2 custom CSS
- ~ preparing HTML for Card 3
- ~ Card 3 custom CSS part 1
- ~ Custom CSS for card 3 - part 2
- ~ Custom CSS for card 3 - part 3
- ~ preparing HTML for Card 4
- ~ Custom CSS for card 4

=> Bonus sign-up page :

- ~ Download project 5 files
- ~ Preparing our HTML
- ~ CSS for background image
- ~ Purple Styling of buttons
- ~ Adding colors to buttons
- ~ Fixing custom forms
- ~ Fixing errors and media queries

# ReactJS Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : REACT

Course link : <https://ineuron.ai/course/ReactJS-Crash-Course>

## Course Description :-

This course will help you to grab the fundamentals of ReactJS.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => React templates
- => React states
- => React hooks
- => React context API
- => Firebase integration
- => Redux Apps

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> ReactJS crash course :

~ ReactJS crash course

=> NaN :

- ~ NaN
- ~ NaN
- ~ NaN
- ~ NaN
- ~ NaN
- ~ NaN

# Node JS

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : NODE JS

Course link : <https://ineuron.ai/course/Node-JS>

## Course Description :-

Node.js is an open source, cross-platform runtime environment for developing server-side and networking applications. Node.js applications are written in JavaScript, and can be run within the Node.js runtime on OS X, Microsoft Windows, and Linux. It utilizes an event-driven, non-blocking I/O model that makes it lightweight, efficient and excellent for data-intensive real-time applications that run across shared devices.

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Certificate
- => Assignment
- => Quiz

## What you will learn :-

- => Practical implementation of Node JS in real world
- => End to End concepts understanding

## Requirements :-

- => Computer with Internet connectivity
- => Basic Programming understanding

## Instructors :-

- => Keshav Singh :
- ~

## Curriculum details :-

- => Introduction About Node JS :
  - ~ *Introduction Preview*
- => Internal Of Node JS (Hindi)
- => Why Node JS (Hindi)
- => My First Server in Node JS (Hindi)
- => Serving File based on Url Node JS - (Hindi)
- => Event Emitters in - Node JS | Hindi

# Java Coding Interview Preparation

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Java-Coding-Interview-Preparation>

## Course Description :-

This course is designed mostly for Java test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Java Theoretical Test

=> Java Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Java Test :

- ~ *Java Test 1*
- ~ *Java Test 2*
- ~ *Java Test 3*
- ~ *Java Test 4*
- ~ *Java Test 5*
- ~ *Java Test 6*
- ~ *Java Test 7*
- ~ *Java Test 8*
- ~ *Java Test 9*
- ~ *Java Test 10*
- ~ *Java Test 11*
- ~ *Java Test 12*
- ~ *Java Test 13*
- ~ *Java Test 14*
- ~ *Java Test 15*
- ~ *Java Test 16*
- ~ *Java Test 17*
- ~ *Java Test 18*
- ~ *Java Test 19*
- ~ *Java Test 20*

# Pro Max Interview Preparation Edition 3

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Max-Interview-Preparation-Edition-3>

## Course Description :-

Pro Max Edition 3. These are interview preparation tests with a singular goal, to make sure you get a little better in real-world interviews. Leaderboards are ranked based on 1st attempt.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => Interview Preparation Theoretical Test
- => Interview Preparation Practical Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

- => Interview Preparation Test :
  - ~ Interview Preparation Test 1
  - ~ Interview Preparation Test 2
  - ~ Interview Preparation Test 3
  - ~ Interview Preparation Test 4
  - ~ Interview Preparation Test 5
  - ~ Interview Preparation Test 6

# Class 10th Chemistry

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10th-Chemistry>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the important concepts of Science and their importance in our daily life. Class 10th is crucial and is the foundation for higher education of students. In this, the Chemistry section focuses on concepts like Chemical reactions, Acids, Bases, Salts, Metals, Non-Metals, etc.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Chemical reactions and equations
- => Acids, Bases and Salts
- => Metals and Non-metals
- => Carbon and its compounds
- => Periodic classification of elements

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => Ch1 Chemical reactions and eq :
  - ~ Lecture 1 : chemical reaction and equations introduction Preview
  - ~ Lecture 2 : Type of Changes in matter
  - ~ Lecture 3 : chemical reaction
  - ~ Lecture 4 : Chemical Equations
  - ~ Lecture 5 : Combination & Decomposition Reaction
  - ~ Lecture 6 : Displacement & Double Displacement Reaction
  - ~ Lecture 7 : Other Chemical Reactions
  - ~ Lecture 8 : Oxidation & Reduction
- => Ch2 Acid, base and salt :
  - ~ Lecture 1 : Introduction to acids, bases & salts Preview
  - ~ Lecture 2 : Acids & their Classification
  - ~ Lecture 3 : Preparation & Properties of Acids
  - ~ Lecture 4 : Uses Of Acids
  - ~ Lecture 5 : Bases Classification & Preparation
  - ~ Lecture 6 : Bases Properties & Uses
  - ~ Lecture 7 : Neutralization & Its Uses
  - ~ Lecture 8 : Indicators
  - ~ Lecture 9 : Strength of Acids & Bases
  - ~ Lecture 10 : Salts Introduction & Classification
  - ~ Lecture 11 : Salt Preparation & Properties
  - ~ Lecture 12 : NaCl & Bleaching Powder
  - ~ Lecture 13 : Washing Soda, Baking Soda & Plaster
- => Ch3 Metals and non-metals :
  - ~ Lecture 1 : Introduction & Overview of the chapter\_2
  - ~ Lecture 2 : Metals Occurrence & Physical Properties\_2
  - ~ Lecture 3 : Activity Series of Metals\_2
  - ~ Lecture 4 : Chemical Properties of Metals\_2 Preview
  - ~ Lecture 5 : Uses Of Metals
  - ~ Lecture 6 : Non Metal Occurrence
  - ~ Lecture 7 : Physical Properties of Non Metals
  - ~ Lecture 8 : Chemical Properties of Non Metals
  - ~ Lecture 9 : Comparative Account of Metals & Non Metals
  - ~ Lecture 10 : Interaction in Metals & Non Metals
  - ~ Lecture 11 : Ionic Bond Formation, Nature & Structure
  - ~ Lecture 12 : Covalent Bond Examples & Types
  - ~ Lecture 13 : Characteristics of Covalent Bond & Octate Rule
  - ~ Lecture 14 : Comparative Study Of Covalent & Ionic Bond

~ Lecture 15 : Metallurgy, Alloy & Corrosion

=> Ch4 Carbon and its compound :

~ Lecture 1 Preview

~ Lecture 2

~ Lecture 3

~ Lecture 4

~ Lecture 5

~ Lecture 6

~ Lecture 7

~ Lecture 8

~ Lecture 9

~ Lecture 10

~ Lecture 11

=> Ch5 Classification Elements :

~ Lecture 1 : Periodic Classification of Elements part1

~ Lecture 2 : Periodic Classification of Elements part2



# Fundamentals of Database

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Fundamentals-of-Database>

## Course Description :-

By taking up this course, you will learn the fundamentals of Databases along with SQL concepts such as Datatypes, Operators, Expressions, DDL, DML, TCL, DQL, and many more. Upon successful completion of the course, students will be able to create their Databases and also will be able to manage pre-existing Databases.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction of Database
- => Introduction to DBMS
- => Types of DBMS
- => Object-oriented Database
- => Introduction to SQL
- => Features of SQL
- => Applications of SQL
- => Datatypes in SQL
- => SQL operators
- => SQL Data Definition Language
- => SQL Data Manipulation Language
- => SQL View

## Requirements :-

- => Interest to learn
- => Dedication
- => System with good internet connection

## Curriculum details :-

- => Introduction to Database :
  - ~ Dashboard overview
  - ~ Course overview
  - ~ Who is this course for?
  - ~ Course outcome
  - ~ What is a database?
  - ~ Why you should learn database?
  - ~ History of database
  - ~ Types of database
  - ~ What is DBMS?
  - ~ Types of DBMS
  - ~ What is a hierarchical database?
  - ~ What are network databases?
  - ~ What are relational databases(RDBMS)?
  - ~ What is a object oriented database?
- => Assignment 1 :
  - ~ How would you distinguish RDBMS apart from other databases?
- => SQL Introduction :

- ~ What is SQL?
- ~ History of SQL
- ~ Features of SQL
- ~ Applications of SQL
- ~ Why we should learn SQL?
- ~ Prerequisites to learn SQL
- ~ SQL comments
- ~ What is SQL server languages?
- ~ Types of SQL server languages
- ~ Online SQL Server Editor

=> Assignment 2 :

- ~ What kinds of real-time applications can be created with SQL?

=> SQL Syntax :

- ~ SQL Syntax
- ~ SQL Keywords
- ~ SQL Comments
- ~ SQL Commands
- ~ SQL Statements

=> Assignment 3 :

- ~ Write SQL syntax for WHERE and SELECT clause.

=> SQL Datatypes :

- ~ What are the SQL datatypes?
- ~ Why we use datatypes in SQL?
- ~ String data types
- ~ Numeric data types
- ~ Date and Time data types
- ~ Binary datatypes
- ~ MISC datatypes

=> Assignment 4 :

- ~ Do categorized SQL datatypes with examples for real time applications?

=> SQL Operators :

- ~ What is an SQL operator?
- ~ Why we use SQL operator?
- ~ Types of SQL operator
- ~ SQL Arithmetic operators
- ~ SQL Comparison operators
- ~ SQL Logical operators

=> Assignment 5 :

- ~ Write SQL syntax using arithmetic and logical operators.

=> SQL Expressions :

- ~ What is SQL expression?
- ~ Why we use SQL expression?
- ~ Types of SQL expression
- ~ Boolean expression
- ~ Numeric expression
- ~ Date expression

=> Assignment 6 :

- ~ Write at least three SQL queries with boolean and numeric expressions

=> SQL DDL :

- ~ What is SQL data definition language(DDL)?
- ~ Types of DDL
- ~ CREATE command
- ~ DROP command
- ~ ALTER command
- ~ TRUNCATE command
- ~ COMMENT command
- ~ RENAME command

=> Assignment 7 :

- ~ Write six SQL queries using all DDL commands for student database.

=> SQL DML :

- ~ What is SQL data manipulation language(DML)?
- ~ Types of DML
- ~ INSERT command
- ~ UPDATE command
- ~ DELETE command

=> Assignment 8 :

- ~ Write three SQL queries using all DML commands for student database.

=> SQL DCL :

- ~ What is SQL data control language(DCL)?
- ~ Types of DCL
- ~ GRANT command
- ~ REVOKE command

=> Assignment 9 :

- ~ Write two SQL queries using all DCL commands for student database.

=> SQL DQL :

- ~ What is SQL data query language(DQL)?
- ~ SELECT command

=> Assignment 10 :

~ Write SQL queries using *SELECT* commands for student database.

=> SQL Functions :

~ What is SQL function?

~ Why we use SQL function?

~ Types of SQL functions

=> Assignment 11 :

~ Write SQL queries using *AVG()*, *COUNT()*, *FIRST()*, *LAST()*, *MAX()*, *MIN()*, *SUM()* for using student database.

=> SQL Sub queries :

~ What are the SQL subqueries?

~ Types of SQL Subqueries

~ Subquery with Statements

=> Assignment 12 :

~ Write three SQL queries using subqueries for student database.

=> SQL Clauses :

~ What is SQL clauses?

~ Types of SQL clauses

~ Group by clause

~ Having clause

~ Order by clause

=> Assignment 13 :

~ Write three SQL queries using SQL clauses Group by, Having, Order by for student database.

=> SQL Joins :

~ What is SQL Joins?

~ Importance of SQL Joins

~ Types of SQL Joins

~ Inner join with example

~ Left outer join with example

~ Right outer join with example

~ Full outer join with example

=> Assignment 14 :

~ Perform all join operations with subqueries for student database.

=> Other SQL Operations :

~ Create database

~ Drop database

~ Create database table

~ Drop database table

~ Alter operation

~ SQL Constraints

~ SQL Not Null

~ SQL Primary Key

~ SQL Foreign Key

~ SQL Unique

=> Assignment 15 :

~ Create a family tree database using primary key & foreign key and perform all basic operations.

=> SQL Views :

~ What is SQL Views?

~ Why we use it ?

~ Creating Views

~ Dropping views

=> Assignment 16 :

~ Perform SQL view operations for student database.

=> Project Work :

~ Create an SQL database for your school classes, following things, should be included:

- Classes timings

- Class subjects

- Student attendance

- No. of students with their names.

- faculty name with respected subjects

- Write SQL query for extract all students name with their assign subjects.

- Write SQL query for extract all subjects name with their assign faculties name.

- Find out students attendance with their absent and present days using SQL queries.

# Stats for Data Science

---

Topic Name : DATA SCIENCE

Sub-topic Name : STATS

Course link : <https://ineuron.ai/course/Stats-for-Data-Science>

## Course Description :-

Understand what a Normal Distribution is, Explain the difference between continuous and discrete variables, Understand the Central Limit Theorem, Use the Z-Score and Z-Tables, Understand the difference between a normal distribution and a t-distribution, Create confidence intervals, Understand standard deviations, Understand what a sampling distribution is, Apply Hypothesis Testing for Proportions, Use the t-Score and t-Tables

## Course Features :-

- => Roadmap
- => Challenges
- => Interview questions
- => Resume preparation
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Basic understanding of Statistics
- => Importance of Statistics in the world of Data Science

## Requirements :-

- => No prior understanding of stats
- => A system with internet connection
- => Dedication

## Instructors :-

=> krish naik :

*~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.*

## Curriculum details :-

- => Statistics part-1 :
  - ~ Stats for data science Preview
- => Statistics part-2
- => Statistics part-3
- => Statistics part-4
- => Statistics part-5
- => Statistics part-6

# Android Technical Interview Preparation

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : ANDROID

Course link : <https://ineuron.ai/course/Android-Technical-Interview-Preparation>

## Course Description :-

This course is designed mostly for Android test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Android Theoretical Test

=> Android Practical Test

=> Android Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Android Test :

- ~ Android Test 1
- ~ Android Test 2
- ~ Android Test 3
- ~ Android Test 4
- ~ Android Test 5
- ~ Android Test 6
- ~ Android Test 7
- ~ Android Test 8
- ~ Android Test 9
- ~ Android Test 10
- ~ Android Test 11
- ~ Android Test 12
- ~ Android Test 13
- ~ Android Test 14
- ~ Android Test 15
- ~ Android Test 16
- ~ Android Test 17
- ~ Android Test 18

# Blockchain

---

Topic Name : BLOCKCHAIN

Sub-topic Name : BLOCKCHAIN MASTERS

Course link : <https://ineuron.ai/course/Blockchain>

## Course Description :-

Blockchain course is designed to provide an in depth knowledge on various aspects & concepts of blockchain. A step by step learning will help to focus on each & every parameter of Blockchain. This course will take you into a deep dive into the state of the art blockchain technology and how to create our own mini blockchain using Javascript. Moreover, this is a project-ready course which will help you take whatever you learn and apply it into a real-world portfolio-ready app, which you can showcase to the world.

## Course Features :-

- => Understand the underlying concepts of blockchain
- => Understand the underlying computer science concepts that are required in blockchain
- => Clear out all the concepts about Blockchain
- => Have a firm and good grasp over all the technical and non-technical aspects of blockchain
- => Create your own mini blockchain and deploy that in the real world

## What you will learn :-

- => Origin of Blockchain
- => Review of Blockchain
- => Blockchain Primitives
- => Security and Privacy Mechanisms
- => Ethereum Basics
- => Capstone Project

## Requirements :-

- => A computer/laptop
- => Good internet connection
- => Beginner Level knowledge of JS and NodeJs
- => Will to learn

## Curriculum details :-

- => Origin of Blockchain :
  - ~ *What is Blockchain ? Preview*
  - ~ *What is a distributed network and how can it be a curse and a boon at the same time ? Preview*
  - ~ *History of blockchain network before bitcoin Preview*
- => Review of Blockchain :
  - ~ *What are P2P Networks and Why is Blockchain a Distributed, P2P Network*
  - ~ *Distributed Ledgers*
  - ~ *Blockchain vs Cryptocurrency*
  - ~ *Types of Blockchain networks and different blockchain platforms.*
  - ~ *Mining*
- => Blockchain Primitives :
  - ~ *Cryptographic Hash functions*
  - ~ *Public Key Cryptography*
  - ~ *RSA algorithm*
  - ~ *Merkel Trees*
  - ~ *Nodes*
  - ~ *Blockchain Structure & forks*
- => Security and Privacy Mechanisms :
  - ~ *What is Consensus Mechanism*
  - ~ *Proof-of-Work(PoW) & Proof-of-Stake(PoS)*
  - ~ *Proof-of-capacity (PoC) & Proof-of-Activity(PoA)*
  - ~ *Proof-of-Burn(PoB) & Proof-of-weight (PoW)*
  - ~ *Leased Proof-of-Stake (LPoS) & Delegated Proof-of-Stake (DPoS)*
- => Ethereum Basics :
  - ~ *Accounts and creating our own account.*
  - ~ *Transactions*
  - ~ *Gas And Fees Preview*
  - ~ *EVM*
- => Projects :
  - ~ *Lets create our own mini blockchain from scratch in javascript*
  - ~ *Setting up our project and creating our own mini blockchain*

- ~ *Creating a web app to visualize our blockchain*
- ~ *Deploying the web app on cloud to showcase your work*

# Create A Data Pipeline based on Messaging Using PySpark and Airflow

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA PROJECTS

Course link : <https://ineuron.ai/course/Create-A-Data-Pipeline-based-on-Messaging-Using-PySpark-and-Airflow>

## Course Description :-

In this Project, we will learn how to Build a Big Data pipeline on AWS at scale. You will be using the Covid-19 dataset. This will be streamed in real time from an external API using NiFi. The complex JSON data will be parsed into CSV format using NiFi and the result will be stored in HDFS. Then this data will be sent to Kafka for data processing using PySpark. The processed data will then be consumed from Spark and stored in HDFS. Then a Hive external table is created on top of HDFS. Finally the cleaned, transformed data is stored in the data lake and deployed. Visualization is then done using Tableau and AWS QuickSight.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => Create A Data Pipeline based on Messaging Using PySpark and Airflow
- => Build End to End Datapipeline
- => How to Extract Streaming Data into NFFI
- => Data Encryption
- => Data processing using pyspark
- => Build Dashboards

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> MD Imran :

~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Project :- Create A Data Pipeline based on Messaging Using PySpark and Airflow :

- ~ Introduction of Instructor
- ~ Introduction to Data Pipeline
- ~ What is Data Engineering
- ~ Project Overview
- ~ End Notes
- ~ Problem Description
- ~ Understand the application scope
- ~ Tour to existing solution
- ~ End Notes
- ~ Data Infrastructure: Components used
- ~ Nifi
- ~ Hdfs
- ~ Kafka
- ~ Hive
- ~ Airflow
- ~ Pyspark
- ~ Aws services
- ~ Data Visualization Tools
- ~ End Notes
- ~ Solution Description
- ~ Data Architecture
- ~ Tour to Architecture diagram



- ~ Cost Involved
- ~ End Notes
- ~ system Requirements
- ~ Create EC2 Instance
- ~ SSH into EC2 Instance
- ~ Environment setup with docker
- ~ Copy Important folder from local to ec2 and give required permissions
- ~ To connect to different services locally after port forwarding
- ~ To get into bash shell of different containers
- ~ Data Extraction with Nifi
- ~ Data encryption parsing
- ~ Data sources hdfs kafka
- ~ streaming data from kafka to pyspark
- ~ pyspark streaming output kafka nifi hdfs
- ~ Move Data HDFS to hive Table
- ~ Dataflow Orchestration with Airflow
- ~ Connecting with Data Visualization Tool
- ~ Building Dashboard and Report
- ~ End Notes
- ~ Conclude the project
- ~ Assignments & External Resources

# Linux Live Class

---

Topic Name : DEVOPS

Sub-topic Name : LINUX

Course link : <https://ineuron.ai/course/Linux-Live-Class>

## Course Description :-

This Linux course looks at the tools and techniques that Linux system administrators and end-users use on a daily basis to complete their tasks in a Linux environment.

## Course Features :-

- => Online classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Linux Introduction
- => Setting up Our Linux Space
- => Linux Concepts
- => Package Management
- => Linux Commands
- => Working with Terminal
- => Permissions & Security

## Requirements :-

- => A system with Internet Connection
- => Your dedication

## Instructors :-

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Linux Introduction :

- ~ Introduction to Linux
- ~ What is Linux
- ~ Important Pieces in Linux
- ~ Features of Linux
- ~ Evolution of Linux
- ~ Differences between Windows and Linux

=> Setting up Our Linux Space :

- ~ Downloading Necessary tools
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH Clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting Up Passwordless SSH

=> Linux Concepts :

- ~ What is Kernel
- ~ Types of Kernel
- ~ What is Shell
- ~ Types of Shell
- ~ Distros in Linux
- ~ Linux Boot Process
- ~ File System
- ~ RunLevels in Linux
- ~ Filetypes of Linux

=> Package Management :

- ~ *Package Management*
- ~ *Package Managers & DPKG*
- ~ *Working with APT & APT GET*
- ~ *Apt-get Advanced Part 1*
- ~ *Apt-get Advanced Part 2*

=> Linux Commands :

- ~ *Linux Commands Part1*
- ~ *Linux Commands Part2*
- ~ *Linux Commands Part3*
- ~ *Linux Commands Part4*
- ~ *Cat Command Usages*

=> Working with Terminal :

- ~ *File Archival*
- ~ *File Compression*
- ~ *Files and Patterns Search*
- ~ *Input-Output Redirection*
- ~ *Working with Vi Editor*
- ~ *Advanced Vi Editor Part 1*
- ~ *Advanced Vi Editor Part 2*

=> Permissions & Security :

- ~ *Types of Account in Linux*
- ~ *User Management*
- ~ *Group Management*
- ~ *Files Access Controls*
- ~ *Linux File Permissions*
- ~ *Modifying File Ownership*
- ~ *Sudoers in Linux*
- ~ *Cronjobs*
- ~ *SCP*
- ~ *Special Permissions*
- ~ *System Management*
- ~ *System tools*
- ~ *Hard link and Soft link*
- ~ *Aliasing in Linux*
- ~ *Creating users in Multiple ways*

=> Linux in AWS Cloud- Deploy an App in EC2 :

- ~ *Launching an Ubuntu VM and SSH Setup*
- ~ *Package installation in VM*
- ~ *Running our Calculator App*
- ~ *Gunicorn & Nginx Setup*
- ~ *Creating a Gunicorn Service*
- ~ *Attaching an Elastic IP*
- ~ *Attaching OpenSSL Certificates for HTTPS*

# ML and DL Foundations

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/ML-and-DL-Foundations>

## Course Description :-

This course will help you to grab the foundations of Machine learning and Deep learning.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to Machine Learning
- => Introduction to Machine Learning with Python
- => Preparing Data
- => Training and Testing Data
- => Linear Regression
- => Logistic Regression
- => Support Vector Machine
- => K means Clustering
- => KNN
- => Decision Tree in Machine Learning
- => Activation Functions
- => Optimizers
- => Cost Function and Gradient Descent
- => Artificial Neural Network (ANN)
- => Convolutional Neural Network (CNN)
- => Recurrent Neural Network (RNN) and LSTM

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shridhar Mankar :

~ Myself Shridhar Mankar. Making Things Easy is my Specialty. Did my Mtech from COEP. Founder of 5 Minutes Engineering (329k+ Subscribers) 4crore+ Views. I work as a Influencer Marketing Specialist @ iNeuron

## Curriculum details :-

- => Day1 :
  - ~ Introduction to Machine Learning
- => Day2 :
  - ~ Introduction to Machine Learning with Python
- => Day3 :
  - ~ Preparing Data
- => Day4 :
  - ~ Training and Testing Data
- => Day5 :
  - ~ Linear Regression
- => Day6 :
  - ~ Logistic Regression

=> Day7 :

~ *Support Vector Machine*

=> Day8 :

~ *K means Clustering*

=> Day9 :

~ *KNN*

=> Day10 :

~ *Decision Tree in Machine Learning*

=> Day11 :

~ *Activation Functions*

=> Day12 :

~ *Optimizers*

=> Day13 :

~ *Cost Function and Gradient Descent*

=> Day14 :

~ *Artificial Neural Network (ANN)*

=> Day15 :

~ *Convolutional Neural Network (CNN)*

=> Day16 :

~ *Recurrent Neural Network (RNN) and LSTM*

# PL SQL

---

Topic Name : DATABASE

Sub-topic Name : MYSQL

Course link : <https://ineuron.ai/course/PL-SQL>

## Course Description :-

In this Oracle 11g PL/SQL course you will receive introduction training on PL/SQL database programming language covering syntax, structure and features of the language within the context of database applications and programming.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to PLSQL
- => Features and syntax of PL/SQL
- => Use PL/SQL programming constructs
- => Conditionally control code flow
- => Create overloaded package subprograms
- => Collections
- => Implicit and Explicit Cursors

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn
- => Basic SQL

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => PLSQL :
  - ~ Introduction Preview
  - ~ Installation Preview
  - ~ Basic Syntax part 1
  - ~ Basic Syntax part 2
  - ~ Data Types
  - ~ Variables part 1
  - ~ Variables part 2
  - ~ Literals
  - ~ Operators
  - ~ Conditions
  - ~ Loops
  - ~ Strings
  - ~ Arrays
  - ~ Procedures
  - ~ Functions
  - ~ Cursors Preview
  - ~ Records
  - ~ Exceptions
  - ~ Trigger
  - ~ Packages
  - ~ Collections
  - ~ Transactions
  - ~ Data Types
  - ~ DBMS output

# Azure Certification Road Map

---

Topic Name : CLOUD

Sub-topic Name : AZURE INTERVIEW PREPARATION

Course link : <https://ineuron.ai/course/Azure-Certification-Road-Map>

## Course Description :-

Microsoft introduced a new role-based certification methodology, which emphasises the development of skills and knowledge in line with specific work functions. Involving their IT community and analysing what's needed for certain job roles helped them make this decision. As of September of last year, the new certificates were in beta testing, with full release dates for the tests scheduled for 2019 and 2020.

## Course Features :-

- => Roadmap
- => Challenges
- => Interview Questions
- => Resume Preparation
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => How to design solutions for the Microsoft Azure platform
- => Pass exam & Get Certificates of Microsoft Azure Exams
- => Understand the main concepts of Azure, beyond the ones you normally use
- => Be up-to-date on the latest updates to this ever-changing platform.

## Requirements :-

- => Computer with i3 and above configuration
- => Azure free or paid account
- => Your dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

=> Cloud Computing Concepts :

- ~ Which of the following statements is not true about Infrastructure as a Service (IaaS)? (Select one.) Preview
- ~ Which of the following statements regarding a private cloud delivery model are inaccurate? (Select two.)
- ~ An organization has decided to host its website on Microsoft Azure using WordPress. The CFO would like to know what the best delivery model is for all customers. The CFO wants to be assured the website is publicly accessible. What would you recommend?
- ~ Your company has decided to move its data and resources of an old Microsoft Access database. It would like to use the Microsoft Azure SQL migration wizard to move the records. The database administrator indicates that Microsoft Azure SQL is a Software as a Service (SaaS) delivery offering. Is that statement accurate?
- ~ Your organization, a health care practice, is required by law to maintain patient records for seven years. Recently, the organization invested in an electronic health records (EHR) system. The business has been in practice for 18 years and still maintains 5,000+ previous patient files from the past. By law, all these records must be digitized. What type of cloud solution deployment model should the EHR company suggest the health practice implement?
- ~ You are a member of a large accounting firm that works with large corporations. By law, corporations are required to file quarterly tax reports. Traffic is extremely light to the applications except during specific filing periods, usually one week per quarter. Which of the Azure Cloud Service benefits best reflects the usage behaviour that should be addressed?
- ~ Your organization has recently instituted a 100 % telework policy to reduce expenses. As part of the planning, the IT operations team is looking for ways to utilize as many enterprise vendors pre-built software solutions so that there is no need to install custom applications and maintain a dedicated helpdesk. Which cloud architecture should you include in your suggestion to leadership?
- ~ Select the appropriate architecture to match the applications from the following drop-down menus.
- ~ Is the following statement true or false with regards to a hybrid cloud offering?
- ~ What is the difference between fault tolerance and disaster recovery?
- ~ Indicate if the following statements are true (Yes) or false (No) based on the questions regarding capital expenditure (CapEx) and operational expenditure (OpEx).
- ~ Which of the following service types are best aligned with serverless computing?
- ~ When an administrator shuts off a virtual machine instance, which of the following statements regarding operational costs is accurate?
- ~ In order to assure that an organization has a commitment from its cloud service provider for guaranteed uptime, service reliability, and continuous operations, a service-level agreement is signed to ensure what?
- ~ What is meant by multi-tenancy in describing a public cloud deployment model?

=> Azure Concepts and Architecture Components :

- ~ Which of the following best describes the concept of geography? Preview
- ~ How does a user access the Azure portal?
- ~ Where can a cloud administrator go to find pre-built solutions to expedite virtual machines online?

- ~ Which of the following best describes a deployment and management service allowing Azure cloud administrators to create, update, and delete resources in a provisioned account?
- ~ Answer the following question by selecting Yes or No to these three questions.
- ~ How long does a user have access to free Azure account features before one must pay for services under a Pay-As-You-Go plan?
- ~ A company requires 24/7 support for their custom applications running on Microsoft Azure. Besides, they may want to speak to an architect by phone or Microsoft Teams to review their new Platform as a Service deployment. Which service plan must the company purchase to retain these services?
- ~ A customer with a BASIC account can still submit a support ticket for an Azure Cloud issue?
- ~ Answer the following question by selecting Yes or No to these three questions.
- ~ Which of the following are reasons to have multiple subscriptions? (Select two.)
- ~ What is an alternate utility integrated within the Azure Portal a user can access to complete cloud-based support actions?
- ~ What contains web apps, databases, and storage accounts that are deployable and managed in Azure?
- ~ Azure SQL and Azure Cosmos DB are considered what type of service in an Availability Zone.
- ~ Review the following statement. Look at the italicized text. Indicate if the statement requires any corrective actions.
- ~ Review the following statement. Look at the italicized text. Indicate if the statement requires any corrective actions.

#### => Azure Resources :

- ~ Review the following scenario and select the most appropriate response.
- ~ Indicate if the following statements are true or false.
- ~ Indicate if the following statements are true or false.
- ~ Which of the following is a PaaS-based nonrelational Azure Database offering?
- ~ When you are looking to implement a development virtual machine instance with excess storage in a particular region at a significantly reduced rate, what would you need to select during the configuration process of your virtual machine instances?
- ~ You recently received an invoice from Microsoft indicating 720 hours of virtual machine usage. You were surprised, considering you only accessed the virtual machine twice the entire month. To avoid being charged for unnecessary usage, what must you do?
- ~ Which of the following describes a virtual machine that can be deployed across multiple update and fault domains to maximize availability? Which also ensures resiliency due to data centre outages and unplanned maintenance events.
- ~ Which of the following is not a configuration you must identify when setting up an app service plan?
- ~ Which of the following are open-source relational database platforms that Microsoft Azure supports as managed service offerings? (Select two.)
- ~ Your organization requires a managed solution that can support its massive online transactional processing database solution. To ensure optimal performance, your team requires a solution that supports applications with high volume activities and low input/output rates. Autoscaling and fluid storage capacity are desired. Which service tier should you select?
- ~ Review the following scenario and select the most appropriate response.
- ~ You have a website with light traffic. Which type of disk storage is appropriate?
- ~ There are three types of blobs, also referred to as containers. Which of the following is not one of those types?
- ~ What are the differences between virtual machines and Azure Container Instances? (Select two.)
- ~ You need to deploy an Azure virtual machine running Windows 2019. You need to ensure that the services running on the virtual machine are available if one of the assigned data centres fails. You deploy the virtual machines to two Availability Zones. Does that meet the goal?

#### => Management Tools and Solutions :

- ~ Which of the following management tools is a command-line interface that is browser-based and machine-OS-independent?
- ~ Select the response that best fits the questions:
- ~ Review the following scenario and replace the word you believe is inaccurate with one of the following choices.
- ~ Review the following scenario and replace the word you believe is inaccurate with one of the following choices.
- ~ If a user cannot run PowerShell as an Admin or Superuser, which command line should they execute?
- ~ Which of the following is not a capability that one can complete with the Azure Mobile App?
- ~ The most efficient way to distribute Azure Resource Management (ARM) templates is using which tool?
- ~ Your organization is experiencing an outage on all its virtual machine instances. Where should you check first to determine the cause of this issue?
- ~ Select the correct pairing of definitions based on the options presented. Place the items found in the right column in the correct order.
- ~ Under what circumstances would you configure your Synapse Analytics environment to be always available?
- ~ Which of the following solutions is used to help model data from sources such as data warehouses and data lakes to train machine learning models?
- ~ Which of the following solutions is like a Hadoop cluster for processing big data?
- ~ Review the following scenario and replace the word you believe is inaccurate with one of the following choices.
- ~ Review the following scenario and replace the word you believe is inaccurate with one of the following choices.
- ~ Select the response that best fits the questions:

#### => General Security and Network Security :

- ~ Which of the following is not a type of Azure Firewall rule?
- ~ When managing NSG traffic, what is the available traffic range allowed at the uppermost limit?
- ~ Match the correct definition from the left column to the right column.
- ~ The only way to ensure FIPS 140-2compliant security for keys, certificates, or secrets using Azure Key Vault is to:
- ~ Fill in the following statement with the correct response.
- ~ Which of the following is not true about Secure Score?
- ~ Select the regulatory and compliance measure that does not appear as part of the Secure Center dashboard.
- ~ Determine if the answer is True or False in the following three statements:
- ~ Resource Hygiene quality is determined by two factors. What are they?
- ~ Complete the following statement:
- ~ Which of the following Microsoft network security products utilize IP addresses and domains data to protect victims of attacks? The data collected becomes part of the Microsoft Threat Intelligence Feed.
- ~ Defence in Depth is analogous to what type of building?

#### => Identity, Governance, Privacy, and Compliance :

- ~ Which of the following is not a sovereign region?
- ~ Azure Blueprint may contain which of the following governance assets? (Select all that apply.)
- ~ Which Microsoft document repository centrally houses all the security, privacy, and compliance information about Azure?
- ~ Complete the following statement by selecting the correct term.
- ~ Which of the following are not best practices as part of Azure Policy processes?
- ~ The Microsoft Privacy Statement incorporates all of the business terms except:
- ~ Which of the following supplies information or metadata about a resource when classifying and codifying resource management, cost management, optimization, operations management, security, governance and regulatory compliance, workloads such as virtual machines, and automated solutions?
- ~ Complete the following statement.
- ~ What is the purpose of single sign-on?
- ~ This authenticating requires a secondary device such as an e-mail, SMS message, or voice-based call to generate a random number for authentication.
- ~ Which tool allows for collaboration for users who need to connect from outside an organization to access specific resources, specifically guest enterprise users?
- ~ Complete the following statement:

#### => Cost Management and Service-Level Agreements :

- ~ Which of the following service lifecycle states offer products and services SLAsupport?
- ~ Correct the following statement, should it be needed.



- ~ This question requires you to evaluate the use case. Select the condition that makes the following statements correct.
- ~ Select True or False for each of the following statements.
- ~ Service-level agreements typically include all the following, except:
- ~ Select all the options that are applicable from the drop-down menu.
- ~ Select True or False for each of the following statements.
- ~ Which of the following conditions can heavily influence the pricing of a product or service in Azure Marketplace?
- ~ Which of the following is a guarantee for Azure SQL Server?
- ~ Match the terms found on the right with the definition found on the left.

=> others :

- ~ To what should an application connect to retrieve security tokens?
- ~ What is required to use Azure Cost Management?
- ~ Which Azure service should you use to correlate events from multiple resources into a centralized repository?
- ~ What are two characteristics of the public cloud? Each correct answer presents a complete solution. NOTE: each right selection is worth one point.

# Complete iOS 16 Developer with SwiftUI and 8 Apps

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : IOS

Course link : <https://ineuron.ai/course/Complete-iOS-16-Developer-with-SwiftUI-and-8-Apps>

## Course Description :-

Learn iOS development with SwiftUI and building a lot of apps.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to iOS development
- => Xcode
- => Operators and Range in Swift
- => String and interpolation
- => Array and methods in Array in swift
- => Dictionary in depth in swift
- => Sets in swift programming
- => Tuples in swift
- => Structs in swift
- => Structs Vs Class
- => Building Project 1 - Profile app
- => Project 2 - Custom shape and slots
- => Project 3 - Calculator with animation
- => Project 4 Splash screen
- => Project 5 - Shopping app with multi screen

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to iOS development :
  - ~ Introduction to iOS development and prerequisites
  - ~ A tour of XCode
  - ~ Hello world in Swift
  - ~ A bit of history of swift with Objective C
- => Getting started with swift :
  - ~ Variables and Constants in Swift
  - ~ Operators and Range in Swift
  - ~ String and interpolation
  - ~ Methods in Strings
  - ~ A caution in type conversion
  - ~ Can user pay Logical Operators
  - ~ Optional binding and forced unwrapping
  - ~ We missed reading the docs
- => More datatypes in swift :
  - ~ Array and methods in Array in swift

- ~ Dictionary in depth in swift
- ~ Sets in swift programming
- ~ Tuples in swift

=> Going all indepth of swift :

- ~ if else and optional unwrapping
- ~ Control flow statements
- ~ Functions in swift programming
- ~ Indepth of Closure 2C autoclosure and escaping
- ~ Enums and indirect enums
- ~ Structs in swift
- ~ Structs Vs Class
- ~ Classes and reference type
- ~ Properties in swift
- ~ Methods in swift

=> Advance swift programming concept :

- ~ Inheritance in swift
- ~ init in depth in swift
- ~ Deinit in swift
- ~ Error handling in swift
- ~ Protocols in swift

=> Building Project 1 - Profile app :

- ~ Zstack 2C HStack and VStack
- ~ Create a new app in XCode
- ~ Getting started with Zstack and VStack
- ~ Moving into VStack
- ~ Nested Stacks in swift UI
- ~ Finishing our first app

=> Project 2 - Custom shape and slots :

- ~ Theory behind custom shapes in iOS
- ~ From figma to XCode shape
- ~ State 2C rawValue and Identifiable
- ~ More on State and HStack
- ~ Getting button in our app
- ~ Finishing up slot machine game

=> Project 3 - Calculator with animation :

- ~ RawValue in swift
- ~ Starting a calculator project - assets
- ~ Defining Model for calculator
- ~ Getting keys sorted out for calculator
- ~ Animation in swift ui
- ~ Adding buttons for calculator
- ~ Learn to calculate element width and height
- ~ Loading up views on home screen
- ~ Finishing up the calculator logic part

=> Project 4 Splash screen :

- ~ Getting started with Splash screen
- ~ Finishing up a splash screen

=> Project 5 - Shopping app with multi screen :

- ~ Demo of Shopping app with Navigation
- ~ Importing all assets of fruits
- ~ Building on boarding screen with navigation
- ~ Models for fruit and near you
- ~ Handling the fruit card
- ~ Horizontal scroll view
- ~ Passing value from one screen to another
- ~ Design detail view part 1
- ~ Counter in detail screen
- ~ Vertical scroll view
- ~ Assemble fruit cart app
- ~ Resolving minor UI issue

=> Project 6 - Building LinkedIn UI clone :

- ~ What we will build - LinkedIn
- ~ Search bar component
- ~ Models in linkedin UI
- ~ Each connection request
- ~ Building my Network screen
- ~ Making home cards
- ~ Home screen top view
- ~ Building Home Screen
- ~ Launch linkedin UI in simulator

=> Project 7 - Todo App - Read the docs :

- ~ What are user defaults
- ~ What is Codable protocol
- ~ Model with Identifiable and Codable
- ~ What are ObservableObject and Published
- ~ UserDefaults with unique key
- ~ Get values from UserDefaults
- ~ CRUD operations in Todo list
- ~ DispatchQueue in depth
- ~ Navigation View and Link
- ~ State management in swift ui
- ~ Take user input and add it to Model

- ~ Adding Todo 27s on Home screen
- ~ Finishing up todo app with gesture implementation

=> Project 8 - Handling API and building pokemon app :

- ~ What is API and formatting
- ~ Create a model for API response
- ~ Fetching data from API endpoint
- ~ List and async calls
- ~ Kingfisher - Third party packages
- ~ Install third party packages
- ~ What are extensions in swift
- ~ Issues in Data and API call
- ~ Creating a data extension
- ~ Using KFIImage
- ~ Gridviews and LazyVStack
- ~ Debugging the pokemon app

# AWS Data Engineering

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA ON CLOUD

Course link : <https://ineuron.ai/course/AWS-Data-Engineering>

## Course Description :-

Welcome to AWS Data Engineering. If you're new to the cloud, whether you're in a technical or non-technical role such as finance, legal, sales, and marketing, this course will provide an understanding of fundamental AWS Cloud concepts to help you gain confidence to contribute to your organization's cloud initiatives. This course is also the starting point to prepare for your AWS Certified Cloud Practitioner certification whenever it's convenient for you.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What is EMR
- => EBS Hands On
- => VPC, Subnet, Internet Gateway & NAT Gateways

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => EMR :
  - ~ Why Cloud & Big Data on Cloud
  - ~ What is Virtual Machine
  - ~ On-Premise vs Cloud Setup
  - ~ Major Vendors of Hadoop Distribution
  - ~ Hdfs vs S3
  - ~ Important Instances in AWS
  - ~ Overview of EMR
  - ~ What is EMR
  - ~ Tez vs mapreduce
  - ~ Launching an emr cluster
  - ~ connecting to your cluster
  - ~ Create a tunnel for web ui
  - ~ Use Hue to interact with EMR
  - ~ Part 1 analyze movie ratings with hive on emr
  - ~ Part 2 analyze movie ratings with hive on emr
  - ~ Transient vs Long Running Cluster Running
  - ~ Copy File From S3 to Local Zeppelin Notebook
  - ~ How to Create a VM
  - ~ S3 & EBS
  - ~ Public ip Vs Private Ip
  - ~ Aws Command Line Interface
  - ~ What is Athena
  - ~ When do we require Athena What problem Athena Solve How Athena Works
  - ~ Athena Pricing
  - ~ Athena Practical Demonstration
  - ~ AWS Glue
  - ~ Introduction to Amazon Redshift
  - ~ Redshift Master Slave Architecture
  - ~ redshift demo
  - ~ redshift spectrum
  - ~ Redshift Distribution Styles
  - ~ Redshift Fault Tolerance
  - ~ Redshift Sort Keys
- => EC2 Instance Storage :
  - ~ EBS Overview
  - ~ EBS Hands On

- ~ *EBS Snapshots Overview*
- ~ *EBS Snapshots Hands On*
- ~ *AMI Overview*
- ~ *AMI Hands On*
- ~ *EC2 Image Builder Overview*
- ~ *EC2 Image Builder Hands On*
- ~ *EC2 Instance Store*
- ~ *EFS Overview*
- ~ *Shared Responsibility Model for EC2 Storage*
- ~ *Amazon FSx Overview*
- ~ *EC2 Instance Storage Summary*

=> VPC and Networking :

- ~ *VPC Overview*
- ~ *VPC, Subnet, Internet Gateway & NAT Gateways*
- ~ *Security Groups & Network Access Control List (NACL)*
- ~ *VPC Flow Logs & VPC Peering*

# AIOPS Live Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS PROJECT

Course link : <https://ineuron.ai/course/AIOPS-Live-Projects>

## Course Description :-

Learn how to create a machine learning system from start to finish. Develop skills in training, deploying, scaling, and monitoring your machine learning model's performance in production. This course is specifically designed for deploying and scaling machine learning and deep learning applications.

## Course Features :-

- => Online live classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Design end-to-end machine learning system
- => Monitor and visualize the performance of apps
- => Build CI/CD pipelines
- => Optimizing the model training & prediction pipelines

## Requirements :-

- => Minimum System requirement: Intel Core i3 processor and 4GB RAM or Higher
- => A system with a decent internet connection
- => AWS, Azure, GCP, Digital Ocean accounts
- => Your dedication
- => Interest to learn

## Instructors :-

=> Avnish Yadav :

~ 3+ years of experience in various domains such as data scientist, data analyst, database developer, and .net developer. Implemented various sophisticated business requirements, performed an analysis of various data to capture insights and hidden patterns. Fine and tuned various regression and classification-based algorithms for prediction. Implemented various ETL pipelines to fulfil the business requirement. Automated various machine learning pipelines such as data loading, data cleaning, data validation, model selection, model tuning, and model monitoring pipeline. Implemented machine learning pipeline in azure machine learning studio. I have a keen interest to solve complicated machine learning problems to fulfil business requirements.

## Curriculum details :-

=> Building Machine learning Pipeline :

- ~ Overview of Machine Learning Pipeline
- ~ Need for Machine Learning Pipeline
- ~ Discussion on each step of ML Pipeline
- ~ Introduction to Tensorflow Extend
- ~ Task Communication with each other
- ~ TFX component Internal Mechanism
- ~ Machine Learning Meta data Store and It's uses
- ~ Introduction to Apache Beam
- ~ TFX component internal uses of Apache Beam

=> Data Ingestion :

- ~ About Data Ingestion
- ~ Retrieval of data and data versioning

=> Data Validation :

- ~ About Data Validation
- ~ Data Validation using TensorFlow Data Validation
- ~ Implementation of alter due to data drift

=> Data Preprocessing :

- ~ About Data Preprocessing
- ~ Feature Engineering using Tensorflow Transform

=> Model Training and Model Tuning :

- ~ Discussion on model training
- ~ Implementation of model training in ML Pipeline
- ~ Discussion on model tuning
- ~ Implementation of model tuning

=> Model Evaluation :

- ~ Overview of Model Evaluation
- ~ Understanding useful metrics to evaluate model performance
- ~ Capturing biases of model
- ~ Versioning of Model

=> Model Deployment :

- ~ Model Deployment using TensorFlow Serving
- ~ Simple Flask Implementation
- ~ Implementation of monitoring deployed model
- ~ Deployment using Kubernetes

=> Integration with Apache Beam and Apache Airflow :

- ~ Implementation of Pipeline
- ~ Configure pipeline for Orchestration platform

=> Feedback Loop :

- ~ Understanding of feedback loop
- ~ Implementation of feedback loop to improve models

=> Data Privacy for Machine Learning :

- ~ Understanding the need for data privacy
- ~ Methods to implement data privacy
- ~ Differential Privacy
- ~ Federated Learning
- ~ Encrypted machine Learning

=> Deployment of End to End Pipeline :

- ~ Cloud Deployment (AWS/ GCP/ Azure)



# Class 8th Chemistry

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Class-8th-Chemistry>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. In this, the Chemistry section focuses on concepts like Synthetic Fibers and Plastics, Metals, Non-metals, Fossil Fuels, etc.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Coal and Petroleum
- => Synthetic fibers and plastics
- => Materials - Metals and Non-metals
- => Pollution of Air and Water
- => Combustion and Flame

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => Ch3 Synthetic Fibres & Plastics :
  - ~ Synthetic Fibres and Plastics
- => Ch4 Metals&Non-Metals :
  - ~ Material Metals and Non Metals
- => Ch5 Coal and Petroleum :
  - ~ Coal and Petroleum
- => Ch 6 Combustion and Flame :
  - ~ Combustion and Flame
- => Ch18 Pollution of Air and Water :
  - ~ Pollution of Air and Water Preview

# Chatbot using Rasa NLU

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Chatbot-using-Rasa-NLU>

## Course Description :-

Rasa NLU implementation

## Course Features :-

=> Lifetime Dashboard

=> Free Course

## What you will learn :-

=> Chatbot Using RASA NLU

=> Chatbot Introduction

=> Introduction To RASA

=> FAQ Problem Statement

=> FAQ Chatbot Implementation

=> Telegram Integration

=> Conclusion and Further Work

## Requirements :-

=> Programing understanding

=> NLP understanding

## Curriculum details :-

=> Tutorial 1- FAQ Chatbot Using RASA NLU :

~ *Introduction Preview*

=> Tutorial 2- FAQ Chatbot Using RASA NLU- Chatbot Introduction

=> Tutorial 3- FAQ Chatbot Using RASA NLU- Introduction To RASA

=> Tutorial 4- FAQ Chatbot Using RASA NLU- FAQ Problem Statement

=> Tutorial 5- FAQ Chatbot Using RASA NLU- FAQ Chatbot Implementation

=> Tutorial 6- FAQ Chatbot Using RASA NLU- Telegram Integration

=> Tutorial 7- FAQ Chatbot Using RASA NLU- Conclusion and Further Work

# Industry Safety Detection using YOLO v7

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION PROJECT

Course link : <https://ineuron.ai/course/Industry-Safety-Detection-using-YOLO-v7>

## Course Description :-

The purpose of this project is to create a detection system using Computer Vision and Machine Learning to monitor, track and enforce employees/workers to wear the necessary protection gear. ISD is designed and modeled to take a real-time image of the personnel as the input and determine if the five segments - helmet, gloves, jacket, goggles, and footwear are worn before entering the workplace, and record the procedures as well. If ISD does not find any of the safety gears, the worker will not be allowed to proceed and the prohibition alarm in the system will alert the authorities

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self-Paced Learning
- => Dashboard Access
- => Course Materials
- => Assignments

## What you will learn :-

- => Real Time Projects
- => Industry Safety Detection using YOLO v7
- => Object detection with YOLO v7
- => Data Annotation
- => How to work with Docker
- => Modular coding approach for training and prediction pipeline
- => Building Flask app
- => Learn about AWS basics
- => CICD tools like Github actions
- => Production-grade deployment

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Boktiar Ahmed Bappy :

~ This is Bappy. I aim for simplicity in Data Science. Real Creativity won't make things more complex. Instead, I will simplify them, Interested in a Data Science Career and so develop myself accordingly. Data Scientist and lecturer with working experience in Machine Learning, Deep Learning, Microcontrollers and Electronics systems. Hands-on experience in classification, regression, clustering, computer vision, natural language processing and transfer learning models to solve challenging business problems. I have a huge interest in Robotics. I have innovated a lot of innovations, ideas, projects & robots and got a lot of achievements.

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Project :- Industry Safety Detection using YOLO v7 :

- ~ Introduction of Instructor
- ~ Project Overview
- ~ End Notes
- ~ Problem Description
- ~ Understand the application scope
- ~ Tour to existing solution
- ~ End Notes
- ~ Solution Description
- ~ Notebook Walkthrough
- ~ Tour to Architecture diagram

- ~ cost involved
- ~ End Notes
- ~ Structure overview
- ~ Data Ingestion
- ~ Data Validation
- ~ Data Transformation
- ~ Model Training and Tunning
- ~ Model Evaluation
- ~ Model Pusher
- ~ Training Pipeline
- ~ Prediction pipeline
- ~ Frontend app design
- ~ Tour to the cloud and Service Overview (AWS)
- ~ IAM setup
- ~ ECR setup
- ~ EC2 setup
- ~ Self hosted runner
- ~ Docker
- ~ Conclude the project
- ~ Assignments & External Resources

# Django Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : DJANGO

Course link : <https://ineuron.ai/course/Django-Course>

## Course Description :-

The most widely used Python web development framework is Django. Django is a Python framework that covers all elements of web development, from handling requests and answers to creating dynamic HTML pages using templates and making database access and maintenance simple. This course has it all baked in, and it's all covered in excellent depth. Django is taught from the ground up in this course. We'll start from the beginning and work our way up, learning how to construct Django projects, execute them, and add functionality step by step.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Webpages in Django
- => Admin control
- => Sliders
- => Search fields
- => Navbars
- => Components
- => User authentication
- => Facebook and Google authentication
- => Django message frameworks
- => Django contact forms

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

=> Getting Started with Tubers :

- ~ What we will build
- ~ Introduction to application
- ~ Preparation work for full stack app
- ~ Moving into virtual environment
- ~ Django Installation for tubers app
- ~ Install postgresql and PGAdmin
- ~ Django postgresql config and bugs

=> Django Admin Customization :

- ~ Django superuser
- ~ Theming options for django admin
- ~ Our new theme for django admin
- ~ Full customization of Django admin
- ~ Deploying static files

=> Web pages in Django app :

- ~ creating web pages app

- ~ Handling webpages routes
- ~ Advantages of using templates in django
- ~ loading the base template
- ~ adding static CSS and JS files
- ~ Making project modular in django
- ~ Home page is all set

=> Admin control of slider :

- ~ Creating model for slider
- ~ Adding slider from admin
- ~ passing data to web page
- ~ Fetching values in front end and assignment

=> Team section and admin :

- ~ Team model and registration in admin
- ~ Admin customization for Team
- ~ Read some django docs to edit admin
- ~ Team section in front end
- ~ Assignment for team section

=> YouTubers section on Home Page :

- ~ Creating a youtuber app
- ~ Model for youtubers
- ~ Restricting user choices
- ~ Add youtubers from admin
- ~ Admin modification for youtubers
- ~ Featured youtubers on front end
- ~ Latest onboard section

=> Youtubers component :

- ~ Fixing the navbar
- ~ youtubers views preparation
- ~ Tubers component in front end
- ~ fetching single tuber detail
- ~ Fetching formatted description

=> Search fields in Django :

- ~ Keyword based search
- ~ Search component on home page
- ~ Exact search feature in django

=> Authentication of users :

- ~ Creating accounts app and setting views
- ~ Configure templates
- ~ header changes for login and register
- ~ Messages and registering a user
- ~ Safety nets for registration of user
- ~ login feature of tubers app
- ~ Django decorators and authentication
- ~ FB and Google authentication
- ~ create a new Facebook app
- ~ Final check for FB login

=> Django messages framework and more :

- ~ Messaging framework
- ~ Fixing remaining navbar
- ~ Adding about and contact page

=> Contact form in Django :

- ~ a new hiretuber app
- ~ creating a model for the form
- ~ Handling views for contact form
- ~ Form front end part 1
- ~ Form front end part 2 and debugging

=> Assignments for you :

- ~ assignments for you

# Java Bootcamp

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Java-Bootcamp>

## Course Description :-

This course has been designed to help you become a complete and professional Java engineer at the conclusion of the course, rather than only teaching essential Java skills. To guarantee that you grasp the Java language, the course has been designed to be very thorough, covering the majority of Java language features and explaining them in great detail. Tons of best practises and design ideas are described and illustrated in code to guarantee you are industry-ready and can create well-designed, professional code.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Java data types and variables
- => Java operations and conditionals
- => Arrays
- => Object-Oriented java
- => Searching and Sorting
- => Linked Lists
- => Various java projects

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Before we get started with java :

- ~ Important instructions before getting started
- ~ Installing Java on Windows and setting path
- ~ Installing IntelliJ on WINDOWS
- ~ Installing IntelliJ on MAC
- ~ Hello World in Java
- ~ Explore your IDE

=> Basics of java :

- ~ What are variables in java
- ~ Add two numbers by taking input from users
- ~ Comments and TODOs are important
- ~ Primitive Datatypes in Java
- ~ Int Float and double
- ~ Unicode values and Booleans
- ~ Strings in JAVA

=> Operations and conditionals in java :

- ~ Arithmetic Operations and pre postfix in Java
- ~ Relational Operations in Java
- ~ Logical Operations in Java
- ~ Assignment operations and reading materials
- ~ Decision making with If and Else
- ~ Switch case and break in Java
- ~ While loop in Java

- ~ Do while loop in Java
- ~ For loop in Java

=> University question specials- Practice java :

- ~ Comparing two numbers
- ~ Finding Odd and Even numbers
- ~ Leap Year or not
- ~ Largest of 3 numbers
- ~ Fahrenheit to Celsius
- ~ Factorial of a number
- ~ Printing Star and number patterns
- ~ Generating random numbers between 1 and hundred
- ~ Students grade greeting systems

=> Array and a small project of TODO list :

- ~ Array - introduction and loop exercise
- ~ Fun exercise with Java
- ~ Access Modifiers in Java
- ~ Methods in Java
- ~ ArrayList in Java
- ~ Todo list project part 1
- ~ Todo list with Arraylist part 2
- ~ Todo list part 3
- ~ Todo list in java FINAL

=> Object Oriented Programming Concepts :

- ~ Modern datatypes, Autoboxing and unboxing
- ~ Classes and constructor in Java
- ~ Objects in java
- ~ basics of inheritance
- ~ Method override and type of inheritance in Java
- ~ Encapsulation - getters and setters
- ~ Abstraction and its usage in Java
- ~ Polymorphism in Java

=> Basics of sorting Algorithms and their code in java :

- ~ Bubble sort algo and code in Java
- ~ Selection sort code in Java
- ~ insertion sort in java
- ~ Process API updates and jShell in Java 9
- ~ Try catch and exception in Java
- ~ Malformed URL exception
- ~ Finding day of any date

=> Generics, Advance Collection and Linked List :

- ~ Generics introduction and forIn loop in java
- ~ Generic methods and Comparable class
- ~ Generic classes and objects
- ~ New object initialization is tricky sometimes
- ~ Linked list part 1
- ~ Linked List part 2
- ~ You are ready to move into Android now

=> Getting started with GUI - fxml :

- ~ Getting started with GUI Applications
- ~ Your first fxml file
- ~ Layouts and buttons for GUI
- ~ Connecting code and buttons
- ~ Getting input from users
- ~ Connecting Multiple buttons
- ~ Fixing your scene builder issues

=> Video Player Project #2 :

- ~ Getting basics of Video player
- ~ Adding sliders for time and volume
- ~ Making Volume and seekbar to work

=> Improve your desktop design skills with scene builder :

- ~ Learn to design your desktop App



# Python Crash Course

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON

Course link : <https://ineuron.ai/course/Python-Crash-Course>

## Course Description :-

Throughout this course, you will learn everything you need to know about Python, from the basics to advanced topics. In addition, python applications such as Download Manager will be built using sophisticated techniques to assist you in becoming a professional programmer capable of securing well-paying jobs.

## Course Features :-

- => Quizzes
- => Assignments
- => Hands-on practicals
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Python installation and setup
- => OOPs concept
- => Databases
- => Create API using Flask
- => Pandas

## Requirements :-

- => Basic knowledge of Python programming
- => A system with stable internet connection
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Python :

- ~ Python installation and setup Preview
- ~ Python basics Preview
- ~ while loop
- ~ For loop
- ~ String and list manipulation
- ~ List manipulation
- ~ Tuple, set, dictionary
- ~ Function
- ~ Logging and debugging
- ~ Modules and exception
- ~ Class and object Preview
- ~ Abstraction and inheritance

=> Databases :

- ~ Class reusing your classes, functions and methods, intro to databases
- ~ Connection of SQL with Python
- ~ Introduction to MongoDB and Atlas
- ~ Cassandra

=> Flask :

- ~ Flask
- ~ Flask API
- ~ Django
- ~ Pandas part-1
- ~ Pandas part-2

# OpenCV Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION PROJECT

Course link : <https://ineuron.ai/course/OpenCV-Projects>

## Course Description :-

This course will provide you with various OpenCV-based exciting projects to improve your skill and creativity. These projects will also inspire you to build computer vision-based projects in real-time.

## Course Features :-

- => Challenges
- => Various projects
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Develop various projects with OpenCV library and deep learning techniques.
- => Project presentation skills
- => Feature Extraction

## Requirements :-

- => Prior understanding of OpenCV library.
- => Prior knowledge and hands-on in Python programming
- => A system with decent internet connection
- => Your dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Introduction :
  - ~ Overview Preview
- => Feature extraction :
  - ~ Project overview Preview
  - ~ Inpainting
  - ~ Haar cascading
  - ~ Corner detection
  - ~ Advance corner detection
  - ~ Feature matching
  - ~ Template matching

# MERN Stack Beginner to Advance with Internship

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/MERN-Stack-Beginner-to-Advance-with-Internship>

## Course Description :-

This is a MERN stack live mentor led certification program along with full time 6 months internship provided by iNeuron intelligence private limited, where you will learn all the stack required to work in MERN, hosting including cloud technologies and real time industry project and product development along with iNeuron product development team and you will contribute on various level with iNeuron .

## Course Features :-

- => Online Instructor-led learning: Live teaching by instructors
- => Every week doubt clearing session after the live classes
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Assignment in all the module
- => Quiz in every module

## What you will learn :-

- => HTML
- => CSS
- => Javascript
- => Diving deep in core JS
- => Exploring Functional nature of JS
- => JSX in Depth
- => MongoDB
- => Node.js
- => Databases in Node.js
- => NoSql databases
- => File Handling
- => Building a Full fledge e-commerce application
- => Socket API
- => Testing in Node

## Requirements :-

- => Dedication
- => PC with i3 processor and internet connectivity

## Instructors :-

- => Keshav Singh :
- ~

## Curriculum details :-

- => Course Introduction :
  - ~ Overview of whole course
  - ~ Roadmap
  - ~ How to make most from this course
  - ~ Prerequisite
- => HTML :
  - ~ Introduction to HTML
  - ~ Different types of elements
  - ~ Container elements
  - ~ Handling forms in HTML
  - ~ Diving Deep in HTML
  - ~ DOM
  - ~ Performance and Optimisations
- => CSS :
  - ~ Selectors in CSS
  - ~ Diving into Basics
  - ~ Positioning of Elements
  - ~ Understanding Background Images and Simple Images
  - ~ Sizes and Units

- ~ Applying CSS to our Portfolio website
- ~ Flexbox
- ~ Grids in CSS
- ~ Making our Portfolio website responsive
- ~ Animations
- ~ Implementing animations into our Portfolio website

=> Javascript :

- ~ Introduction to JS
- ~ Types in JS
- ~ Coercion
- ~ Diving deep into basics
- ~ Hoisting
- ~ Scopes and Closure
- ~ Lexical Scope
- ~ Prototypes in JS
- ~ Async nature of JS
- ~ How JS controls the webpage
- ~ Events
- ~ Event Loop in JS
- ~ AJAX
- ~ Different types of interfaces such as File, Blob, FormData, etc
- ~ Animations using JS

=> Diving deep in core JS :

- ~ Functions
- ~ Different types of Functions
- ~ Closures in Deep
- ~ This binding
- ~ Objects
- ~ Synchronous and Asynchronous JS
- ~ Iterators
- ~ Generators
- ~ Promises
- ~ Async and Await

=> Exploring Functional nature of JS :

- ~ Pure Functions
- ~ Higher Order Functions
- ~ Immutability
- ~ Point free code style

=> Introduction To React and Why to use React Today :

- ~ Why JSX
- ~ How React Works under the hood
- ~ Components (Stateful and Stateless component)
- ~ Conditional Rendering
- ~ Lifecycle Hooks of Components in React
- ~ Handling Events and What is synthetic events in react
- ~ Abstraction Layer of React
- ~ Handling Forms

=> Introduction to some JS new features :

- ~ Spread Operator
- ~ Arrow Function
- ~ Rest Parameters
- ~ Higher Order Functions
- ~ Closures
- ~ Block Level Scoping(let and const)
- ~ How to share logic between Components by different methods
- ~ Render Props
- ~ Higher Order Components
- ~ Context Api
- ~ Hooks (An Introduction to it here)
- ~ JSX in Depth

=> Some More JS new features :

- ~ Callbacks
- ~ Promises
- ~ Async and Await
- ~ Iterators and Iterables
- ~ Generators
- ~ Introduction to Asynchronous tasks and how to handle it in JS world
- ~ Handling error using promises and Error Boundaries in React
- ~ State Management in React using Context Api and Redux
- ~ Diving Deep in Redux and making a loose version of Redux from scratch.
- ~ Getting Redux Thunk and Redux Saga

=> Handling Forms :

- ~ Validating the form values in synchronous and asynchronous way
- ~ Displaying Error
- ~ Routing in React
- ~ Protecting Routes and redirecting as needed
- ~ Deep Dive in React Router Library and get our feet wet with it
- ~ Fragments, Profilers and deep dive in Virtual DOM.
- ~ Optimizing our React Application
- ~ Reconciliation algo in react
- ~ Diving Deep in React Hooks
- ~ More Optimizing the React Apps using Hooks
- ~ Code Splitting and making our app more fast

=> MongoDB :

- ~ Understanding the basics and CRUD operations
- ~ Schemas and Relations
- ~ Exploring the Shell and the server
- ~ Understanding the MongoDB compass and setting it for our project
- ~ Read Operations diving deep
- ~ Create Operations diving deep
- ~ Understanding Indexes
- ~ Understanding Aggregation
- ~ Handling security

=> Node.Js :

- ~ How Node.js works
- ~ Request and Response Mechanism in node
- ~ Redirecting based on some condition

=> Introduction to Express :

- ~ Diving Deep in some Express concepts
- ~ Express Middlewares and why it is important
- ~ Building our custom middlewares
- ~ Understanding some of middlewares such as body-parser and cors and its uses.
- ~ Introduction to Server Side Validations
- ~ Express validators and its uses.
- ~ Writing our custom validator (synchronous and asynchronous)
- ~ Handling errors while validation

=> Introduction to session management in Node. :

- ~ How to manage session using cookies and jwt(json web token)
- ~ Introduction to express-session

=> Handling Dynamic Routes in Node :

- ~ How to fetch Query Parameters and Params in Node
- ~ Redirection based on url
- ~ Absolute and Relative path handling using build in Node module
- ~ Introduction to streams
- ~ How Node is based on streams and how to work with it
- ~ Readable and Writable Streams
- ~ Process and Subprocess in Node
- ~ Multithreading in Node

=> Handling Databases in Node.js :

- ~ First, Mysql Database
- ~ Using mysql library we build mysql database
- ~ Using sequelize library for handling mysql database
- ~ Introduction to NoSql databases
- ~ Using MongoDB as a NoSql database
- ~ Using mongoose package
- ~ Templating Engines in Node.js
- ~ Introduction to EJS and handlebars
- ~ Handling errors and flash messages in this

=> Handling File Uploads in node.js :

- ~ Handling One file upload and then multiple files uploads
- ~ Using package multer for this.
- ~ Introduction to socket api and understanding how to real time application

=> Testing in Node :

- ~ Testing in Node
- ~ Unit Testing in Node

## Project details :-

=> HTML :

- ~ Building a simple Portfolio website skeleton

=> CSS :

- ~ Building a Blog applications Frontend using above discussed concepts

=> Javascript :

- ~ Building a Todo Application using JS
- ~ Building a Snake Game using core JS

=> Exploring Functional nature of JS :

- ~ Refactoring our Todo Application in Functional style

=> Introduction To React and Why to use React Today. :

- ~ Building a Tic Tac Toe game over the concepts given above

=> Introduction to some JS new features :

- ~ A Hangman Application

=> Some More JS new features :

- ~ Building a TODO Application using the above concepts

=> Handling Forms :

- ~ Building a React Quiz Platform using Hooks
- ~ Stating our main project that is a e-commerce application.
- ~ Building authentication using firebase.
- ~ Persisting the session on the client
- ~ Making different parts of the app in sync

=> MongoDB :

- ~ Transactions and implementing this in our own project

=> Node.Js :

~ Building a simple multipage application using Node.js

=> Introduction to Express :

~ Re-writing our Part 4 project using express

~ Building a TODO Application using Node.js and building the server side validations as well

=> Introduction to session management in Node :

~ Making a user authentication using all the above concepts such as express-session, express-validator and middlewares

=> Handling Databases in Node.js :

~ Building a Basic Info Management System using No Sql

~ Building a TODO Application using ejs

=> Handling File Uploads in node.js :

~ Building a Resume Reader System using node

~ Building a Full fledged e-commerce application

~ Handling Payments in this

~ Error Handling

~ Making a chatting application like whats app

# Fundamentals of Game Development

---

Topic Name : K12

Sub-topic Name : CLASS9

Course link : <https://ineuron.ai/course/Fundamentals-of-Game-Development>

## Course Description :-

Learn the Fundamentals of Game Development from Scratch using Pygame. This course is intended for anybody interested in learning Game Development with Python. The game's difficulty rises with each component, and you'll be able to expand your knowledge as you go through the course. You'll create wonderful games and learn how PyGame works by moving things about on the screen and interacting with items. You'll also learn how to build and import game-related images, as well as how to generate randomly moveable enemies, animate game characters, and play music and noises while playing the game.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with tech
- => Real-time project
- => Live class recording
- => Doubt clearing
- => Assignment in all the module
- => Quiz in every module
- => Career Counselling
- => Completion certificate

## What you will learn :-

- => Introduction to Pygame
- => Deep-dive into Pygame
- => Game world
- => Sprites
- => Projects on game development

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction :
  - ~ Course Introduction
  - ~ Course Pre-requisites
  - ~ Who is this course for?
  - ~ What you will get from this course?
  - ~ Introduction to digital games
  - ~ How to get access to course materials?
  - ~ What career path you can follow after completion of this course?
- => Assignment 1: :
  - ~ What do you mean by graphics? How it is displayed?
- => Introduction to Pygame :
  - ~ What is Pygame?
  - ~ History of Pygame
  - ~ Installing Pygame
  - ~ Modules in Pygame
  - ~ Importing and initializing Pygame
  - ~ Creating Pygame window
  - ~ Opening full screen displays
  - ~ Pygame surface
  - ~ Pygame Clock
  - ~ Pygame blit
  - ~ Adding image in Pygame
  - ~ Keydown and Keyup
  - ~ Pygame draw
  - ~ Drawing rectangle on Pygame
  - ~ Other shapes

- ~ Pygame text and drawing
- ~ Hello world program in Pygame
- ~ Sprite
- ~ Collision detection

=> Assignment 2: :

- ~ Create a circle and when pressed keydown it should come down and become cylinder

=> Deep-dive into Pygame :

- ~ Understanding events
- ~ Standard events
- ~ Frame rate
- ~ Moving sprite into straight line
- ~ Diagonal movement
- ~ Vectors
- ~ Using vectors to create movement
- ~ Controlling the game
- ~ Keyboard control
- ~ Detecting key press
- ~ Directional movements with keys
- ~ Mouse Control

=> Assignment 3: :

- ~ Create your own sprite and move it using keyboard

=> Game world :

- ~ Introduction to Game Entities
- ~ Building world for entity
- ~ Building the brain: Actions for entities
- ~ Practical: Game World with actions for entities

=> Assignment 4: :

- ~ Add or draw your own entity in the game and define action for it.

=> Projects :

- ~ Project: Bricks game

=> Projects :

- ~ Project: Snake game

=> Summary :

- ~ Course Outro
- ~ Future Scope of Pygame



# Business Analytics

---

Topic Name : DATA ANALYTICS

Sub-topic Name : BUSINESS ANALYTICS MASTERS

Course link : <https://ineuron.ai/course/Business-Analytics>

## Course Description :-

Learn the power of using powerful visualization tools such as PowerBi and Tableau alongside advanced excel coupled with the most important fundamentals of Python

## Course Features :-

- => Business Analytics Certification
- => Online Instructor-led learning: Live teaching by instructors
- => Hands-on project implementation
- => 100+ hours of live interactive classes
- => Every week doubt clearing session after the live classes
- => Lifetime Dashboard access
- => Assignments in all the module
- => Live class recordings and materials
- => Interview Questions

## What you will learn :-

- => Python
- => PowerBI
- => Tableau
- => Advanced Excel
- => Statistics

## Requirements :-

- => Laptop
- => Stable internet connection
- => Your Dedication

## Curriculum details :-

=> Introduction to Analytics

=> Python for Data Analytics :

- ~ Install setup and overview
- ~ Ipython/Jupyter Notebook overview
- ~ Intro to NUMPY
- ~ Creating Arrays.
- ~ Using Arrays and Scalar
- ~ Indexing Arrays
- ~ Arrays transposition
- ~ Universal arrays function
- ~ Arrays processing
- ~ Array input and output
- ~ Series
- ~ DataFrames
- ~ Index Objects
- ~ Re-index
- ~ Drop entry
- ~ Selecting entries
- ~ Data alignment
- ~ Rank and Sort
- ~ Summary statistics
- ~ Missing data
- ~ Index Hierarchy
- ~ Reading and writing text files
- ~ JSON with Python
- ~ HTML with Python
- ~ Microsoft Excel files with Python
- ~ Merge
- ~ Merge on Index
- ~ Concatenate
- ~ Combining Data Frames
- ~ Reshaping
- ~ Pivoting
- ~ Duplicates in DataFrames

- ~ Mapping
- ~ Replace
- ~ Rename index
- ~ Binning
- ~ Outliners
- ~ Permutation
- ~ GroupBy on DataFrames
- ~ GroupBy on Dict and Series
- ~ Aggregation
- ~ Splitting, Applying and combining.
- ~ Cross Tabulation
- ~ Installing Seaborn
- ~ Histograms
- ~ Kernel Density estimate plots
- ~ Combining plot styles
- ~ Box and Violin plots
- ~ Regression Plots
- ~ Heat maps and clustered matrices
- ~ Introduction to SQL with Python
- ~ SQL - SELECT, DISTINCT, WHERE, AND & OR
- ~ SQL WILDCARDS, ORDER BY, GROUP BY, and Aggregate Functions

#### => SQL FOR DATA ANALYTICS :

- ~ Introduction.
- ~ ER Diagram.
- ~ Schema Design.
- ~ Normalization.
- ~ SQL SELECT statement.
- ~ SQL SELECT using common functions.
- ~ SQL JOIN overview.
- ~ INNER JOIN.
- ~ LEFT JOIN.
- ~ RIGHT JOIN.
- ~ FULL JOIN.
- ~ SQL best practice.
- ~ INNER JOIN Advanced.
- ~ INNER JOIN and LEFT JOIN combo.
- ~ SELF JOIN.
- ~ JOINS and AGGREGATION Subqueries.
- ~ Sorting.
- ~ Independent Subqueries.
- ~ Co related Subqueries.
- ~ Analytic function.
- ~ Set operations.
- ~ SQL views.
- ~ Create a view.
- ~ Create a view using DDL.
- ~ SQL insert Advanced Technique.
- ~ Insert to create table.
- ~ INSERT to new data on existing table 1.
- ~ INSERT to new data on existing table 2.
- ~ INSERT to new data on existing table 3
- ~ INSERT to new data on existing table 4.
- ~ SQL update Advance technique and TCL.
- ~ SQL delete and TCL.
- ~ SQL constraints.
- ~ SQL aggregations.
- ~ SQL programmability.
- ~ SQL query performance.
- ~ SQL Extras.

#### => Advance Excel

#### => Data wrangling with Excel :

- ~ Microsoft Excel fundamentals.
- ~ Entering and editing texts and formulae.
- ~ Working with basic Excel functions.
- ~ Modifying an Excel worksheet.
- ~ Formatting data in an excel worksheet.
- ~ Inserting images and shapes into an Excel worksheet.
- ~ Creating Basic charts in Excel.
- ~ Printing an Excel worksheet.
- ~ Working with an Excel template.
- ~ Working with an excel list.
- ~ Excel list function.
- ~ Excel data validation.
- ~ Importing and exporting data.
- ~ Excel pivot tables.
- ~ Working with excels PowerPivot tools.
- ~ Working with large sets of Excel data.
- ~ Conditional function.
- ~ Lookup functions.
- ~ Text based functions.
- ~ Auditing and Excel worksheet.
- ~ Protecting Excel worksheets and workbooks.
- ~ Mastering Excel "What if?" Tools?
- ~ Automating Repetitive Tasks in Excel with Macros.
- ~ Macro Recorder Tool.
- ~ Excel VBA Concepts.

- ~ Advance VBA.
- ~ Preparing and Cleaning Up Data with VBA.
- ~ VBA to Automate Excel Formulas.
- ~ Preparing Weekly Report.
- ~ Working with Excel VBA User Forms.
- ~ Importing Data from Text Files.

#### => Business Statistics :

- ~ Descriptive Analytics.
- ~ Inferential Statistics.
- ~ Hypothesis Test 1 & 2.
- ~ Covariance.
- ~ Correlation.
- ~ Regression.
- ~ Conjoint & Discriminant Analysis.
- ~ Discrete Uniform Distribution.
- ~ Continuous Uniform Distribution.
- ~ Binomial Distribution.
- ~ Poisson Distribution.
- ~ Normal Distribution.
- ~ Sampling Techniques.
- ~ T Distribution.
- ~ Hypothesis Testing and Confidence Intervals.
- ~ Chi Square Test and Distribution.
- ~ Bayes Theorem.

#### => Visual Analyst :

- ~ Talking about Business Intelligence.
- ~ Tools and Methodologies used in BI.
- ~ Why Visualization is getting more popular.
- ~ Why Tableau?
- ~ Gartner Magic Quadrant of Market Leaders.
- ~ Future business impact of BI.
- ~ Let's Explore
- ~ Tableau Products.
- ~ Tableau Architecture.
- ~ BI Project Execution.
- ~ Tableau Installation in local system.
- ~ Introduction to Tableau Prep.
- ~ Tableau Prep Builder User Interface.
- ~ Data Preparation techniques using Tableau Prep Builder tool.
- ~ How to connect Tableau with different data source.
- ~ Visual Segments.
- ~ Visual Analytics in depth.
- ~ Filters, Parameters & Sets.
- ~ Tableau Calculations using functions.
- ~ Tableau Joins.
- ~ Working with multiple data source (Data Blending).
- ~ Building Predictive Models.
- ~ Dynamic Dashboards and Stories.
- ~ Sharing your Reports.
- ~ Tableau Server.
- ~ User Security.
- ~ Scheduling.
- ~ PDF File.
- ~ JSON File.
- ~ Spatial File.
- ~ Statistical File.
- ~ Microsoft SQL Server.
- ~ Salesforce.
- ~ AWS.
- ~ Azure.
- ~ Google Analytics.
- ~ R.
- ~ Python.
- ~ Hadoop.
- ~ OneDrive.
- ~ Microsoft Access.
- ~ SAP HANA.
- ~ SharePoint.
- ~ Snowflake.
- ~ Subject.
- ~ Planning.
- ~ Pen & Paper approach.
- ~ Tools.
- ~ Color theme.
- ~ Shapes.
- ~ Fonts.
- ~ image Selection.
- ~ text position.
- ~ visual placing.
- ~ Story layout & design.
- ~ Dashboard planning.
- ~ Power BI introduction and overview.
- ~ Key Benefits of Power BI.
- ~ Power BI Architecture.
- ~ Power BI Process.
- ~ Components of Power BI.
- ~ Power BI Building Blocks.

- ~ Power BI vs other BI tools.
- ~ Power Installation.
- ~ Overview of Power BI Desktop.
- ~ Data Sources in Power BI Desktop.
- ~ Connecting to a data Sources.
- ~ Query Editor in Power BI.
- ~ Views in Power BI.
- ~ Field Pane.
- ~ Visual Pane.
- ~ Custom Visual Option.
- ~ Filters.
- ~ Introduction to using Excel data in Power BI.
- ~ Exploring live connections to data with Power BI.
- ~ Connecting directly to SQL Azure, HD Spark, SQL Server Analysis Services/ My SQL.
- ~ Introduction to Power BI Development API.
- ~ Import Power View and Power Pivot to Power BI.
- ~ Power BI Publisher for Excel.
- ~ Content packs.
- ~ Introducing Power BI Mobile.
- ~ Power Query Introduction.
- ~ Query Editor Interface.
- ~ Clean and Transform your data with Query Editor.
- ~ Data Type.
- ~ Column Transformations vs Adding Columns.
- ~ Text Transformations.
- ~ Cleaning irregularly formatted data Transpose.
- ~ Date and Time Calculations.
- ~ Advance editor: Use Case.
- ~ Query Level Parameters.
- ~ Combining Data Merging and Appending.
- ~ Data Modelling.
- ~ Calculated Columns.
- ~ Measures/New Quick Measures.
- ~ Calculated Tables.
- ~ Optimizing Data Models.
- ~ Row Context vs Set Context.
- ~ Cross Filter Direction.
- ~ Manage Data Relationship.
- ~ Why is DAX important?
- ~ Advanced calculations using Calculate functions
- ~ DAX queries.
- ~ DAX Parameter Naming.
- ~ Time Intelligence Functions.
- ~ Types of visualization in a Power BI report.
- ~ Custom visualization to a Power BI report.
- ~ Matrixes and tables.
- ~ Getting started with color formatting and axis properties.
- ~ Change how a chart is sorted in a Power BI report.
- ~ Move, resize, and pop out a visualization in a Power BI report.
- ~ Drill down in a visualization in Power BI.
- ~ Drill Through.
- ~ Histograms.
- ~ Basic Area chart.
- ~ Combo Chart in Power BI.
- ~ Customize visualization title, background, and legend.
- ~ Doughnut charts in Power BI.
- ~ Scatter Charts in Power BI.
- ~ Funnel charts in Power BI.
- ~ KPI Visuals.
- ~ Radial Gauge charts in Power BI.
- ~ Bookmarks in Power BI.
- ~ Slicers in Power BI.
- ~ Filters.
- ~ Report Level Parameters.
- ~ Z Order.
- ~ Waterfall charts in Power BI.
- ~ Create a Power BI dashboard.
- ~ Dashboard tiles in Power BI.
- ~ Pin a tile to a Power BI dashboard from a report.
- ~ Pin an entire report page to a Power BI dashboard.
- ~ Data alerts in Power BI service.
- ~ Add an image, text box, video, hyperlink or web code to your dashboard.
- ~ Configuring a Dashboard.
- ~ Power BI Q&A.
- ~ Display a dashboard tile in Focus mode.
- ~ Power BI embedded.
- ~ Row Level Security in Power BI.
- ~ Report Server Basics.
- ~ Refresh a dataset created from a Power BI Desktop file local.
- ~ Refresh a dataset created from a Power BI Desktop file cloud.
- ~ Web Portal.
- ~ Paginated Reports.
- ~ Data Gateways.
- ~ Scheduled Refresh.
- ~ Resources (Rest API/ SOAP APIs/ URL Access).
- ~ R Integration in Power BI Desktop.
- ~ R Powered Custom Visuals.
- ~ Creating R visuals in Power BI.

- ~ *R Visuals in Power BI Service.*
- ~ *R Scripts Security.*
- ~ *Creating visual using Python.*

=> Predictive Analytics :

- ~ *Machine Learning*
- ~ *Deep Learning*

=> Descriptive Analytics :

- ~ *EDA*

### Project details :-

=> Python for Data Analytics :

- ~ *Stock Market Analysis.*
- ~ *House prices : Advanced Regression Techniques.*
- ~ *Election Analysis.*

=> SQL FOR DATA ANALYTICS :

- ~ *Ecommerce analysis Tableau integration.*
- ~ *Sales Data Analysis Tableau integration.*

=> Data wrangling with Excel :

- ~ *E Commerce Customer Analysis.*
- ~ *Project Management Dashboard.*
- ~ *Sales Dashboard.*

# Explainable AI

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Explainable-AI>

## Course Description :-

Explainable AI

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => All about XAI
- => Explaining AI with Python
- => West Nile virus a case of life or death
- => Explaining Machine Learning with Facets

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Rishav Dash :

~ This is Rishav Dash. I am a Jr. Data Scientist and mentor at INeuron.ai with working experience in computer vision, natural language processing, Machine Learning, and AOps. Hands-on experience leveraging machine learning, deep learning, transfer learning models to challenging real-world problems, and building products to solve peoples problems.

## Curriculum details :-

=> Explainable AI :

- ~ Introduction to Explainable AI (XAI)
- ~ All about XAI
- ~ Explaining AI with Python
- ~ West Nile virus a case of life or death
- ~ XAI can save lives using Google Location H
- ~ Explaining Machine Learning with Facets
- ~ Microsoft Azure ML Model Interp SHAP
- ~ SHAP Implementation
- ~ Building XAI solution from scratch
- ~ AI fairness with Google\_s What-if-Tool(WIT)
- ~ Local Interpretable Model-Agnostic Explanation(LIMEI)
- ~ The END

# Pre Ethical Hacking Community Class

---

Topic Name : CYBER SECURITY

Sub-topic Name : CYBERSECURITY MASTERS

Course link : <https://ineuron.ai/course/Pre-Ethical-Hacking-Community-Class>

## Course Description :-

Worried about getting into Cyber Security Career, if you are from Non-technical background or even technical background and want to switch your career to Cyber Security. So join with me in community class to boost your future proof career in Cyber Security.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Information Gathering
- => Scanning Networks
- => Gaining Access
- => Malware Introductory
- => Maintaining Access
- => Be Anonymous on Internet
- => Ethical Hacker in Enterprise
- => Protocols and Roles for Ethical Hacker in an Organisation
- => Case Studies on Project at Organization
- => Bug Bounty is good career options or not?

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Mukesh Kumar Rao :

~ Cyber Security Consultant and Instructor at iNeuron. Cyber Security Consultant at Cyber Security Society. Cyber Security Career Consultant at Expertrons. Cyber Security researcher and Bug Bounty at Hackerone and Bugcrowd. Instructor at Udemy. Cyber Security Consultant and Coach | AWS Builder Community | AZURE | AWS Solution Associate | AzDev Lead Chapter | CEHv11, having 3+ years of experience with trained over 10k+ Students in Cyber Security and Cloud Technologies.

## Curriculum details :-

- => Day 1 :
  - ~ Ethical Hacking
- => Day 2 :
  - ~ Computer Networking
- => Day 3 :
  - ~ Linux OS Overview
- => Day 4 :
  - ~ Be Anonymous on Internet
- => Day 5 :
  - ~ Linux Basic Practice
- => Day 6 :
  - ~ Overview on Windows Security Features
- => Day 7 :
  - ~ Type of Hackers
- => Day 8 :
  - ~ Type of Testing
- => Day 9 :

~ *Phases of Hacking*

=> Day 10 :

~ *Information Gathering*

=> Day 11 :

~ *Scanning Networks*

=> Day 12 :

~ *Gaining Access*

=> Day 13 :

~ *Malware Introductory*

=> Day 14 :

~ *Maintaining Access*

=> Day 15 :

~ *Be Anonymous on Internet*

=> Day 16 :

~ *Ethical Hacker in Enterprise*

=> Day 17 :

~ *Protocols and Roles for Ethical Hacker in an Organisation*

=> Day 18 :

~ *Case Studies on Project at Organization*

=> Day 19 :

~ *Bug Bounty is good career options or not?*

=> Day 20 :

~ *Resume Discussion and Templates*

=> Day 21 :

~ *Interview Questions Freshers level, Medium Level and Advanced*



# Microsoft PowerApps

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Microsoft-PowerApps>

## Course Description :-

This program is specialized in providing theoretical as well as practical understanding of the microsoft power apps platform which allows you to learn customized app building process using microsoft powerapps. Course curriculum includes powerapps features, datasources, functions, common data services, model driven apps and much more!

## Course Features :-

- => Source Code
- => Downloadable resources
- => Quiz Questions
- => Completion Certificate

## What you will learn :-

- => Microsoft PowerApps in details
- => Account Creation
- => Dashboard Overview
- => Create your first app using Datasource
- => Understanding of Data Cards
- => Building a PowerApp using premade Microsoft template
- => Various user functions & formulas in PowerApps
- => Common Data Service
- => Model Driven Apps

## Requirements :-

- => No prior knowledge in apps
- => System with good internet connection
- => Interest to learn
- => Your dedication
- => Basic knowledge of apps

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Course Introduction :
  - ~ Syllabus overview Preview
  - ~ Microsoft PowerApps in details Preview
  - ~ Powerapp users Preview
- => Account Creation :
  - ~ How to create Microsoft PowerApps account?
  - ~ Create account without work or school email
- => Dashboard Overview :
  - ~ Dashboard Overview
- => Create your first app using Datasource :
  - ~ How to connect data from your Google sheet to PowerApps?
  - ~ Save your application
- => Understanding of Data Cards :
  - ~ What are Data Cards in PowerApps?
- => Building a PowerApp using premade Microsoft template :
  - ~ Introduction
  - ~ Customize the app
  - ~ edit the data
- => Various user functions & formulas in PowerApps :
  - ~ Introduction

- ~ *Display a clock that updates in real time*
- ~ *Grouping Elements/Launch function*
- ~ *Dropdowns*

=> Common Data Service :

- ~ *Common Data Service*

=> Model Driven Apps :

- ~ *model driven app*

=> PowerApps Portals :

- ~ *Create first portal app*

# Power BI Course

---

Topic Name : DATA ANALYTICS

Sub-topic Name : POWER BI

Course link : <https://ineuron.ai/course/Power-BI-Course>

## Course Description :-

Learn why Power BI delivers a comprehensive collection of Business Intelligence tools for your data analysis needs, and how to utilise these tools to do all of the aforementioned activities and more in this course. It's a fantasy to be able to organise your data in a matter of minutes, effortlessly add computations to it, and then generate and share beautiful charts from the data.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Tables and Matrix in Power BI
- => Working with Maps
- => Cards and Filters
- => Power Query

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Pawan Lalwani :

~ Pawan is a highly skilled and self motivated trainer who has expertise in various business intelligence tools like Power BI, Tableau and Microsoft Excel. He comes with 10 years of experience in training individuals in different industry sectors like Banking, Finance, Healthcare, IT, Automobile, Manufacturing and Pharmaceutical.

## Curriculum details :-

=> Introduction :

- ~ 1.1 Introduction to Power BI
- ~ 1.2 Download, Install and Update Features in Power BI

=> Basic Charts in Power BI :

- ~ 2.0 Basic Charts in Power BI Desktop
- ~ 2.1 Column Chart in Power BI
- ~ 2.2 Stacked Column Chart in Power BI
- ~ 2.3 Pie Chart in Power BI
- ~ 2.4 Donut Chart in Power BI
- ~ 2.5 Funnel Chart in Power BI
- ~ 2.6 Ribbon Chart
- ~ 2.7 Include and Exclude
- ~ 2.8 Export data from Visual

=> Working with Maps :

- ~ 3.0 Maps in Power BI
- ~ 3.1 Creating a Map in Power BI
- ~ 3.2 Filled Map
- ~ 3.3 Map with Pie Chart
- ~ 3.4 Formatting in Map
- ~ 3.5 Change Background in Map
- ~ 3.6 Map of India in Power BI
- ~ 3.7 Map of Australia in Power BI

=> Tables and Matrix in Power BI :

- ~ 4.0 Table and Matrix in Power BI
- ~ 4.1 Creating a Table in Power BI
- ~ 4.2 Formatting a Table
- ~ 4.3 Conditional Formatting in Table
- ~ 4.4 Aggregation in Table
- ~ 4.5 Matrix in Power BI

- ~ 4.6 Conditional Formatting in Matrix
- ~ 4.7 Hierarchies in Matrix
- ~ 4.8 Sub-Total and Total in Matrix
- ~ 4.9 Number Formatting in Table

#### => Other Charts in Power BI :

- ~ 5.0 Other Charts in Power BI
- ~ 5.1 Line Chart in Power BI
- ~ 5.2 Drill Down in Line Chart
- ~ 5.3 Area Chart in Power BI
- ~ 5.4 Line vs Column Chart in Power BI
- ~ 5.5 Scatter Plot in Power BI
- ~ 5.6 Waterfall Chart in Power BI
- ~ 6.7 TreeMap in Power BI
- ~ 5.8 Guage Chart in Power BI

#### => Cards and Filters :

- ~ 6.0 Cards and Filters in Power BI
- ~ 6.1 Number Card
- ~ 6.2 Text Card
- ~ 6.2.1 Formatting of Text Card
- ~ 6.3 Date Card
- ~ 6.3.1 Date Card (Relative Filtering)
- ~ 6.4 Multi-Row Card
- ~ 6.5 Filter on Visual
- ~ 6.6 Filter on This Page
- ~ 6.7 Filter on All Pages
- ~ 6.8 Drillthrough in Power BI

#### => Slicers in Power BI :

- ~ 7.0 Slicers in Power BI
- ~ 7.1 Text Slicers in Power BI
- ~ 7.2 Formatting a Text Slicer
- ~ 7.3 Date Slicers in Power BI
- ~ 7.4 Formatting a Date Slicer
- ~ 7.5 Number Slicers in Power BI

#### => Advanced Charts in Power BI :

- ~ 8.0 Advanced Charts in Power BI
- ~ 8.1 Animated Bar Chart Race
- ~ 8.2 Drill down donut Chart
- ~ 8.3 Drill down Column chart
- ~ 8.4 Word Cloud in Power BI
- ~ 8.5 Sankey Chart in Power BI
- ~ 8.6 Infographic in Power BI
- ~ 8.7 Play Axis in Power BI
- ~ 8.8 Scroller in Power BI
- ~ 8.9 Sunburst Chart in Power BI
- ~ 8.10 Histogram in Power BI

#### => Objects in Power BI :

- ~ 9.1 Insert Image in Power BI
- ~ 9.2 Insert Text in Power BI
- ~ 9.3 Insert Shapes in Power BI
- ~ 9.4 Insert Buttons in Power BI
- ~ 9.5 Web URL Action in Power BI
- ~ 9.6 Page Navigation Action in Power BI
- ~ 9.7 Bookmark Action in Power BI
- ~ 9.8 Drillthrough Action in Power BI

#### => Power BI Service Introduction :

- ~ 10.1 Create a Superstore Report in Power BI
- ~ 10.2 Create an Account on Power BI Service
- ~ 10.3 Publish Report to Power BI Service Account
- ~ 10.4 Export Power BI Report to PPT, PDF or PBIX
- ~ 10.5 Comment, Share and Subscribe to Power BI Report
- ~ 10.6 Create a Dashboard in Power BI Service
- ~ 10.7 Problem in Power BI Dashboard and its solution
- ~ 10.8 Automatic Refresh in Power BI using Gateway

#### => Power Query - Text Functions :

- ~ 11.0 Text Functions in Power Query (Power BI)
- ~ 11.1 Merge Columns in Power Query (Power BI)
- ~ 11.2 Split and Trim in Power Query (Power BI)
- ~ 11.3 Upper, Lower and ProperCase in Power Query (Power BI)
- ~ 11.4 Prefix and Suffix in Power Query (Power BI)
- ~ 11.5 Left, Right and Mid Functions in Power Query (Power BI)
- ~ 11.6 Extract Text with Delimiters

#### => Power Query - Date Functions :

- ~ 12.0 Date Functions in Power Query (Power BI)
- ~ 12.1 Year, Quarter, Month and Day Functions in Power Query (Power BI)
- ~ 12.2 Find Difference between Dates in Power Query (Power BI)
- ~ 12.3 Month and Day Name in Power Query (Power BI)
- ~ 12.4 Day, Week of Month, Year in Power Query (Power BI)
- ~ 12.5 Extract Date, Time in Power Query (Power BI)
- ~ 12.6 Calculate Age in Power Query (Power BI)
- ~ 12.7 Day of Year, Quarter, Month in Power Query (Power BI)

#### => Power Query - Number Functions :

- ~ 13.0 Number Functions in Power Query (Power BI)

- ~ 13.1 Basic Number Functions in Power Query (Power BI)
- ~ 13.2 Percentage, Percent Of, Module in Power Query (Power BI)
- ~ 13.3 Round Functions in Power Query (Power BI)
- ~ 13.4 IsEven, IsODD, Sign in Power Query (Power BI)

#### => Power Query - Append Files :

- ~ 14.1 Append multiple CSV files in a folder in Power Query (Power BI)
- ~ 14.2 Append multiple excel sheets, Tables in Power Query (Power BI)
- ~ 14.3 Append Excel sheets or Tables with different columns in Power BI
- ~ 14.4 Append multiple Excel files from a folder in Power BI
- ~ 14.5 Append different data sources in Power BI

#### => Power Query - Merge Files :

- ~ 15.0 Merge Files and Tables in Power BI
- ~ 15.1 Merge Sheets or Tables in Power Query (Power BI)
- ~ 15.2 Merge Data from multiple Excel files or Workbooks in Power BI
- ~ 15.3 Merge data from different data sources in Power Query (Power BI)
- ~ 15.4 Merge data having multiple criteria in Power BI

#### => Power Query - Conditional Columns :

- ~ 16.0 Conditional Column and Column from example in Power BI
- ~ 16.1 Column from examples in Power BI - Split Text
- ~ 16.2 Column from examples in Power BI - Merge Columns
- ~ 16.3 Column from Examples in Power BI - Date
- ~ 16.4 Column from Examples in Power BI - Alphanumeric
- ~ 16.5 Conditional Column in Power BI - One Column
- ~ 16.6 Conditional Column in Power BI - two columns
- ~ 16.7 Conditional Column in Power BI - Compare two columns
- ~ 16.8 Conditional Column in Power BI - on Dates

#### => Power Query - - Important Topics :

- ~ 17.0 Very Important Topics in Power Query (Power BI)
- ~ 17.1 Fill Down in Power BI
- ~ 17.2 Grouping in Power Query (Power BI)
- ~ 17.3 Transpose in Power Query (Power BI)
- ~ 17.4 Unpivot In Power Query (Power BI)
- ~ 17.5 Data Types in Power Query (Power BI)
- ~ 17.6 Replace Errors and Values in Power Query (Power BI)
- ~ 17.7 Keep and Remove Rows in Power Query (Power BI)
- ~ 17.8 Add, Remove and Goto Columns in Power Query (Power BI)

#### => M Language Introduction :

- ~ 18.0 M Language in Power Query
- ~ 18.1 Introduction to M Language
- ~ 18.2 IsIn Date Functions in M Language - Power BI
- ~ 18.3 Add and Subtract Date M Functions in Power BI
- ~ 18.4 Basic Date M Functions in Power BI
- ~ 18.5 Basic Text M Functions in Power BI
- ~ 18.6 Simple M Code in Power BI
- ~ 18.7 Trick to get all 900+ M Functions in Power BI

# NLP Crash Course

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP

Course link : <https://ineuron.ai/course/NLP-Crash-Course>

## Course Description :-

Natural language processing (NLP) is one of the artificial intelligence's most essential and helpful application fields. As new methodologies and toolsets combine with ever-increasing data availability, NLP is rapidly evolving. In this course, you'll learn about the core concepts of natural language processing (NLP) and how it applies to current and new technologies. You will obtain a comprehensive understanding of contemporary neural network techniques for linguistic data processing. You'll be able to progress from word representation and syntactic processing to creating and executing complicated deep learning models for question answering, machine translation, and other language understanding problems by mastering cutting-edge methodologies.

## Course Features :-

- => Quizzes
- => Assignments
- => Hands-on practicals
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => NLP important topics
- => Transfer learning mechanism
- => Real-time project implementation

## Requirements :-

- => Basic knowledge of Python programming
- => A system with stable internet connection
- => Your dedication

## Instructors :-

- => Sudhanshu Kumar :
  - ~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => NLP overview :
  - ~ NLP overview Preview
  - ~ NLP very basic
- => Word Embedding :
  - ~ TFIDF
  - ~ Word embeddings part-1
  - ~ Word embeddings part-2
- => RNN :
  - ~ RNN basic
  - ~ RNN implementation
- => LSTM :
  - ~ LSTM introduction
  - ~ GRU
- => Attention based model :
  - ~ Encoder-Decoder and Attention mechanism
  - ~ Understanding paper "Attention Is All You Need"
- => Transfer learning in NLP :
  - ~ GPT and BERT Model Preview
  - ~ SOTA model with paper discussion
  - ~ ALBERT & DistilBERT project discussion Preview
- => Project :
  - ~ Megatron project
  - ~ Brand measures project

# Chatbot Using Google Dialogflow

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Chatbot-Using-Google-Dialogflow>

## Course Description :-

Google Dialog flow Chatbot

## Course Features :-

=> Lifetime Dashboard

=> Free Course

## What you will learn :-

=> End to End Chatbot with Deployment And Custom Service Integration

=> Chatbot Fundamentals

=> Understanding The Framework

=> Business Problem- Let's Build The Chatbot

=> Building The Solution

=> Implementing The Fulfillment

=> Integration With Telegram

=> Summary And Further Work

## Requirements :-

=> Programming Understanding

=> NLP understanding

## Curriculum details :-

=> End to End Chatbot with Deployment And Custom Service Integration

=> Chatbot With Google Dialogflow- Chatbot Fundamentals :

~ Introduction to chatbot Preview

=> Chatbot With Google Dialogflow- Understanding The Framework

=> Chatbot With Google Dialogflow- Business Problem- Let's Build The Chatbot

=> Chatbot With Google Dialogflow- Building The Solution

=> Chatbot With Google Dialogflow- Implementing The Fulfillment

=> Chatbot With Google Dialogflow- Integration With Telegram

=> Chatbot With Google Dialogflow- Summary And Further Work

# Reasoning Live Class

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Reasoning-Live-Class>

## Course Description :-

This Course will help you to develop important skills you use during all kinds of daily situations. It will help you make important decisions, apprehend the truth, and solve problems. Logical reasoning is also used to measure intelligence during an IQ test. Since reasoning abilities affect how people perceive, analyse, and accept arguments and facts, it has a significant effect on one's ability to learn from new information and experiences. Reasoning qualities are also essential for developing and maintaining opinions and beliefs that are compatible with and supported by various knowledge.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Alphabet Test
- => Analogy
- => Arithmetical Reasoning
- => Blood Relations
- => Calendar and Clock Test
- => Classification
- => Coding-Decoding
- => Cubes and Dices Test
- => Data Sufficiency
- => Images
- => Mathematical Operations
- => Non-Verbal Series
- => Number-Ranking-Time Sequence
- => Puzzle Test
- => Series Completion
- => Statements
- => Syllogism
- => Verbal Reasoning

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => REASONING :
  - ~ *Alphanumeric Series*
  - ~ *Reasoning Analogies*
  - ~ *Blood Relations*
  - ~ *Calendars*
  - ~ *clocks*
  - ~ *Cause and Effect*
  - ~ *Coding-Decoding*
  - ~ *Data Sufficiency*
  - ~ *Critical path*
  - ~ *Cubes and cuboids*



- ~ *Decision Making*
- ~ *Dices*
- ~ *Deductive Reasoning/Statement Analysis*
- ~ *Directions*
- ~ *Mirror and Water Images*
- ~ *Embedded Images*
- ~ *Figure Matrix*
- ~ *Puzzles*
- ~ *Order and Ranking*
- ~ *Odd One Out*
- ~ *Picture Series and Sequences*
- ~ *Paper Folding*
- ~ *Seating Arrangement*
- ~ *Pattern Series and Sequences*
- ~ *Statement and Assumptions*
- ~ *Statement and Conclusion*
- ~ *Shape Construction*
- ~ *Syllogism*

# Live Virtual Interview

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/Live-Virtual-Interview>

## Course Description :-

Interview for Freshers, experienced, not ID domain candidate

## Course Features :-

=> Lifetime Dashboard

=> Free Course

=> Interview Questions

## What you will learn :-

=> How to prepare for Interview

## Requirements :-

=> no prerequisite

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Live- Data Science Virtual Interview By Krish And Sudhanshu-Part 1 :

~ Virtual interview Preview

=> Live Virtual Interview For Data Science By Krish And Sudhanshu Part 2

=> Live Virtual Interview For Internship For College Student By Krish And Sudhanshu

=> Live Virtual Interview For Data Science By Krish And Sudhanshu

=> Live Transition Story Of Civil Engineer To Data Scientist With 2 Years Gap

=> Live Virtual Nervous Interview Of Mechanical Engineer For Data Science

=> Live Data Science Q&A With Krish And Sudhanshu- Give Away ML for Deployment+Internships For Women

=> Live Interview Of Lakshay For Data Science- Commerce And Statistics Background

=> Live Virtual Interview For Data Science From Teaching Assistant To Data Scientist

=> Live Virtual Interview For Data Science- Background Applied Geology From IIT Kharagpur

=> Live -Virtual Interview Of Fresher For Data Science - Session 6

# NextJS Full stack crash course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : NEXTJS

Course link : <https://ineuron.ai/course/NextJS-Full-stack-crash-course>

## Course Description :-

This course will help you to grab the fundamentals of NextJs.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Nextjs full stack application
- => Creating models in Nextjs
- => Connecting to database in nextjs
- => Create REST API in nextjs part 1
- => Create REST API in nextjs part 2
- => Testing REST API in nextjs
- => Global layouts in nextjs
- => Server side rendering SSR in nextjs
- => Create a new hero in nextjs
- => Routing and SSR in nextjs
- => Edit a superhero in nextjs

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> NextJS :

- ~ Nextjs full stack application
- ~ Creating models in Nextjs
- ~ Connecting to database in nextjs
- ~ Create REST API in nextjs part 1
- ~ Create REST API in nextjs part 2
- ~ Testing REST API in nextjs
- ~ Global layouts in nextjs
- ~ Server side rendering SSR in nextjs
- ~ Create a new hero in nextjs
- ~ Routing and SSR in nextjs
- ~ Edit a superhero in nextjs

# Getting started with Cloud

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Getting-started-with-Cloud>

## Course Description :-

This course will assist students in grasping the basic concepts of cloud computing with its various applications. This course will equip students with the proper start that they require to begin their career in cloud computing in today's modern environment, where cloud engineers are in high demand. This hands-on practical-oriented course will enable students to apply their cloud computing skills and help them in starting with a lucrative career in cloud computing.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Cloud computing fundamentals
- => Types of Clouds
- => Risks
- => Features of Cloud Computing
- => Virtualization
- => Characteristics of Cloud
- => Cloud Computing Architecture

## Requirements :-

- => Interest to learn
- => Dedication
- => System with good internet connection

## Curriculum details :-

- => Course Introduction :
  - ~ Course introduction
  - ~ Who is this course for?
  - ~ Course overview & course outcome
  - ~ Course Pre-requisite
  - ~ What is Cloud?
  - ~ What is Cloud computing?
  - ~ Why cloud is important?
  - ~ History of Cloud computing
- => Assignment 1 :
  - ~ What are the benefits of Cloud?
  - ~ Write down the benefits of cloud computing?
- => Basic Cloud concepts :
  - ~ Types of Cloud
  - ~ What is Public cloud?
  - ~ What are the different types of Public cloud?
  - ~ What is a Private cloud?
  - ~ What are the different types of Private cloud?
  - ~ What is Hybrid cloud?
  - ~ What are the different types of Hybrid cloud?
  - ~ What is Community cloud?
  - ~ What are the different types of Community cloud?
  - ~ Service Models
  - ~ What is Infrastructure as a Service (IaaS)
  - ~ What is Platform as a Service (PaaS)

~ What is Software as a Service (SaaS)

=> Assignment 2 :

~ Name the clouds which are most popular?

=> Assignment 3 :

~ What is the difference between IaaS, PaaS, and SaaS?

=> Risks :

~ What is Security?

~ Why is Security important?

~ What is Privacy?

~ Why is Privacy important?

~ What is LOCK-IN?

~ Why is LOCK-IN risky?

~ What is Isolation failure?

~ What is Management Interface compromise?

~ What is Insecure or Incomplete data deletion?

=> Assignment 4 :

~ How to avoid fraudulent activities on the cloud?

=> Virtualization :

~ What is Virtualization?

~ What is the intuition of Virtualization?

~ Types of Virtualization

~ What is Hardware virtualization?

~ What is the use of Hardware virtualization?

~ What is Operating system virtualization?

~ What is the use of the Operating system virtualization?

~ What is Server virtualization?

~ What is the use of Server virtualization?

~ What is Storage virtualization?

~ What is the use of Storage virtualization?

=> Assignment 5 :

~ How does virtualization work in Cloud computing?

=> Data Virtualization :

~ What is Data virtualization?

~ What are the advantages of Data virtualization?

~ What are the disadvantages of Data virtualization?

~ What are the uses of Data virtualization?

~ Data virtualization tools

=> Assignment 6 :

~ How well do you know about Data virtualization?

=> Characteristics :

~ Essential Characteristics

~ On-demand self-service

~ Broad network access

~ Resource pooling

~ Rapid elasticity

~ Measured service

~ Common Characteristics

~ Massive scale

~ Homogeneity

~ Virtualization

~ Low cost software

~ Resilient computing

~ Geographic distribution

~ Service orientation

~ Advanced security

=> Assignment 7 :

~ What is the difference between Essential and Common characteristics?

=> Cloud Computing-Architecture :

~ What is Front End?

~ What is Back End?

~ What is Client infrastructure?

~ What are the Applications?

~ What are the Services?

~ Software as a Service (SaaS)

~ Example of Software as a Service

~ Platform as a Service (PaaS)

~ Example of Platform as a Service

~ Infrastructure as a Service (IaaS)

~ Example of Infrastructure as a Service

~ What is the Runtime cloud?

~ What is Storage?

~ What is Infrastructure?

~ What is Management?

~ What is Security?

~ What is Internet?

=> Assignment 8 :

~ Describe the cloud computing architecture in your own words?

=> Course Summary :

~ Course outro

~ Future learning path

# Angular JS

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : ANGULAR JS

Course link : <https://ineuron.ai/course/Angular-JS>

## Course Description :-

Angular is a platform and framework for building single-page client applications using HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that you import into your apps. AngularJS is a client side JavaScript MVC framework to develop a dynamic web application. AngularJS was originally started as a project in Google but now, it is open source framework.

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Certificate
- => Assignment
- => Quiz

## What you will learn :-

- => Practical implementation of Angular JS in real world
- => End to End concepts understanding

## Requirements :-

- => Computer with Internet connectivity
- => Basic Programming understanding

## Instructors :-

- => Keshav Singh :
- ~

## Curriculum details :-

- => Introduction About Angular JS (Hindi) :
  - ~ *Introduction Preview*
- => Course Structure Breakup and Environment Setup 2 Angular JS (Hindi)
- => Components in Angular JS (Hindi)
- => Bindings In Angular (Hindi)
- => Bindings in Angular JS Part-2 (Hindi)
- => Communication Between Components - Angular JS | Hindi
- => Two Way Data Binding Angular JS | Hindi

# Spark AR Live Class

---

Topic Name : AR VR

Sub-topic Name : SPARK AR

Course link : <https://ineuron.ai/course/Spark-AR-Live-Class>

## Course Description :-

This course will get your journey started with Augmented Reality. In this course, you will learn the fundamentals of augmented reality with Spark AR. After successful completion of this, you will be able to create and deploy amazing AR effects on social media platforms like Instagram and Facebook.

## Course Features :-

- => Live instructor led classes
- => Completion certificate

## What you will learn :-

- => Introduction to AR & VR
- => Spark AR
- => Publishing filters
- => Trackers
- => Animation

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

=> Introduction :

- ~ Course Introduction
- ~ Course pre-requisites
- ~ Who is this course for?
- ~ What you will get from this course?
- ~ What is AR?
- ~ How to get access to course materials?
- ~ What career path you can follow after completion of this course?

=> System setup :

- ~ Spark AR installation
- ~ Blender installation

=> Getting Started :

- ~ What do you mean by XR?
- ~ Reality of Augmented Reality in 2022
- ~ Augmented Reality Roadmap
- ~ Difference between AR & VR
- ~ Possibilities of AR
- ~ How AR and AI are connected?

=> Spark AR :

- ~ Navigating Spark AR
- ~ Scene panel
- ~ Importing assets
- ~ Directional and Ambient light
- ~ Working with Material properties
- ~ Layers
- ~ Your first spark AR filter
- ~ Spark AR player
- ~ Testing your project on your device
- ~ Publishing your first Spark AR filter
- ~ Importing assets outside of Spark AR
- ~ Optimizing assets
- ~ Platform requirements
- ~ Working with lights
- ~ World AR effects

=> Advance SparkAR :

- ~ Deep dive: Trackers
- ~ Face assets
- ~ Tracking the face
- ~ Intuition behind face tracking
- ~ Working with plane & plane tracker
- ~ Intuition behind plane tracking
- ~ Target tracking
- ~ Intuition behind target tracking
- ~ What is semantic segmentation?
- ~ Working with segmentation
- ~ Occlusion
- ~ Project: Rotating assets around face
- ~ Importing animatable objects
- ~ Working with patch editor
- ~ Animating assets using Patch editor
- ~ Working with particle
- ~ Project: Crashing plane
- ~ Hand tracker
- ~ Project: Hand Resizable gradient filter

=> Summary :

- ~ Course Outro
- ~ Future Scope of AR



# Graph QL Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Graph-QL-Crash-Course>

## Course Description :-

This course will help you to grab the fundamentals of GraphQL.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Graph QL

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Graph QL :

~ Graph QL

=> NaN :

~ NaN

~ NaN

~ NaN

~ NaN

~ NaN

~ NaN

# Jenkins

---

Topic Name : DEVOPS

Sub-topic Name : JENKINS

Course link : <https://ineuron.ai/course/Jenkins>

## Course Description :-

Jenkins is an open source automation server that allows software developers all around the world to build, test, and deploy their code with confidence. The world's most popular open source automation server, with hundreds of plugins to let you build, launch, and automate any project.

## Course Features :-

- => Source Code
- => Downloadable resources
- => Quiz Questions
- => Completion Certificate

## What you will learn :-

- => Jenkins Installation Basics
- => Jenkins Dashboard
- => Jenkins Plugins
- => Jenkins Security
- => Jenkins Jobs
- => Freestyle Jobs
- => Pipeline Job

## Requirements :-

- => Prior Knowledge of Linux
- => System with good internet connection
- => Interest to learn
- => Your dedication

## Instructors :-

=> Ritesh Yadav :

~ Ritesh is truly passionate about data science, machine learning and DevOps in general, he likes what he does, and is keen to learn. Currently, He is working as a Jr. Data Scientist at Ineuron.ai. He also loves to Contribute to Open Source Projects, which are mainly under CNCF Landscape. Ritesh loves to work in Cloud-Native technologies and Golang ( Go ). Apart from this, Ritesh has been actively involved in the open-source community for over a year, helping many open-source DevOps tools and CNCF Projects like Porter, Meshery, Keptn, TensorFlow, and Thanos through his contributions.

## Curriculum details :-

=> What is Jenkins and CI & CD :

- ~ Course Overview, What is Jenkins, Why we need Jenkins Preview
- ~ What is CI and What is CD Preview

=> Jobs in Jenkins :

- ~ Build a Freestyle Job
- ~ Schedule a Jenkin Job
- ~ Build Triggers in Jenkins

=> Triggers and Webhook :

- ~ Parameterized Job in Jenkins
- ~ Polling Source Code using Jenkins
- ~ Building a Webhook for github

=> Maven Integration :

- ~ Deploy a SpringBoot App

=> Roles and User :

- ~ Create a New User in Jenkins
- ~ Roles in Jenkins
- ~ Jenkins Configure System

=> Notifications in Jenkins :

- ~ Notifications in Jenkins

=> Jenkins Pipeline and Jenkinfile :

- ~ Jenkin Pipelines Preview
- ~ What is Jenkin file and its Uses Preview

# OpenCV using Python

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/OpenCV-using-Python>

## Course Description :-

This course will teach the learner the fundamentals of the OpenCV library, including an overview of the library and image manipulation using OpenCV. It is a free and open-source computer vision and machine learning library. This course will introduce learners to the fundamentals of using OpenCV to explore computer vision and AI (AI). With the aid of many practical real-world image processing tasks, students who finish this course will gain hands-on expertise in image processing using OpenCV.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to OpenCV
- => Python programming
- => Image processing
- => Basics of NumPy
- => Image Manipulation
- => Color spaces
- => Projects

## Requirements :-

- => Interest to learn
- => Dedication
- => System with good internet connection

## Curriculum details :-

- => Introduction :
  - ~ Course Introduction
  - ~ Course Pre-requisites
  - ~ Who is this course for?
  - ~ What you will get from this course?
  - ~ What is Image Processing?
  - ~ How to get access to course materials?
  - ~ What career path you can follow after completion of this course?
- => Assignment 1 :
  - ~ Explain various fields where image processing is used and why with example
- => System Setup :
  - ~ Introduction and Installation on Colab
- => Assignment 2 :
  - ~ Uninstall open-cv and try to install open-cv-contrib
- => OpenCV overview :
  - ~ Importing packages
  - ~ Numpy basics
  - ~ Reading/Writing images and videos
  - ~ Argparse introduction
  - ~ Creating script to read image path from cmd and displaying it

=> Assignment 3 :

- ~ Read a RGB image in B/W mode and display its height, width and no of channels.

=> Image Basics :

- ~ What is pixel?
- ~ Overview of coordinate system
- ~ Practical: Manipulating pixels
- ~ Creating canvas and drawing lines and rectangles

=> Assignment 4 :

- ~ Draw a bullseye using OpenCV function.

=> Image processing :

- ~ Image translation
- ~ Image Rotation
- ~ Image resizing
- ~ Image Flipping
- ~ Image Cropping
- ~ Image Arithmetic
- ~ Bitwise Operations
- ~ Image splitting and merging
- ~ OpenCV colour spaces
- ~ Smoothing And Blurring
- ~ Thresholding

=> Assignment 5 :

- ~ Shift Image up and left using image translation
- ~ Shift Image up and rotate the image by 25 degrees.
- ~ Create a function to take input from user for the degree of rotation and final (w,h) of image from user.
- ~ Apply the rectangular mask on image show the images values where the mask colour is white.
- ~ Splitting multiple channels of images into separate images.
- ~ Convert BGR to HSV colour space

=> Project Explanation :

- ~ Face detection with OpenCV Cascades
- ~ Virtual Painting

=> Summary :

- ~ Course Outro
- ~ Future Scope of openCV

# Fundamentals of Electronics

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Fundamentals-of-Electronics>

## Course Description :-

This course is designed for students who are interested in electronics and want to learn the foundations of the subject as well as undertake a range of experiments. Students will get an idea of diodes, transistors, circuits and symbols, instrumentation and measurement of various electronic components, alternating current and direct current, frequencies, semiconductors, digital signals, and sensors, along with many other topics.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Electronics Basics
- => Circuits and symbols
- => Instrument and measurement
- => AC/DC
- => Frequencies
- => Semiconductor
- => Transistors
- => Digital Signal
- => Sensor Introduction
- => Projects on Electronics

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

=> Introduction :

- ~ Course Introduction
- ~ Course pre-requisites
- ~ Who is this course for?
- ~ What you will get from this course?
- ~ What do you mean by term electronics?
- ~ How to get access to course materials?
- ~ What career path you can follow after completion of this course?

=> Assignment 1: :

~ Assess the world around you and find out which devices are electronics devices.

=> Electronics :

- ~ *Electronics introduction*
- ~ *Abbreviation and symbols*
- ~ *What is electricity?*
- ~ *History of electronics*
- ~ *Moore's law*
- ~ *Electric charge*
- ~ *Conductor and insulator*
- ~ *What is current?*
- ~ *What is voltage?*
- ~ *What is electric power?*
- ~ *Discussion: Electricity required for human brain*
- ~ *Resistance and ohm's law*

=> Assignment 2: :

- ~ *Assess the world around you and find out the voltages of different things like battery, mobile, tv etc.*

=> Circuits and symbols :

- ~ *Components and their symbols*
- ~ *Voltage current characteristics*
- ~ *Series components*
- ~ *Voltage divider*
- ~ *Parallel components*
- ~ *Resistors*

=> Assignment 3: :

- ~ *Explain, how to attach multiple batteries so that they can act as one big battery. Also state different types of battery.*

=> Instrument and measurment :

- ~ *Voltmeter*
- ~ *Ammeters*
- ~ *Ohmmeters*
- ~ *The oscilloscope*

=> Assignment 4: :

- ~ *How to measure speed of sound with an Oscilloscope*

=> AC/DC :

- ~ *AC current*
- ~ *Transformers*
- ~ *Discussion: What actually happens when there is a power cut-off due to transformer*
- ~ *Diodes*
- ~ *DC current from AC*
- ~ *Capacitor*
- ~ *regulator*

=> Assignment 5: :

- ~ *Explain how a diode can be act as a switch?*

=> Frequencies :

- ~ *Clean signal and Noisy signal*
- ~ *Capacitor*
- ~ *Application : Using lowpass and highpass in speaker*

=> Assignment 6: :

- ~ *Explore different frequency hearing ranges of different animals. Also learn the frequency of Sonar.*

=> Semiconductor :

- ~ *What is semiconductor?*
- ~ *Silicon*
- ~ *N-type semiconductor*
- ~ *P-type semiconductor*
- ~ *PN Junction*
- ~ *LED*

=> Assignment 7: :

- ~ *Explain Photovoltaic Cell and where it is used with the help of example?*

=> Transistors :

- ~ *What is transistor?*
- ~ *Why transistor is very important?*
- ~ *Why we need transistor?*
- ~ *NPN and PNP intituition*
- ~ *Emmitter, Base and Collector intituition*

=> Assignment 8: :

- ~ *Draw diagram of home regulator with transistor in it.*

=> Digital signal :

- ~ *Analog vs digital signal*
- ~ *Binary numbers*
- ~ *Logic operations*
- ~ *Basic logic gates*

=> Assignment 9: :

- ~ *Explain different scenarios of using Logic gates?*

=> Sensor introduction :

- ~ *What are sensors?*
- ~ *Temperature sensors*
- ~ *Humidity sensors*
- ~ *Pressure sensors*
- ~ *Proximity sensors*
- ~ *Optical sensors*
- ~ *Lidar sensor*

=> Assignment 10: :

~ *Explain all the sensors used in mobile devices and what purpose they serve?*

=> Basic project discussion based on feedback :

- ~ *Home thermostat*
- ~ *Intelligent light bulb*
- ~ *Constant temperature system*

=> Summary :

- ~ *Course Outro*
- ~ *Future Scope of electronics*

# Digital Marketing Projects

---

Topic Name : DIGITAL MARKETING

Sub-topic Name : DIGITAL MARKETING PROJECT

Course link : <https://ineuron.ai/course/Digital-Marketing-Projects>

## Course Description :-

This course aims to explain the detailed life cycle of digital marketing projects and provide a thorough overview of all digital marketing channels and components. This program specializes in providing hands-on experience and cultivating skills to work on a real-time Problem - solution in digital marketing.

## Course Features :-

- => Case Studies
- => Live Project building
- => Report Analysis
- => Covering multiple domains
- => Downloadable Resources
- => Completion Certificate
- => Resume Discussion

## What you will learn :-

- => Customer Satisfaction for a Digital Marketing Company
- => SEM Case Study
- => A case study of Web Analytics tools
- => SEO study of websites
- => Social Media Strategies for Online Shopping Cart
- => Social Media Campaign Analysis
- => A case study of Business Landing pages

## Requirements :-

- => Prior Knowledge of digital marketing
- => A system with internet connection
- => Dedication

## Instructors :-

=> Ankur Khanna :

~ Highly-motivated, energetic and dynamic Digital Marketing Mentor and Assistant Professor having 7+ years of experience in Digital Marketing Industry.  
Strong practical knowledge of different digital marketing tools aimed at meeting the needs of diverse groups of learners.

## Curriculum details :-

- => Customer satisfaction for a digital marketing company :
  - ~ Understanding the customer's persona Preview
  - ~ Identify the business needs and problem Preview
  - ~ Identify the actual problem in the old strategy
  - ~ Identify the feasible solutions
  - ~ Research on problem solutions
  - ~ Methods, tools and techniques
  - ~ Evaluation
  - ~ Feedback
- => SEM case study and strategy for business website :
  - ~ Identify the business needs and problem
  - ~ Understanding the businesses product & services
  - ~ Study old SEM plan and strategy
  - ~ Identify the actual problem in the old strategy
  - ~ Old strategy report and result analysis
  - ~ Identify the feasible solutions
  - ~ Methods, tools and techniques
  - ~ Evaluation
  - ~ Feedback
- => Return on investment on different digital marketing techniques :
  - ~ Identify and contact different businesses
  - ~ Understanding the budget businesses spend on digital marketing
  - ~ Analysis of the cost spending and different budget strategy by businesses
  - ~ Report analysis for last few months spent on different digital techniques
  - ~ Analyse the return (ROI)
  - ~ Apply methods, tools and techniques



- ~ Evaluation
- ~ Feedback

=> Business website creation with different digital marketing techniques :

- ~ Identify and contact different businesses
- ~ Understanding the businesses product & services
- ~ Identify the business needs and problem
- ~ Understanding the budget businesses spend on website development
- ~ Analysis if business having any working website
- ~ Identify the actual problem in the old business website
- ~ Identify the feasible solutions
- ~ Select website development tool and platform
- ~ Develop business website and publish it
- ~ Understanding the budget businesses spend on digital marketing
- ~ Apply methods, tools and techniques
- ~ Evaluation
- ~ Feedback

=> SEO case study & strategy for business website :

- ~ Identify the business needs
- ~ Understanding the businesses product & services
- ~ Study old SEO plan and strategy
- ~ Identify the actual problem in the old strategy
- ~ Old strategy report and result analysis
- ~ Identify the feasible solutions
- ~ Methods, tools and techniques
- ~ Evaluation
- ~ Feedback

=> Facebook for business( business website and business page ) :

- ~ Identify the business needs/work as a startup business
- ~ Understanding the businesses product & services
- ~ Identify if the business has an old Facebook business page
- ~ Identify the actual problem in the old Facebook business page
- ~ Understanding the budget businesses spend on Facebook page marketing
- ~ Report analysis for last few months spend on Facebook ads and campaign
- ~ Create a Facebook business page and publish it
- ~ Planning for the budget strategy businesses spend on Facebook page marketing.
- ~ Understand the Facebook ad manager
- ~ Work on a different Facebook ad campaign
- ~ Apply methods, tools and techniques
- ~ Evaluation
- ~ Feedback

# Flutter

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : FLUTTER

Course link : <https://ineuron.ai/course/Flutter>

## Course Description :-

Learn how to use Flutter, Google's latest mobile framework, to develop quick and beautiful mobile apps. With no prior expertise, you will rapidly learn how to construct any application with Flutter in this course. upon successful completion of this course, you will be able to create interactive and responsive applications using the flutter development kit.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Stateless widgets
- => Stateful widgets
- => Background changers
- => Randomization
- => Camera and gestures
- => API handling
- => SQLite
- => Firebase
- => BLOC in flutter

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to Flutter :
  - ~ Introduction to flutter
- => Installation of Flutter :
  - ~ Flutter installation on MAC
  - ~ Flutter doctor - Do not panic
  - ~ Flutter installation for Windows
  - ~ Installing plugins for VSCode
- => First Flutter project :
  - ~ Creating the first project in flutter
  - ~ Run your flutter project
  - ~ Actual hello world
- => Stateless and Stateful - 2 projects :
  - ~ Stateless and Stateful Widgets
  - ~ How to read flutter documentation
  - ~ Scaffold widget in flutter
  - ~ A stateless app in flutter
  - ~ Multi child layout in flutter
  - ~ Raised buttons in flutter
  - ~ Converting into stateless widget
  - ~ Making a stateful app

- ~ *Designing Visual part*
- ~ *Finishing stateful number app*

=> Background Changer and randomization :

- ~ *What we will create in this section*
- ~ *Stateless widget work*
- ~ *Random value generation in flutter*
- ~ *Button properties in bgchanger*

=> Dice Roller App project :

- ~ *Getting assets for dice roller flutter*
- ~ *create main dart in dice app*
- ~ *logic part of dice roller*
- ~ *Design of dice roller and assignment*

=> Tic Tac Toe App in flutter :

- ~ *Getting started with TicTacToe in flutter*
- ~ *taking main dart file for TicTacToe*
- ~ *Initialize state for Tic Tac Toe*
- ~ *playgame and reset game tictactoe*
- ~ *Winning logic for Tictactoe in flutter*
- ~ *Understand gridview in flutter*
- ~ *Final design of TicTacToe and assignment*

=> Scratch and win App in flutter :

- ~ *Scratch and win assets in flutter*
- ~ *scratch and win todos*
- ~ *rest game and lucky number in flutter*
- ~ *showall and gameplay*
- ~ *Finishing scratch and win in flutter*

=> Spanish Audio number app :

- ~ *Third party libraries in flutter*
- ~ *Audio helper in flutter*
- ~ *defining audio methods in spanish app*
- ~ *Finishing spanish number app in flutter*

=> Camera and Gesture :

- ~ *Reading gesture docs*
- ~ *Preparing project with imagepicker*
- ~ *methods for camera and gallery and ios fix*
- ~ *Finishup camra app in flutter*

=> Navigation and keys in flutter - 2 apps :

- ~ *Reading assignment for drawer and keys*
- ~ *Creating catogory page*
- ~ *Routing basics in flutter*
- ~ *drawer links and navigation*
- ~ *drawer app assignment*
- ~ *A signup app*
- ~ *bring in logo assets*
- ~ *Design your first input field*
- ~ *Global keys and validator*
- ~ *Collect key values in next screen*
- ~ *HomePage and assignment*

=> API handling in flutter - 2 Apps :

- ~ *introduction to API in flutter*
- ~ *passing key in stateful*
- ~ *Making a web request with Future*
- ~ *storing web response*
- ~ *getting data on screen and debug*
- ~ *Understand the API response*
- ~ *Fetching data with web*
- ~ *picking up data from JSON*
- ~ *Run the app and assignment*

=> Sqlite - A publishable App :

- ~ *Before we start this project*
- ~ *Reading the docs for database*
- ~ *adding dependencies for database*
- ~ *custom notes class part 1*
- ~ *custom notes class part 2*
- ~ *Start with database helper file*
- ~ *create table in sqlite*
- ~ *insert update and delete query*
- ~ *Get value count from database*
- ~ *creating semi list screen*
- ~ *rewriting stateful widget of detail class*
- ~ *saving notes and helper method*
- ~ *delete and UI part of details screen*
- ~ *Adding methods in listview*
- ~ *Finally done with this app*

=> Firebase and flutter - Authentication :

- ~ *Firebase for flutter*
- ~ *exploring firebase*
- ~ *Configure iOS and Android app for firebase*
- ~ *Define router in main*
- ~ *add lister to check state of login*
- ~ *Signin with email and password*
- ~ *Android X bug and signin UI*

- ~ Home page logic methods
- ~ Home page UI and link for signup
- ~ logic part of singup page
- ~ Final one on authentication

#### => Firebase Database and Storage :

- ~ getting started with database and storage
- ~ stackoverflow and file structure
- ~ creating model for contact
- ~ upgrading to AndroidX and homePage
- ~ uploading image and data to firebase
- ~ UI for add screen
- ~ UI for home page and bug fix
- ~ Get values from snapshot
- ~ Phone and sms intent launch
- ~ Delete contact from firebase
- ~ Edit screen - passing id
- ~ upload new photo in editcontact
- ~ final touch to database app - contact

#### => UI Challenge - WhatsApp :

- ~ Create whatsapp project and exercise files
- ~ reusable widgets
- ~ creating whatsapp title bar
- ~ creating tab bar
- ~ creating chat UI page
- ~ Design calls ui
- ~ Status screen ui

#### => BLOC in flutter :

- ~ What is BLOC in flutter
- ~ Creating a flutter block app - structure
- ~ Creating BLOC pattern code in flutter
- ~ Creating UI for BLOC project and calls

# Full Stack Data Science Feb'21 Tech Neuron

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science-Feb'21-Tech-Neuron>

## Course Description :-

This is a full stack data science self-paced course with recordings of live mentor-led certification program and a full-time one-year internship provided by iNeuron intelligence private limited, where you will learn all the stack required to work in the data science, data analytics, and big data industries, including machine learning operations and cloud infrastructure, as well as real-time industry project and product development with the iNeuron product development team, and you will contribute on various levels.

## Course Features :-

- => Full stack Data Science Recorded Lectures
- => One year of internship Anytime
- => 56 + hands-on industry real-time projects.
- => 500 hours live interactive classes.
- => Lifetime Dashboard access
- => Assignment in all the module

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => Cloud
- => Architecture
- => Databases

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sunny Savita :

~ I'm an AI enthusiast, graduate in Computer science and engineering. Currently working with iNeuron.ai as a Data Scientist and having 2+ years of experience. I have skills in big data, machine learning, computer vision, Natural language processing. My expertise also includes project design development and implementation with AIOps tools.

## Curriculum details :-

- => Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

#### => Python Fundamentals :

- ~ Python Basic
- ~ String, List, Indexing
- ~ Tuple, Set & Dict
- ~ If, Else & For Loop
- ~ For Loops & While loops
- ~ Python Program Discussion in loops
- ~ Function Part - 1
- ~ Function Part - 2

#### => Advanced Python :

- ~ Iterator Generator & File System
- ~ Exception handling Class 1 part 1
- ~ Exception handling Class 1 part 2
- ~ Exception handling Class 2
- ~ Module & Packages
- ~ OOPS Part 1
- ~ OOPS Part 2
- ~ OOPs Concepts - Polymorphism

#### => Working with Databases & Python :

- ~ SQL Part 1
- ~ SQL Part 2
- ~ OOPS Discussion
- ~ Introduction to MongoDB
- ~ Working with Python & MongoDB Part1
- ~ Working with Python & MongoDB Part2
- ~ SQL lite, map, reduce, filter, zip

#### => Working with Pandas & Numpy :

- ~ Introduction to Pandas
- ~ Working with Pandas
- ~ Pandas Data Analysis Part 1
- ~ Pandas Data Analysis Part 2
- ~ Pandas and Numpy
- ~ Numpy methods

#### => GUI Programming :

- ~ GUI Programming with Tkinter

#### => Working with Graphs & Charts :

- ~ Introduction to Graphs & Charts
- ~ Working with Graphs in Python

#### => API :

- ~ API Testing

#### => Python Projects :

- ~ Flask End-to-End Project
- ~ Review Scrapper
- ~ Image Scrapper and deployment on Heroku, AWS, and Azure

#### => Statistics :

- ~ Introduction to Stats - Day 1
- ~ Stats - Day 2
- ~ Extra doubt session
- ~ Stats - Day 3
- ~ Stats - Day 4
- ~ Stats - Day 5

#### => EDA & Feature Engineering :

- ~ Introduction to EDA
- ~ Doubt Clearing session
- ~ EDA and Feature Engineering

#### => Machine Learning :

- ~ Linear Regression
- ~ Ridge Lasso Regression, Elastic & Logistic Regression
- ~ Naive Bayes Algorithm and practical implementation of Ridge Lasso and Logistic Regression
- ~ Logistic Practical, SVM, SVR
- ~ Decision Tree Classification
- ~ Random Forest & SVM
- ~ Adaboost
- ~ Gradient Boosting
- ~ Clustering
- ~ Introduction to Machine learning
- ~ Linear Regression
- ~ Linear Regression live coding demonstration part-1
- ~ Linear Regression live coding demonstration part-2
- ~ Project Admission Prediction, Lasso, Ridge & Elastic Net
- ~ Project deployment in Heroku, Azure & AWS
- ~ Logistic Regression
- ~ Logistic Regression implementation
- ~ Decision Tree
- ~ Decision Tree Part 2, Ensemble Tech, Random Forest & Boosting
- ~ KNN and SVM
- ~ Decision Tree Practical Implementation
- ~ Decision Tree Live Coding & Grid Search
- ~ Grid Search, Bagging Classifier & Random Forest

- ~ KNN, SVC, SVR & Stacking
- ~ Clustering
- ~ Clustering and PCA
- ~ PCA practical, DBSCAN and Naive Bayes
- ~ XG Boost, NLTK & TF-IDF

#### => Machine Learning End to End Project :

- ~ Machine learning project
- ~ Machine learning project
- ~ ML End to End project Pipeline Explanation
- ~ ML Project Explanation along with GitHub and Docker
- ~ Machine Learning Pipelines Live Coding Part-1
- ~ Machine Learning Pipelines Live Coding Part-2
- ~ 2nd July Live Class
- ~ Machine Learning Pipelines Live Coding Part-2
- ~ Revision Class
- ~ Model training, evaluation and push
- ~ Model training, evaluation and push
- ~ Revision

#### => PCA in ML :

- ~ PCA
- ~ PCA Implementation

#### => NLP for Machine Learning :

- ~ NLP in ML
- ~ Spam Classification

#### => Time Series Analysis :

- ~ Introduction to Time Series
- ~ Time Series Implementation

#### => Stats :

- ~ Introduction
- ~ Different types of Statistics
- ~ Population vs Sample
- ~ Mean, Median and Mode
- ~ Variance, Standard Deviation
- ~ Sample Variance why  $n-1$
- ~ Standard Deviation
- ~ Variables
- ~ Random Variables
- ~ Percentiles & quartiles
- ~ 5 number summary
- ~ Histograms
- ~ Gaussian - Normal distribution
- ~ Standard Normal distribution
- ~ Application Of Zscore
- ~ Basics Of Probability
- ~ Addition Rule In Probability
- ~ Multiplication rule in probability
- ~ Permutation
- ~ Combination
- ~ Log Normal Distribution
- ~ Central Limit theorem
- ~ Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median And Mode
- ~ Covariance
- ~ Pearson And Spearman Rank Correlation
- ~ What is P-Value?
- ~ What is Confidence Interval?
- ~ How To Perform Hypothesis Testing - Confidence Interval Z Test Statistics Derive Conclusion
- ~ Hypothesis testing part 1
- ~ Hypothesis testing part 2
- ~ Finalizing statistics

#### => ML Projects :

- ~ Detailed Project Report explanation
- ~ Project:- Wafer Fault Detection Part 1
- ~ Project:- Wafer Fault Detection Part 2
- ~ Deployment in Heroku using docker and CircleCI

#### => ML Project 1 :- Fault detection in wafers based on sensor data :

- ~ Introduction
- ~ The problem statement and Data Description
- ~ The Application Flow
- ~ Ingestion and Validation Part1
- ~ Validation Part2
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 2 :- Cement Strength Prediction :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code Intro and Logging
- ~ Validation and Transformation
- ~ DB Operations

- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

=> ML Project 3 :- Credit Card Defaulters :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Deployment

=> ML Project 4 :- Forest Cover :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

=> ML Project 5 :- Income Prediction :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

=> ML Project 6 :- Insurance Fraud Detection :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code Intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment
- ~ The Problem Statement and Data Description

=> ML Project 7 :- Mushroom Classification :

- ~ Introduction
- ~ The Application Flow
- ~ Code Intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Predictions
- ~ Deployment

=> ML Project 8 :- Phishing Classifier :

- ~ Introduction
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

=> ML Project 9 :- Thyroid Detection :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operation
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction



- ~ Deployment

## => ML Project 10 :- Visibility Climate :

- ~ Introduction
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validations and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

## => Time Series :

- ~ Arima, Sarima, Auto Arima
- ~ Time series using RNN LSTM, Prediction of NIFTY stock price
- ~ Time series using RNN LSTM, Prediction of NIFTY stock price

## => DL ANN - Introduction :

- ~ Introduction to Deep Learning
- ~ Importance of Deep learning
- ~ Why you should study Deep Learning? (Motivation)
- ~ ANN vs BNN
- ~ The first Artificial Neuron

## => DL ANN - Perceptron :

- ~ Overview of Perceptron
- ~ More about Perceptron
- ~ Perceptron implementation using python - 1
- ~ Perceptron implementation using python - 2
- ~ Perceptron implementation using python - 3
- ~ Perceptron implementation using python - 4
- ~ Perceptron implementation using python - 5
- ~ Perceptron implementation using python - 6
- ~ Perceptron implementation using python - 7
- ~ Python scripting & modular coding for Perceptron
- ~ Python logging basics and docstrings
- ~ Python packaging, Github actions, and PyPI

## => DL ANN - 1 :

- ~ Multilayer Perceptron
- ~ Forward propagation
- ~ Why we need an Activation function?
- ~ ANN implementation using tf.keras - 1
- ~ ANN implementation using tf.keras - 2
- ~ ANN implementation using tf.keras - 3
- ~ ANN implementation using tf.keras - 4
- ~ ANN with Callbacks | Tensorboard | Early Stopping | Model Checkpointing

## => DL ANN - 2 :

- ~ Vector
- ~ Differentiation
- ~ Partial differentiation
- ~ Maxima and minima concept
- ~ Gradient descent basics
- ~ In-depth understanding of Gradient descent with mathematical proof

## => DL ANN - 3 :

- ~ Chain rule
- ~ Backpropagation

## => DL ANN - 4 :

- ~ General problems in training Neural Networks
- ~ Vanishing and Exploding gradients
- ~ Activation Function Basics
- ~ Weight initialization
- ~ Activation Functions - 1
- ~ Activation functions - 2
- ~ Activation functions - 3
- ~ Transfer learning
- ~ Batch normalization -1
- ~ Batch normalization -2
- ~ Batch normalization -3

## => DL ANN - 5 :

- ~ Introduction to fast optimizers
- ~ Momentum optimization
- ~ NAG
- ~ Elongated bowl problem | AdaGrad
- ~ RMSProp
- ~ Adam
- ~ Loss functions
- ~ Regularization
- ~ Dropout

## => Computer Vision - Introduction :

- ~ Introduction to Course
- ~ Course Overview
- ~ Installing Anaconda, Pycharm & Postman
- ~ Working with Conda Envs

- ~ *Pycharm Introduction*
- ~ *Pycharm with Conda*
- ~ *Pycharm with venv*
- ~ *Pycharm with Pipenv*

#### => Computer Vision - CNN Foundations :

- ~ *Why CNN? Building an Intuition for CNN*
- ~ *CNN, Kernels, Channels, Feature Maps, Stride, Padding*
- ~ *Receptive Fields, Image Output Dimensionality Calculations, MNIST Dataset Explorations with CNN*
- ~ *MNIST CNN Intuition, Tensorspace.js, CNN Explained, CIFAR 10 Dataset Explorations with CNN*
- ~ *Dropout & Custom Image Classification Dog Cat Dataset*
- ~ *Deployment in Heroku, AWS, Azure*
- ~ *Deployment in GCP, AWS EBS*

#### => Computer Vision - CNN Architectures :

- ~ *LeNet-5*
- ~ *LeNet-5 Practical*
- ~ *AlexNet*
- ~ *AlexNet Practical*
- ~ *VGGNet*
- ~ *VGG16 Practical*
- ~ *Inception*
- ~ *Inception Practical*
- ~ *ResNet*
- ~ *Resnet Practical*

#### => Computer Vision - Image Classification Hyper Parameter Tuning :

- ~ *Keras Tuner*
- ~ *Building a simple model*
- ~ *Tuning with Keras Tuner*

#### => Computer Vision - Data Augmentation :

- ~ *What is Data Augmentation?*
- ~ *Benefits of Data Augmentation*
- ~ *Exploring Papers like RICAP, Random Erasing, Cutout*
- ~ *Exploring Augmentor*
- ~ *Exploring Roboflow*

#### => Computer Vision - Object Detection Basics :

- ~ *What is Object Detection?*
- ~ *Competitions for Object Detection*
- ~ *Bounding Boxes*
- ~ *Bounding Box Regression*
- ~ *Intersection over Union (IoU)*
- ~ *Precision & Recall*
- ~ *What is Average Precision?*

#### => Computer Vision - Object Detection Architectures :

- ~ *Object Detection Family*
- ~ *RCNN*
- ~ *RCNN Network Architecture*
- ~ *Cons of RCNN*
- ~ *FAST RCNN*
- ~ *FAST RCNN Network Architecture*
- ~ *Cons of FAST RCNN*
- ~ *FASTER RCNN*
- ~ *FASTER RCNN Network Architecture*
- ~ *YOLO*
- ~ *YOLO Architecture*
- ~ *YOLO Limitations*
- ~ *SSD*
- ~ *SSD Network*

#### => Computer Vision - Practicals Object Detection using Tensorflow 1.x :

- ~ *Introduction to TFOD1.x*
- ~ *Using Google Colab with Google Drive*
- ~ *Installation of Libraries in Colab*
- ~ *TFOD1.x Setup in Colab*
- ~ *Visiting the Model Zoo*
- ~ *Inferencing in Colab*
- ~ *Inferencing in Local*
- ~ *Important Configurations Files*
- ~ *Webcam Testing*

#### => Computer Vision - Practicals Training a Custom Cards Detector using Tensorflow1.x :

- ~ *Custom Model Training in TFOD1.x*
- ~ *Our Custom Dataset*
- ~ *Doing Annotations or labeling data*
- ~ *Selection of Pretrained Model from Model Zoo*
- ~ *Files Setup for Training*
- ~ *Let's start Training in Colab*
- ~ *Export Frozen Inference Graph*
- ~ *Inferencing with our trained model in Colab*
- ~ *Training in Local*
- ~ *Inferencing with our trained model in Local*

#### => Computer Vision - Practicals Creating an Cards Detector Web App with TFOD1 :

- ~ *Code Understanding*
- ~ *WebApp Workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*

~ *Debugging our Application*

=> Computer Vision - Practicals Object Detection using Tensorflow 2.x :

~ *Introduction to TFOD2.x*

~ *Using the Default Colab Notebook*

~ *Google Colab & Drive Setup*

~ *Visiting TFOD2.x Model Garden*

~ *Inference using Pretrained Model*

~ *Inferencing in Local with a pretrained model*

=> Computer Vision - Practicals Training a Custom Chess Piece Detector using Tensorflow2 :

~ *Custom Model training in TFOD2.x*

~ *Our Custom Dataset TF2*

~ *File Setup for Training*

~ *Let's start Training*

~ *Let's start Training*

~ *Stop Training or resume Training*

~ *Evaluating the trained model*

~ *Convert CKPT to Saved Model*

~ *Inferencing using the Custom Trained Model in Colab*

~ *Inferencing using the Custom Trained Model in Local PC*

=> Computer Vision - Practicals Creating an Chess Piece Detector Web App with TFOD2 :

~ *Creating a Pycharm project & Environment Setup TF2*

~ *Application Workflow*

~ *Code understanding*

~ *Testing our App with Postman*

~ *Debugging our Application*

=> Computer Vision - Practicals Object Detection using Detectron2 :

~ *Introduction to Detectron2*

~ *Detectron2 Colab Setup*

~ *Visiting Detectron2 Model Zoo*

~ *Detectron2 Pretrained Model Inferencing*

=> Computer Vision - Practicals Training a Custom Detector using Detectron2 :

~ *Detectron2 Custom Training*

~ *Exploring the Dataset*

~ *Registering Dataset for Training*

~ *Let's start Training*

~ *Inferencing using the Custom Trained Model in Colab*

~ *Evaluating the Model*

=> Computer Vision - Practicals Creating an Custom Detector Web App with Detectron2 :

~ *Creating a Pycharm project & Environment Setup Detectron2*

~ *Application Workflow*

~ *Code understanding*

~ *Testing our App with Postman*

~ *Debugging our Application*

=> Computer Vision - Practicals Object Detection using YoloV5 :

~ *Introduction to YoloV5*

~ *YoloV5 Colab Setup*

~ *Inferencing using Pre Trained Model*

=> Computer Vision - Practicals Training a Custom Warehouse Apparel Detector using YoloV5 :

~ *Custom Training with YoloV5*

~ *Exploring the Dataset*

~ *Doing Annotations or labeling data*

~ *Setting up Google Colab & Drive*

~ *Let's start Training*

~ *Inferencing using the Custom Trained Model in Colab*

=> Computer Vision - Practicals Creating an Warehouse Apparel Detector Web App with YOLOV5 :

~ *Creating a Pycharm project & Environment Setup Yolo*

~ *Application Workflow*

~ *Code understanding*

~ *Testing our App with Postman*

~ *Debugging our Application*

=> Computer Vision - Image Segmentation :

~ *Segmentation Introduction*

~ *From Bounding Box to Polygon Masks*

~ *What is Image Segmentation?*

~ *Types of Segmentation*

~ *MASKRCNN*

~ *MASK RCNN Architecture*

=> Computer Vision - MASK RCNN Practicals with TFOD :

~ *Segmentation with TFOD1.x*

~ *Local Setup MASKRCNN*

~ *Exploring the Dataset*

~ *Data Annotation*

~ *Model Selection*

~ *Files Setup for Training*

~ *Model Training*

~ *Export Frozen Inference Graph*

~ *Model Prediction*

=> Computer Vision - MASKRCNN practical with Detectron2 :

~ *Introduction to Detectron2*

~ *Detectron2 Colab Notebook*

- ~ *Exploring the Model Zoo*
- ~ *Detecron2 Colab Setup*
- ~ *Custom Training with Detectron2*
- ~ *Exploring our Dataset*
- ~ *Data Annotation*
- ~ *Data Preparation*
- ~ *Setup for Training*
- ~ *Let's start Training*
- ~ *Inferencing using the Custom Trained Model in Colab*
- ~ *Evaluating the Model*

#### => Computer Vision - Face Recognition Project :

- ~ *Introduction to Project*
- ~ *Requirement Gathering*
- ~ *Techstack Selection*
- ~ *Project Installation*
- ~ *Project Demo*
- ~ *Project Workflow*
- ~ *Core Components of the Application*
- ~ *Data Collection Module*
- ~ *Generate Face Embeddings*
- ~ *Training Face Recognition Module*
- ~ *Prediction Pipeline*
- ~ *Entry point of the Application*
- ~ *Application Workflow*
- ~ *Debugging our Application*

#### => Computer Vision - Object Tracking Project :

- ~ *Object Tracking project*
- ~ *Project Installation Tracking*
- ~ *Project Demo*
- ~ *Code Understanding*

#### => Computer Vision - GANS :

- ~ *Introduction to GANS*
- ~ *GAN Architecture*
- ~ *GAN PRACTICALS Implementation*

#### => Computer Vision Project - Traffic Vehicle Detection :

- ~ *Introduction to Vehicle Detection project*
- ~ *Requirement Gathering*
- ~ *Framework Selection*
- ~ *Detailed Project Workflow*
- ~ *Data Collection Scrap*
- ~ *Data Preparation*
- ~ *Data augmentation augementer*
- ~ *Data Annotations*
- ~ *Model Training*
- ~ *Creating a Pycharm project & Environment Setup TVD*
- ~ *WebApp Workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*
- ~ *Debugging our Application*

#### => Computer Vision Project - Helmet Detection :

- ~ *Introduction to Helmet Detection project*
- ~ *Requirement Gathering*
- ~ *Techstack Selection*
- ~ *Detailed Project Workflow*
- ~ *Data Collection*
- ~ *Data Preparation*
- ~ *Data Augmentation*
- ~ *Data Annotations*
- ~ *Model Training*
- ~ *Creating a Pycharm project & Environment Setup HD*
- ~ *WebApp Workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*
- ~ *Debugging our Application*

#### => Computer Vision Project - Fashion Apparel Detection :

- ~ *Introduction to Fashion Apparel Detection project*
- ~ *Requirement Gathering*
- ~ *Techstack Selection*
- ~ *Detailed Project Workflow*
- ~ *Data Collection*
- ~ *Data Preparation*
- ~ *Data Augmentation*
- ~ *Data Annotations*
- ~ *Model Training*
- ~ *Creating a Pycharm project & Environment Setup FAD*
- ~ *Project Demo*
- ~ *WebApp Workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*
- ~ *Debugging our Application*

#### => Computer Vision Project - Image TO Text OCR :

- ~ *Introduction to Project*
- ~ *Project Installation OCR*
- ~ *Project Demo*

- ~ Application Workflow
- ~ Code Understanding
- ~ Debugging our App
- ~ Different OCR's available

#### => Computer Vision Project - Shredder System :

- ~ Introduction to Shredder Systems
- ~ Requirement Gathering
- ~ Techstack Selection
- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation
- ~ Model Selection from Zoo
- ~ Model Training
- ~ Creating a Pycharm project & Environment Setup SS
- ~ Application Workflow
- ~ Project Demo
- ~ Code Understanding
- ~ Debugging our Application
- ~ Project Workflow
- ~ Project Workflow

#### => Computer Vision Project - Automatic Number plate Recognition with TFOD1.x :

- ~ Introduction to ANPR Project
- ~ Requirement Gathering
- ~ Tech Stack Selection
- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation
- ~ Model Selection From Zoo
- ~ Model Training
- ~ Creating a Pycharm project & Environment Setup ANPR
- ~ Application Workflow
- ~ Create Google OCR API Key
- ~ Project Demo
- ~ Code Understanding
- ~ Debugging our Application

#### => NLP Overview :

- ~ NLP Overview
- ~ NLP very basic

#### => NLP Word Embeddings :

- ~ TFIDF
- ~ Word Embeddings Part-1
- ~ Word Embeddings Part-2

#### => NLP RNN :

- ~ RNN basic
- ~ RNN Implementation

#### => NLP LSTM & GRU :

- ~ LSTM Introduction
- ~ GRU

#### => NLP Attention Based Model :

- ~ Encoder Decoder and Attention Mechanism
- ~ Attention All You Need Paper Understanding

#### => NLP Transfer Learning in NLP :

- ~ GPT and BERT Model
- ~ SOTA Model with Paper Discussions
- ~ Albert & DistillBert Project Discussion

#### => NLP Project :- Megatron :

- ~ Megatron Project

#### => NLP Project:- Brand Measures :

- ~ Brand Measures Project

#### => NLP Project:- Text to Speech :

- ~ Introduction
- ~ Project Setup Text to Speech
- ~ Project Demo
- ~ Code Explanation
- ~ Project Workflow
- ~ Prediction with Postman
- ~ Debugging Application

#### => NLP Project:- Speech To Text :

- ~ Introduction
- ~ Project Setup Speech To Text
- ~ Project Demo
- ~ Code Explanation
- ~ Project Workflow
- ~ Prediction with Postman
- ~ Debugging Application

#### => NLP Project:- Spell Corrector :

- ~ Introduction
- ~ Project Setup Spell Corrector

- ~ *Project Demo*
- ~ *Code Explanation*
- ~ *Project Workflow*
- ~ *Prediction with Postman*
- ~ *Debugging Application*

=> NLP Project:- Named Entity Recognition :

- ~ *NER using BERT*

=> NLP Project:- Machine Translation & Keyword Spotting :

- ~ *Machine Translation*
- ~ *Keyword Spotting*

=> NLP Project:- Keyword Extractor & Summarization :

- ~ *Keyword Extraction*
- ~ *Extractive Text Summarization*

=> NLP project:- Paraphrasing :

- ~ *Rephrase Project*

=> BigData - Introduction to Big Data and Data Engineering :

- ~ *Big Data Engineering*

=> BigData - Introduction to Distributed Systems - Hadoop and MapReduce :

- ~ *Big Data Engineering Introduction*

=> BigData - Map Reduce & YARN :

- ~ *Big Data Hadoop Map Reduce YARN*
- ~ *Hadoop Map Reduce Hands On*

=> BigData - Hive :

- ~ *Apache hive*

=> BigData - Hive Hands On :

- ~ *Apache hive Hands On*

=> BigData - NoSQL and Hbase :

- ~ *Big Data HBase*
- ~ *Hbase hands On*

=> BigData - Sqoop :

- ~ *Big Data Sqoop*
- ~ *Big Data Sqoop Hands On*

=> BigData - Spark :

- ~ *Spark - Introduction*
- ~ *Big Data Engineering using PySpark- RDDs*
- ~ *Spark hands on - RDD*
- ~ *Big Data Engineering using PySpark- Core, Internals, Architecture*
- ~ *Apache Spark Actions\_ Transformations*
- ~ *Apache Spark Caching*
- ~ *Big Data Engineering using PySpark- Shared Vars , Coalesce Repartition*
- ~ *Big Data Engineering using PySpark- Dataframes*
- ~ *Spark hands on - Dataframe*
- ~ *Spark hands on - Databricks*
- ~ *Big Data Engineering using PySpark- Catalyst& Tungsten*

=> BigData - Spark ML :

- ~ *Big Data Engineering using PySpark- MLLib*
- ~ *Spark hands On - Spark ML Lib*

=> BigData - Spark Streaming :

- ~ *Big Data Engineering using PySpark- Streaming Part 1*
- ~ *Big Data Engineering using PySpark- Streaming Part 2*
- ~ *Spark hands On - Spark Streaming*

=> BigData - Kafka :

- ~ *Big Data Kafka*
- ~ *Big Data Kafka Hands on*

=> BigData - Apache Airflow - Workflow Management Platform :

- ~ *Big Data - Airflow*
- ~ *Big Data Airflow Hands On*

=> Big Data Projects :

- ~ *IoT Sensor data pipeline using Kafka-Spark Streaming*
- ~ *Product Recommendation Engine using Kafka-Spark Streaming*
- ~ *Short Video App Analytics*

=> Basic Charts in Power BI :

- ~ *2.0 Basic Charts in Power BI Desktop*
- ~ *2.1 Column Chart in Power BI*
- ~ *2.2 Stacked Column Chart in Power BI*
- ~ *2.3 Pie Chart in Power BI*
- ~ *2.4 Donut Chart in Power BI*
- ~ *2.5 Funnel Chart in Power BI*
- ~ *2.6 Ribbon Chart*
- ~ *2.7 Include and Exclude*
- ~ *2.8 Export data from Visual*

=> Cards and Filters :

- ~ *6.0 Cards and Filters in Power BI*
- ~ *6.1 Number Card*
- ~ *6.2 Text Card*
- ~ *6.2.1 Formatting of Text Card*

- ~ 6.3 Date Card
- ~ 6.3.1 Date Card (Relative Filtering)
- ~ 6.4 Multi-Row Card
- ~ 6.5 Filter on Visual
- ~ 6.6 Filter on This Page
- ~ 6.7 Filter on All Pages
- ~ 6.8 Drillthrough in Power BI

#### => Objects in Power BI :

- ~ 9.1 Insert Image in Power BI
- ~ 9.2 Insert Text in Power BI
- ~ 9.3 Insert Shapes in Power BI
- ~ 9.4 Insert Buttons in Power BI
- ~ 9.5 Web URL Action in Power BI
- ~ 9.6 Page Navigation Action in Power BI
- ~ 9.7 Bookmark Action in Power BI
- ~ 9.8 Drillthrough Action in Power BI

#### => Power Query - Number Functions :

- ~ 13.0 Number Functions in Power Query (Power BI)
- ~ 13.1 Basic Number Functions in Power Query (Power BI)
- ~ 13.2 Percentage, Percent Of, Module in Power Query (Power BI)
- ~ 13.3 Round Functions in Power Query (Power BI)
- ~ 13.4 IsEven, IsODD, Sign in Power Query (Power BI)

#### => Power Query - Append Files :

- ~ 14.1 Append multiple CSV files in a folder in Power Query (Power BI)
- ~ 14.2 Append multiple excel sheets, Tables in Power Query (Power BI)
- ~ 14.3 Append Excel sheets or Tables with different columns in Power BI
- ~ 14.4 Append multiple Excel files from a folder in Power BI
- ~ 14.5 Append different data sources in Power BI

#### => Power Query - Merge Files :

- ~ 15.0 Merge Files and Tables in Power BI
- ~ 15.1 Merge Sheets or Tables in Power Query (Power BI)
- ~ 15.2 Merge Data from multiple Excel files or Workbooks in Power BI
- ~ 15.3 Merge data from different data sources in Power Query (Power BI)
- ~ 15.4 Merge data having multiple criteria in Power BI

#### => Introduction to tableau :

- ~ Tableau Introduction
- ~ Download and Install Tableau
- ~ Tableau Vs Excel

#### => Charts - 1 :

- ~ Column Chart
- ~ Horizontal Bar Chart
- ~ Stacked Column Chart
- ~ Stacked Bar Chart
- ~ Keep Only, Exclude
- ~ Keep Only, Exclude2\_Normal
- ~ Publish to Tableau Public

#### => Charts - 2 :

- ~ Pie Chart
- ~ Multiple Pie Chart
- ~ TreeMap\_Editing
- ~ Packed Bubble Chart
- ~ Word Cloud OR Word Map
- ~ Formatting payal

#### => SQL :

- ~ Database Architecture
- ~ Introduction to SQL
- ~ Constraints
- ~ Data Definition Language (DDL)
- ~ Data Query Language (DQL)
- ~ Data Manipulation Language (DML)
- ~ Joins
- ~ Import Export
- ~ Aggregate Functions
- ~ Order by, Having & Limit Clause
- ~ String Functions
- ~ Datetime functions
- ~ Understanding Regular Expressions
- ~ Nested Queries
- ~ Views
- ~ Stored Procedures
- ~ WindowsFn
- ~ Python-SQL Connectivity

# Career Counselling

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Career-Counselling>

## Course Description :-

Career counselling is a type of advice-giving and support provided by career counsellors to their clients, to help the clients manage their journey through life, learning and work changes (career). We at iNeuron will be providing career counselling to students who want to shape their future in a better way with correct guidance and achieve new heights.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Develop soft skills
- => Design Your Own portfolio
- => Career Counselling(One to One)
- => Time Management

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shivan Kumar :

~ Associate Data Scientist, Mentor, and Kaggle Master with 2 Years of Experience. Experience in building models that translate data points into business insights. Highly accurate at collecting, analyzing, and interpreting large datasets, developing machine learning, deep learning, Chatbots models, and deploying the solutions on the cloud. I also love to participate in Hackathons. In my free time, I like to mentor students to learn Data Science and achieve their goals.

## Curriculum details :-

- => How to develop soft skills? :
  - ~ Writing Skills
  - ~ Public speaking
  - ~ Work Ethics
  - ~ Critical Thinking
  - ~ Creativity and Adaptability
- => How to write a professional Email?
- => Time Management
- => How to become a professional? :
  - ~ LinkedIn
  - ~ Github
  - ~ Open Source Contributor
- => How to design a portfolio?
- => Career Counselling(One to One) :
  - ~ Student Screening
  - ~ Discussion on students interest
  - ~ Student interest roadmap
  - ~ Introducing various education domains
  - ~ Expert Advice
  - ~ Roadmap from a domain expert
  - ~ Portfolio Discussion



# AWS QuickSight

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/AWS-QuickSight>

## Course Description :-

AWS QuickSight is one of the most powerful Business Intelligence tools which allows you to create interactive dashboards within minutes to provide business insights into the organizations. There are number of visualizations or graphical formats available in which the dashboards can be created. The dashboards get automatically updated as the data is updated or scheduled. You can also embed the dashboard created in Quicksight to your web application.

## Course Features :-

- => Self-Paced Classes
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Working with filters
- => Building charts
- => Dashboards

## Requirements :-

- => An AWS account
- => Little bit of Data Analysis Knowledge
- => Dedication
- => Internet Connection

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Quick Start to Quicksight :
  - ~ The course overview Preview
  - ~ big data analytics and aws
  - ~ How Quicksight is different than other BI Tools Preview
  - ~ BI solution based on quicksight
  - ~ how to get started with quicksight
  - ~ Performance Your first analysis
- => Importing data to Quicksight :
  - ~ AWS Big data ecosystem
  - ~ importing files to quicksight
  - ~ importing databases to quicksight part 1 Preview
  - ~ importing databases to quicksight part 2
  - ~ importing data from saas services to quicksight
  - ~ edit existing data sources in quicksight
- => Enriching your data in Quicksight :
  - ~ Joining datasets
  - ~ using functions Part 1
  - ~ using functions Part 2
  - ~ applying filters
  - ~ understanding spice layer
- => Building Interactive Visuals Using QuickSight :
  - ~ Creating a Quicksight Analysis
  - ~ Explore various charting options
  - ~ Exploring various Map options
  - ~ Exploring various table and other visual options
- => Building a Mini Project in AWS and Quicksight :
  - ~ Mini project Overview
  - ~ Mini Project Architecture
  - ~ Data ingestion for mini project
  - ~ Reports and dashboards

# Class 6th Chemistry

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Class-6th-Chemistry>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. In this, the Chemistry section focuses on concepts like Fibre to Fabric, Sorting materials into groups, Separation of substances, etc.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Variety in fabrics
- => Fibre
- => Some plant fibres
- => Spinning Cotton Yarn
- => Yarn to fabric
- => History of clothing material
- => Objects around us
- => Properties of materials
- => Methods of Separation of substances
- => Ways to bring a change
- => Can all changes be reversed?

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => CHAPTER 3 - FIBRE TO FABRIC :
  - ~ LECTURE 1 Natural Fibre and Synthetic Fibre Preview
  - ~ LECTURE 2 Yarn to Fibre
  - ~ LECTURE 3 NCERT Question Discussion
- => CHAPTER 5 - Separation of Subs :
  - ~ LECTURE 1 : Introduction
  - ~ LECTURE 2 : Method of Separation
  - ~ LECTURE 3 : Filtration, Sedimentation, Centrifugation Preview
  - ~ LECTURE 4 : Evaporation, Condensation
- => CHAPTER 6 - CHANGES AROUND US :
  - ~ LECTURE 1 : Reversible and Irreversible Change
  - ~ LECTURE 2 : Physical and Chemical Change
  - ~ LECTURE 3 : NCERT Question Discussion
- => NaN :
  - ~ NaN

# Drone Live Class

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Drone-Live-Class>

## Course Description :-

iNeuron has created a course in collaboration with industry experts who are achieving incredible things with autonomous robotics. You can learn about the process of making a drone or how drones are used in various sectors. The majority of drones are ready to go to the skies. We provide Robotics concepts to teach you about dynamics and control so you can get a better understanding of how that works. We present an outline of Drones and Autonomous Systems for individuals who want to learn more about their mechanics. Each lecture is intended to lay the groundwork for how autonomous systems can alter our perceptions of robotics.

## Course Features :-

- => Online live classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Basic understanding of UAV and Maneuvering of Aircraft.
- => Construction(Multirotor drone basic)
- => Drone Body Assembly
- => Transmitter and Receiver
- => Flight controller
- => Calibrations

## Requirements :-

- => No prior experience of Drones

## Instructors :-

- => Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

- => Introduction :

- ~ What is this course all about?
- ~ What is in this course?
- ~ Pre-knowledge required.

- => Basic understanding of UAV and Maneuvering of Aircraft. :

- ~ Aerodynamics of UAV
- ~ Forces of Flight
- ~ Theory of flight
- ~ Centre of Gravity
- ~ Thrust to Weight Ratio
- ~ Mach Number.

- => Types of UAV

&lt;Parts of UAV and their diagrams > :

- ~ Multi-Rotor
- ~ Fixed wings
- ~ Single rotor
- ~ Hybrid VTOL
- ~ Their Pros, and Cons

- => Construction(Multirotor drone basic) :

- ~ Axis of rotation for Drone
- ~ Components needed to construct a drone.
- ~ Terminologies
- ~ Tools needed for drone construction
- ~ Conclusion.

=> Explaining each component and its functions(Motors, Frame, and ESC) :

- ~ Different types of drone frame
- ~ How to choose it
- ~ Motors: what are BLDC motors
- ~ Why we are using it.
- ~ Their Rating and Explaining how to correctly select the motors for the drone.

=> Explaining each component and its functions(Power Distribution Boards & Common Module wire and ESC) :

- ~ How to solder ESC power wires
- ~ How to solder the Battery power wires and connectors

=> Explaining each component and its functions(Drone Body Assembly) :

- ~ Attaching the arms to the frame of the drone.
- ~ Attaching the Motors to the Arms
- ~ Attaching the ESC to the arm and connecting them to the motors.
- ~ Propellers and how to select a propeller for your drone, and how to distinguish between clockwise and anti-clock propellers

=> Explaining each component and its functions(Transmitter and Receiver) :

- ~ Introduction about the Transmitter and Receiver
- ~ Binding the Receiver with the Transmitter
- ~ Explaining about Telemetry
- ~ Explaining the features Of the Transmitter with the help of servo and how we can utilize all of its features

=> Explaining each component and its functions(Flight controller) :

- ~ Explaining Different aspects of Pixhawk flight controller
- ~ Connecting the GPS module to the flight controller
- ~ Attaching all the necessary components to the flight controller (Switch Buzzer, Telemetry, Power module)
- ~ Connecting ESC to the Pixhawk with the correct order
- ~ Connecting RC receiver with the Pixhawk

=> Explaining each component and its functions(Battery and Charge) :

- ~ Explain the Battery and charger

=> Explaining each component and its functions(Software MISSION PLANNER) :

- ~ 1. Downloading and setting up the Mission Planner
- ~ 2. Connecting Pixhawk with the Mission Planner
- ~ 3. Explaining various features of Mission Planner

=> Explaining each component and its functions(Calibrations Part) :

- ~ Calibrating GPS and other onboard sensors with the Pixhawk using Mission Planner
- ~ Calibrating ESC and Motors

=> Explaining each component and its functions(FIRST FLIGHT) :

- ~ Taking First Flight
- ~ Demonstrating how to control UAV using the remote controller

# Class 8th Math

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Class-8th-Math>

## Course Description :-

This course is useful for Grade 8 students. In this course, entire NCERT will be covered, Various questions from NCERT, NCERT exemplar and previous year will also be discussed. Dedicated doubt clearing sessions will also be conducted for helping the students regularly throughout their learning journey.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Algebra
- => Statistics
- => Geometry
- => Numbers
- => Mensuration

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => Rational Numbers :
  - ~ Lecture 1 : Closure Property Preview
  - ~ Lecture 2 : Commutative Property Preview
  - ~ Lecture 3 : Associative Property
  - ~ Lecture 4 : Natural Number Properties
  - ~ Lecture 5
  - ~ Lecture 6 : NCERT Solution Ex1.1 Question 1
  - ~ Lecture 7 : Inserting Rational Numbers Between Two Rational Numbers
  - ~ Lecture 8 : NCERT Solution Ex1.2 Question 1-5
  - ~ Lecture 9 : NCERT Solution Ex1.2 Question 6,7
- => Linear Equations in One Variable :
  - ~ Lecture 1 : NCERT Solution Ex2.1
  - ~ Lecture 3 : NCERT Solution Ex2.2 Question 1-4
  - ~ Lecture 4 : NCERT Solution Ex2.2 Question 5-8
  - ~ Lecture 5 : Part 1 NCERT Solution Ex2.2 Question 9-16
  - ~ Lecture 7 : NCERT Solution Ex2.3
  - ~ Lecture 9 : NCERT Solution Ex2.4
  - ~ Lecture 10 : NCERT Solution Ex2.4 Question 6-9
  - ~ Lecture 12 : NCERT Solution Ex2.5
  - ~ Lecture 13 : NCERT Solution Ex 2.5 Question 7-10
  - ~ Lecture 15 : NCERT Solution Ex 2.6
- => Understanding Quadrilaterals :
  - ~ Lecture 1 : Introduction Preview
  - ~ Lecture 2 : Part 1 Preview
  - ~ Lecture 3 : NCERT Solutions Ex 3.1
  - ~ Lecture 4 : NCERT Solutions Ex3.3q6p3,4
  - ~ Lecture 6 : Theory
  - ~ Lecture 7 : NCERT Solutions Ex 3.2
  - ~ Lecture 8 : Types of Quadrilaterals
  - ~ Lecture 9 : NCERT Solutions Ex3.3 Question 1,2,3,4
  - ~ Lecture 10 : NCERT Solutions Ex3.3 Question 5,6,7,8
  - ~ Lecture 11 : NCERT Solutions Ex3.3 Question 9,10,11,12
  - ~ Lecture 12 : NCERT Solutions Ex 3.4
- => Practical Geometry :
  - ~ Lecture 1 : 4 sides & Dia known
  - ~ Lecture 2 : 2 Dia & 3 sides known quad const
  - ~ Lecture 3 : 2 adjacent sides & 3 angles known

- ~ Lecture 4 : 3 sides & 2 included angles known
- ~ Lecture 5 : Square & Rhombus Const

#### => Data Handling :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : NCERT Solutions Ex5.1 Question 1
- ~ Lecture 3 : NCERT Solutions Ex5.1
- ~ Lecture 4 : Ex5.1 Question 3,4
- ~ Lecture 5 : Circle Introduction
- ~ Lecture 6 : NCERT Solutions Ex5.2 Question 1,2,3
- ~ Lecture 7 : NCERT Solutions Ex5.2 Question 4,5
- ~ Lecture 8 : Probability Introduction
- ~ Lecture 9 NCERT Solutions Ex5.3

#### => Squares and Square Roots :

- ~ Lecture 1 : Perfect Square Numbers
- ~ Lecture 2 : Properties of Perfect Square
- ~ Lecture 3 : Property of Perfect Square Continued and Pythagoren Triplet
- ~ Lecture 4 : Numbers between Square Numbers , Some Patterns in Square Numbers
- ~ Lecture 5, Exericse 6.1
- ~ Lecture 6, Finding the Square of a Number, Ex 6.2
- ~ Lecture 7 : Square Root, Finding Square Root through Prime factorisation
- ~ Lecture 8 : Exercise 6.3 Q1 to 5
- ~ Lecture 9 : Exercise 6.3 Q6 to 10
- ~ Lecture 10 : Square of a Perfect Square Number by Long Division Method
- ~ Lecture 11 : Ex 6.4 Q 1 to 7
- ~ Lecture 12 : Ex 6.4 Q 8 - 9

#### => Cubes and Cube Roots :

- ~ Lecture 1 : Understanding how to find cube of a number, how to check weather a number is a perfect cube or not
- ~ Lecture 2 : Ex 7.1 Q 1 to 4
- ~ Lecture 3 : Cube root , Method of finding Cube root of a number
- ~ Lecture 4 : Estimating Cube root of a number without factorisation

#### => Comparing Quantities :

- ~ Lecture 1 : Concept of ratio and percentage
- ~ Lecture 2 : Ex 8.1 Q 1 to 6
- ~ Lecture 3 : Finding Increase or Decrease Percenet, Finding Discounts
- ~ Lecture 4 : Profit and Loss , Sales Tax/ Value Added Tax/ Goods and Services Tax
- ~ Lecture 5 : Exericse 8.2 Q 1 to 5
- ~ Lecture 6 : Exercise 8.2 Q 6 to 8
- ~ Lecture 7 : Compound Interest
- ~ Lecture 8 : Deducing a formula for Compound Interest
- ~ Lecture 9 : Ex 8.3, Q 1 - 3
- ~ Lecture 10 : Ex 8.3, Q 4 - 7
- ~ Lecture 11 : Ex 8.3, Q 9 - 12

#### => Algebraic Expressions and Identities :

- ~ Lecture1\_Syllabus\_Course\_Contents\_Introduction
- ~ Lecture2\_All\_About\_Algebraic\_Expressions
- ~ Lecture3\_Algebraic\_Terminologies
- ~ Lecture4\_Monomial\_Binomial\_Polynomial
- ~ Lecture5\_Like\_Vs\_Unlike\_Terms
- ~ Lecture6\_Addition\_&\_Subtraction\_Algebraic\_Expressions
- ~ Lecture7\_NCERT\_EX9.1\_Problems\_Discussions
- ~ Lecture8\_Monomial\_Multiplication
- ~ LECTURE9\_NCERT\_EX\_9.2\_PROBLEM\_DISCUSSIONS
- ~ LECTURE10\_MULTIPLICATION\_MONOMIAL\_BY\_POLYNOMIAL
- ~ LECTURE11\_NCERT\_EX9.3\_PROBLEM\_DISCUSSIONS
- ~ LECTURE12\_MULTIPLYING\_POLYNOMIAL\_BY\_POLYNOMIAL
- ~ LECTURE13\_NCERT\_EX9.4\_PROBLEM\_DISCUSSION
- ~ LECTURE14\_ALL\_ABOUT\_IDENTITIES
- ~ LECTURE15\_NCERT\_EX9.5\_PROBLEM\_DISCUSSIONS
- ~ LECTURE16\_HOTS\_ALGEBRAIC\_QUESTIONS

#### => Visualising Solid Shapes :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : NCERT Solutions Ex10.1
- ~ Lecture 3 : Maps
- ~ Lecture 4 : Polyhedron Introduction
- ~ Lecture 5 : NCERT Solutions Ex10.3 Question 1,2,3
- ~ Lecture 6 : NCERT Solutions Ex 10.3 Question 4,5,6,7,8

#### => Mensuration :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : NCERT Solution Ex 11.1 Question 1,2
- ~ Lecture 3 : NCERT Solutions Ex11.1 Question 3,4,5
- ~ Lecture 4 : Area of Quadrilateral Introduction
- ~ Lecture 5 : NCERT Solutions Ex11.2 Question 1,2,3,4
- ~ Lecture 6 : NCERT Solutions Ex11.2 Question 5,6,7,8
- ~ Lecture 7 : NCERT Solutions Ex11.2 Question 9,10,11
- ~ Lecture 8 : Surface Area Introduction
- ~ Lecture 9 : NCERT Solutions Ex11.3 Question 1,2
- ~ Lecture 10 : NCERT Solutions Ex11.3 Question 3,4
- ~ Lecture 11 : NCERT Solutions Ex11.3 Question 5,6,7
- ~ Lecture 12 : NCERT Solutions Ex11.3 Question 8,9,10
- ~ Lecture 13 : Volume Introduction
- ~ Lecture 14 : NCERT Solutions Ex11.4 Question 1,2,3,4
- ~ Lecture 15 : NCERT Solutions Ex11.4 Question 5,6,7,8

#### => Exponents and Powers :

- ~ Lecture 1 : Understanding Exponents, Multiplicative Inverse
- ~ Lecture 2 : Laws of Exponent
- ~ Lecture 3 : Ex 12.1 Q 1 to 3
- ~ Lecture 4 : Ex 12.1 Q 3 to 7
- ~ Lecture 5 : Numbers In Standard Form
- ~ Lecture 6 : Ex 12.2 Q 1 to 4

=> Direct and Inverse Proportions :

- ~ Lecture 1 : Understanding direct Proportion
- ~ Lecture 2 : Inverse Proportion
- ~ Lecture 3 : Ex 13.1 Q 1 to 10
- ~ Lecture 4 : Ex 13.2 Q 1 to 5
- ~ Lecture 5 : Ex 13.2 Q 6 to 11

=> Factorisation :

- ~ Lecture1\_Introduction\_&\_Content\_GoThrough
- ~ Lecture2\_Factors\_Natural\_Numbers\_&\_Algebraic\_Expressions
- ~ Lecture3\_What\_Is\_Factorisation
- ~ Lecture4\_Factorisation\_By\_ReGrouping
- ~ LECTURE5\_NCERT\_EX14.1\_PROBLEM\_DISCUSSIONS
- ~ Lecture6\_Factorisation\_Using\_Identities
- ~ Lecture7\_Factorisation\_For\_(x+a)(x+b)
- ~ LECTURE8\_NCERT\_EX\_14.2\_PROBLEMS\_DISCUSSION
- ~ Lecture9\_Monomial\_Division
- ~ Lecture10\_Polynomial\_Division
- ~ Lecture11\_NCERT\_EX14.3\_PROBLEM\_DISCUSSIONS
- ~ LECTURE12\_SPOTTING\_ERROR\_&\_NCERT\_EX14.4\_PROBLEM\_DISCUSSIONS
- ~ Lecture13\_NCERT\_EXEMPLAR\_MCQ\_QUESTIONS\_DISCUSSIONS
- ~ LECTURE14\_HOTS\_ALGEBRAIC\_FACTORISATION\_QUESTIONS

=> Introduction to Graphs :

- ~ Lecture 1: Coordinates Introduction
- ~ Lecture 2 : Graphs Introduction
- ~ Lecture 3 : Ex 15.1 Que 1-3
- ~ Lecture 4 : Ex 15.1 Question 4
- ~ Lecture 5 Ex 15.1 Question 6&7
- ~ Lecture 6 : Ex 15.2 Question 1&2
- ~ Lecture 7 : Ex 15.2 Question 3&4
- ~ Lecture 8 : Ex15.1 Que 5 part(b)
- ~ Lecture 9 : Ex151 QUE 5part(a)
- ~ Lecture 10 : Ex15.3 QUE 1
- ~ Lecture 11 : Ex 15.3 QUE 2

=> Playing with Numbers :

- ~ LECTURE1\_COURSE\_CONTENT\_&\_INTRODUCTION
- ~ LECTURE2\_NUMBERS\_IN\_GENERAL\_FORM
- ~ LECTURE3\_REVERSING\_TWO\_DIGIT\_NUMBERS
- ~ LECTURE4\_REVERSING\_THREE\_DIGIT\_NUMBERS
- ~ LECTURE5\_PLAYING\_WITH\_LETTERS\_&\_DIGITS
- ~ LECTURE6\_DIVISIBILITY\_TEST
- ~ LECTURE7\_EXEMPLAR\_PROBLEM\_MCQ\_PROBLEM\_DISCUSSIONS
- ~ LECTURE8\_HOTS\_PROBLEM\_DISCUSSION

# Socket io Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Socket-io-Crash-Course>

## Course Description :-

This course will help you to grab the fundamentals of socket.io .

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Socket IO

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Socket IO Crash Course :

~ Socket IO

=> NaN :

~ NaN

~ NaN

~ NaN

~ NaN

~ NaN

~ NaN



# Pro Max Interview Preparation Edition 2

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Max-Interview-Preparation-Edition-2>

## Course Description :-

Pro Max Edition 2. These are interview preparation tests with a singular goal, to make sure you get a little better in real-world interviews. Leaderboards are ranked based on 1st attempt.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => Interview Preparation Theoretical Test
- => Interview Preparation Practical Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

- => Interview Preparation Test :
  - ~ Interview Preparation Test 1
  - ~ Interview Preparation Test 2
  - ~ Interview Preparation Test 3
  - ~ Interview Preparation Test 4
  - ~ Interview Preparation Test 5
  - ~ Interview Preparation Test 6

# Git and GitHub

---

Topic Name : DEVOPS

Sub-topic Name : GIT

Course link : <https://ineuron.ai/course/Git-and-GitHub>

## Course Description :-

Version control is a basic skill you need to thrive as a developer, whether you're just starting started with software development, looking for a developer job, or simply brushing up on your skills. Git (a version control system) and Github (a cloud service for Git controlled projects) make an excellent pair for creating and maintaining a well-structured project history!

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Git foundation
- => Setting, maintaining and tracking git repos
- => Git snapshots
- => Git for team management
- => Git branches
- => Git merging
- => Git and Github ecosystem
- => Tags and tickets

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Why we need GIT :

- ~ Why GIT is important
- ~ Collaboration and Forking
- ~ Installation of GIT
- ~ Autocompletion of GIT

=> Git foundation :

- ~ GIT Architecture and Github Gitlab and bitbucket
- ~ Initializing and exploring GIT
- ~ First commit and log messages
- ~ Git checksum and SHA-1
- ~ Understanding HEAD and Checksum

=> Getting into files :

- ~ Lets do it again
- ~ Track difference between files
- ~ Delete from repos
- ~ repo reset and unstaging
- ~ Can we reset commits?

=> Git Snapshots :

- ~ checkout with previous versions
- ~ Soft, Mixed and Hard reset
- ~ Ignoring the files
- ~ What to ignore

- ~ *Gitignore will not listen*
- ~ *This is not even in git docs*

=> GIT for team managements :

- ~ *Git tree listing*
- ~ *Git log in detail*

=> GIT branches :

- ~ *Git Branching basics*
- ~ *Creating a new branch in GIT*
- ~ *Checkout branches*
- ~ *RD of branches in GIT*

=> GIT Merging :

- ~ *Basics of Merging in GIT*
- ~ *Fast forward*
- ~ *Conflicts and merging in GIT*
- ~ *Stashing a branch*
- ~ *Stashing in multiple branches*
- ~ *Clean your stash*

=> GIT and GitHub :

- ~ *GIT online hosting*
- ~ *Creating a repo at Github*
- ~ *Uploading local repo to remote repo*
- ~ *Push for a remote collab*
- ~ *Merging from origin master*
- ~ *Assisting on open source projects*

=> Tags and Tickets :

- ~ *Concepts of tickets and tags*
- ~ *Pushing tags to Github*

# Deep Authenticator

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION PROJECT

Course link : <https://ineuron.ai/course/Deep-Authenticator>

## Course Description :-

In this project, you will build a two-stage authentication system. We would be using state-of-art algorithms like FaceNet and MTCNN to build this project.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => Deep Authenticator
- => Authentication and Authorization
- => MTCNN
- => FaceNet
- => FastAPI
- => Azure Container Registry and Azure App Service
- => GitHub Actions CI/CD

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Rishav Dash :

~ This is Rishav Dash. I am a Jr. Data Scientist and mentor at INeuron.ai with working experience in computer vision, natural language processing, Machine Learning, and Alops. Hands-on experience leveraging machine learning, deep learning, transfer learning models to challenging real-world problems, and building products to solve peoples problems.

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Project :- Deep Authenticator :

- ~ Introduction of Instructor
- ~ Project Overview
- ~ End Notes
- ~ Problem Statement
- ~ Understand the application scope
- ~ Tour to existing solution
- ~ End Notes
- ~ Solution Description
- ~ Tech Stack used
- ~ Tour to Architecture diagram
- ~ cost involved
- ~ End Notes
- ~ Folder Structure overview
- ~ Environment and Project Setup
- ~ User Login Authentication
- ~ Login Embedding Generation
- ~ User Registration Authentication
- ~ Registration Embedding Generation
- ~ Running project locally
- ~ Running project using Docker
- ~ End Notes
- ~ Azure cloud overview and Services overview
- ~ Provisioning Resources in cloud

- ~ *Pushing Docker Image to Azure Container Registry*
- ~ *Deploying to Azure App Services*
- ~ *CI/CD using GitHub Actions*
- ~ *Conclude the project*
- ~ *Assignments & External Resources*

# Advance Computer Vision

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/Advance-Computer-Vision>

## Course Description :-

Early and mid-career software and machine learning engineers with a fundamental understanding of computer vision who want to enhance their knowledge and skillset by studying advanced features to develop strong models can pursue this specialisation.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => N
- => u
- => l
- => l

## Requirements :-

- => Basic knowledge of Python programming
- => A system with stable internet connection
- => Your dedication

## Instructors :-

- => Sudhanshu Kumar :
  - ~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => CNN :
  - ~ CNN? Building An Intuition For CNN Preview
  - ~ CNN, Kernels, Channels, Feature Maps, Stride, Padding Preview
  - ~ Receptive Fields, Image Output Dimensionality Calculations, MNIST Dataset Explorations With CNN
  - ~ MNIST CNN Intuition, Tensorspace.js, CNN Explained, CIFAR 10 Dataset Explorations With CNN
  - ~ Custom Image Classification
  - ~ Forward And Back Propagation, LeNet
  - ~ AlexNet And VGGNet
  - ~ Unconvention & Pure CNNs, Inception
  - ~ InceptionV1, Inception V2 Continued, Batch Norm
  - ~ Inception
  - ~ Resnet Architecture
  - ~ Resnet Architecture
  - ~ Opencv
  - ~ Plant Disease Classification and Object Detection Intro
  - ~ Tensorflow Object Detection 1
  - ~ Tensorflow Object Detection 2
  - ~ Detectron 2 and Custom Training in Detectron 2
  - ~ Pytorch Basic and FashionMNIST
  - ~ Autograd | FashionMNIST | Transfer Learning
  - ~ Object Classification and Deployment on Heroku, AWS
  - ~ GCP, AWS, Packaging
  - ~ GPU Providers | AWS | GCP | Azure | Paperspace | DataCrunch | Floydhub
  - ~ Yolo
  - ~ Detectron 2 Segmentation | TF Segmentation
  - ~ Mask RCNN Using TF | Annotation of labelme | JSON to TF Records
  - ~ Mask RCNN
  - ~ Shredder Machine Project
  - ~ RCNN
  - ~ Face Recognition
  - ~ Face Recognition Code Discussion
  - ~ Fast RCNN
  - ~ Fast RCNN
  - ~ RPNN
  - ~ Project Discussion
  - ~ Detectron webapp
  - ~ Detectron2 web app

- ~ *Number plate detection project*
- ~ *SSD*
- ~ *Mask R-CNN*

# Class 8th Biology

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Class-8th-Biology>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. The Biology section focuses on concepts like food, bodily movements, surroundings of living organisms, garbage, etc.

## Course Features :-

=> Self Paced Videos

=> Completion Certificate

## What you will learn :-

=> Crop Production and management

=> Microorganisms Friend and Foe

=> Conservation of Plants and Animals

=> Cell Structure and Function

=> Reproduction in Animals

=> Reaching the age of adolescence

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Dr Nishtha Jain :

*~ I am a doctor by profession but a teacher by passion. I have been into the teaching profession for the last 3 years. I have been and am still a mentor for various courses which include technical as well as non-technical ones. These include MS-Excel, Tableau, Computer basics, Biology, English, etc. I love to learn, explore and share my knowledge to whatever extent possible. Being an ardent educator, I have always helped all my students and will continue to do the same.*

## Curriculum details :-

=> Crop Production and management :

- ~ Lecture 1 : Agriculture, Crops and their seasons Preview
- ~ Lecture 2 : Preparation of Soil Preview
- ~ Lecture 3 : Sowing, Adding Manure and Fertilizers Preview
- ~ Lecture 4 : Irrigation
- ~ Lecture 5 : Protection from weeds, Harvesting, Storage
- ~ Lecture 6 : Food from Animals
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions

=> Microorganisms Friend and Foe :

- ~ Lecture 1 : Introduction, Good and bad microorganisms
- ~ Lecture 2 : Uses of microbes and Diseases caused by them
- ~ Lecture 3 : Food spoilage and Food preservation
- ~ Lecture 4 : Different microbes
- ~ Lecture 5 : Nitrogen Cycle and Fixation
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions

=> Conservation of Plants and Animals :

- ~ Lecture 1 : Deforestation - Causes and Consequences
- ~ Lecture 2 : Conservation of Forests & Wildlife, Biosphere Reserves
- ~ Lecture 3 : Wildlife and Sanctuaries
- ~ Lecture 4 : Flora and Fauna, Species
- ~ Lecture 5 : National Parks
- ~ Lecture 6 : Red Data Book, Migration
- ~ Lecture 7 : Paper recycling and Reforestation
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions
- ~ Lecture 5 - NCERT Solutions



=> Cell Structure and Function :

- ~ Lecture 1 : Cell Discovery
- ~ Lecture 2 : Number, Shape and Size of cells
- ~ Lecture 3 : Cell Membrane, Cell wall, Cytoplasm, Protoplasm, Plastids & Vacuoles
- ~ Lecture 4 : Nucleus, Prokaryotes and Eukaryotes
- ~ Lecture 5 : Plant cell Animal Cell
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions

=> Reproduction in Animals :

- ~ Lecture 1 : Reproduction and Types' introduction
- ~ Lecture 2 : Male reproductive organs
- ~ Lecture 3 : Female reproductive organs
- ~ Lecture 4 : Fertilisation and Types
- ~ Lecture 5 : Development of embryo
- ~ Lecture 6 : Internal and external fertilisation, Life cycles
- ~ Lecture 7 : Asexual Reproduction and Types
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions

=> Reaching the age of adolescence :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Noticeable changes - 1
- ~ Lecture 3 : Noticeable changes - 2
- ~ Lecture 4 : Reproductive Phase
- ~ Lecture 5 : Hormones other than Sex Hormones
- ~ Lecture 6 : Nutritional needs, Exercise, Balanced diet, NO to Drugs
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions

# Docker Course

---

Topic Name : DEVOPS

Sub-topic Name : DOCKER

Course link : <https://ineuron.ai/course/Docker-Course>

## Course Description :-

Docker is an open platform that allows developers and system administrators to create, ship, and execute distributed applications on laptops, datacentre virtual machines, and the various cloud services. This course uses straightforward and easy-to-understand lectures to explain Docker to an absolute Beginner. This course will help you practise Docker commands and construct your own images using Dockerfiles, as well as Docker Compose. You will be creating Docker files for various use cases. You won't need to set up your own environment to gain some hands-on experience this way.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Docker containers
- => Containers vs virtual machines
- => Docker fundamentals
- => Custom Docker images
- => Multi container setup
- => Nginx
- => Docker AWS
- => TravisCI

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Docker Installation Basics :

- ~ What is Docker?
- ~ How to install Docker and Hello World
- ~ What is container in Docker
- ~ Docker vs Virtual Machine
- ~ First interaction with busy box image

=> Fundamentals of docker :

- ~ Docker lifecycle and PS
- ~ Start and delete a container
- ~ Getting a mongodb container for fun
- ~ Exploring exec command
- ~ Multiple ways to get inside a container

=> Custom Docker images :

- ~ Analogy for custom docker image
- ~ Our first base image and custom image
- ~ Behind the scene for custom image
- ~ Creating a custom mongodb image
- ~ Concept of caching in docker
- ~ Provide a custom name for your image

=> Project and Docker :

- ~ *Introduction to node project for docker*
- ~ *Introduction to node project for docker part 2*
- ~ *Containerize a node application*
- ~ *Performance upgrade in node project container*

=> Multi container setup :

- ~ *Introduction to multi docker container*
- ~ *A mini mongo connector project*
- ~ *Put your node code in a container*
- ~ *Introduction to docker compose*
- ~ *Connect 2 compose images in docker*
- ~ *Access the compose container app with browser*

=> Ngnix - production grade deployment :

- ~ *Ngnix A production grade docker*
- ~ *Attaching volumes in Docker*
- ~ *Types of docker files*
- ~ *Dev test and production stages*
- ~ *Understand react project for docker deployment*
- ~ *Docker for development*
- ~ *Docker for testing*
- ~ *Docker for production*

=> Docker AWS and Travis CI :

- ~ *Docker CI and AWS*
- ~ *What is CI CD Jenkins vs Travis CI*
- ~ *Moving to AWS Elastic Beanstalk*
- ~ *Moving project to GitHub repo*
- ~ *Reading Travis CI documentation*
- ~ *Writing our 1st Travis CI config file*
- ~ *AWS IAM user generation*
- ~ *Elastic Beanstalk and S3 bucket*
- ~ *Finally hosting app on AWS with CI integrated with docker*
- ~ *TURN OFF those AWS apps*

# Data Visualization using Matplotlib

---

Topic Name : K12

Sub-topic Name : CLASS9

Course link : <https://ineuron.ai/course/Data-Visualization-using-Matplotlib>

## Course Description :-

In this course, you will learn Data visualization using the matplotlib library. Hands-on practical oriented course in which you will learn various plots such as Bar graphs, Pie charts, Line charts, Scatter plots, histograms. After completion of this course, you will be able to visualize the data and find the underlying pattern.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to matplotlib
- => Data visualization
- => Visual Analysis
- => Different types of Chart using Matplotlib

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ Course Introduction
  - ~ Who is this course for?
  - ~ Course overview and course outcome
  - ~ Course Pre-requisites
  - ~ Overview of Python
  - ~ What are Graphs?
  - ~ What can we depict from graphs?
  - ~ What are the different types of graphs?
  - ~ What is Matplotlib?
  - ~ Why is Matplotlib used?
- => Installation of Matplotlib :
  - ~ Installing in Google Colab
- => Different Types of Plot :
  - ~ What is a Bar graph?
  - ~ What can we understand from the Bar graph?
  - ~ Apply Bar graph on chess dataset
- => Assignment 1 :
  - ~ Apply Bar graph on your own datasets and write down your observations from it.
- => Different Types of Plot :
  - ~ What is a Pie chart?
  - ~ What can we understand from the Pie chart?
  - ~ Applying Pie chart on chess dataset
  - ~ What is a Box plot?
  - ~ What can we understand from the Box plot?
  - ~ Applying Box plot on chess dataset
  - ~ What is a line chart?
  - ~ What can we understand from the line chart?
  - ~ Applying line charts on Chess dataset
  - ~ What is a Scatter plot?

- ~ What can we understand from the Scatter plot?
- ~ Applying Scatter plot on chess dataset
- ~ What is Histogram?
- ~ What can we understand from the Histogram?
- ~ Applying Histogram on chess dataset

=> Assignment 2 :

- ~ Apply Pie chart on your own datasets and write down your observations from it.

=> Assignment 3 :

- ~ Apply Box plot on your own datasets and write down your observations from it.

=> Assignment 4 :

- ~ Apply line chart on your own datasets and write down your observations from it.

=> Assignment 5 :

- ~ Apply Scatter plot on your own datasets and write down your observations from it.

=> Assignment 6 :

- ~ Apply Histogram on your own datasets and write down your observations from it.

=> Project :

- ~ Using matplotlib analyze geographical datasets and write down your observations from them.

=> Assignment 7 :

- ~ Using matplotlib analyze historical datasets and write down your observations

=> Course Summary :

- ~ Course outro
- ~ Future learning Path

# Job Ready Automation Tester with JavaScript Tech Neuron

---

Topic Name : TESTING

Sub-topic Name : AUTOMATION TESTING

Course link : <https://ineuron.ai/course/Job-Ready-Automation-Tester-with-JavaScript-Tech-Neuron>

## Course Description :-

In this live training program, you will be learning everything about Automation using different tools and libraries. We will be starting with JavaScript from the scratch (from zero) and will be automating Web Applications and API's as well. This course includes most of leading tools and framework like Cypress, Playwright, WebDriverIO with different libraries and integration with Chai, Mocha, JEST, Git, Github, Github Actions, Docker and Jenkins (CI-CD). After this course you will be ready to attend interviews and will be able to automate different Web Applications.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What are JavaScript engines
- => What is Automation
- => What is Automation Testing
- => Cypress Architecture
- => Selenium vs Cypress - Comparison
- => What is API and why API testing is important
- => REST vs SOAP
- => Playwright Explorer
- => Assertion in details with page and element
- => Async vs Sync
- => Locators In WDIO
- => Xpath in detail

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Mukesh Otmani :

~ Myself Mukesh Otmani having 10 years of experience in Automation Testing and worked with Dell International and SAP Labs India. I am also passionate teacher, mentor, have been into teaching for 8 years now and running a YouTube channel with 137000 subscribers and 480+ videos on different tools and libraries.

## Curriculum details :-

=> JavaScript :

- ~ What are JavaScript engines
- ~ What ES version of JavaScript is good for us
- ~ Variable and datatypes in JavaScript
- ~ Operators in JavaScript
- ~ Type and Operator precedence in JavaScript
- ~ What are conditionals in JavaScript
- ~ Ternary operator in JavaScript
- ~ Switch in JavaScript
- ~ Coercion and falsy values in JavaScript
- ~ Basics of functions in JavaScript
- ~ Functions in variable User Role in JavaScript
- ~ Understand the context in JavaScript
- ~ Hoisting in JavaScript
- ~ THIS in JavaScript

- ~ Introduction to Array in JavaScript
- ~ Callback and arrow function introduction in array
- ~ Fill and Filter in Array in JavaScript
- ~ Slice in JavaScript
- ~ Objects in JavaScript
- ~ Methods and objects in JavaScript
- ~ For loop basics in JavaScript
- ~ While and do while loops in JavaScript
- ~ For Each loop in JavaScript
- ~ For in and for of loop in JavaScript
- ~ Template literals in JavaScript
- ~ Maps in JavaScript
- ~ Classes and module exports in JavaScript
- ~ Private props getters and setters in JavaScript
- ~ Inheritance in JavaScript
- ~ Promise async and await in JavaScript

=> Automation Foundation :

- ~ What is Automation
- ~ What is Automation Testing
- ~ Advantages of Automation Testing
- ~ When to start and when to stop Automation
- ~ What not to automate and why
- ~ Type Of Automation
- ~ Tools for Automation Testing
- ~ Which tool to select for Automation
- ~ POC before selecting any Automation Tools
- ~ Automation Mindset
- ~ How to switch from Manual to Automation

=> Cypress :

- ~ What is Cypress - Advantage and Limitations
- ~ Cypress Architecture
- ~ Selenium vs Cypress - Comparison
- ~ Downloading and Installing Cypress
- ~ Quick tour of Cypress Test Runner and configuration files
- ~ What is Testing Framework In JavaScript and why do we need Testing Framework
- ~ Testing Frameworks - Mocha, Jasmine, Jest, Karma
- ~ What is Mocha and features of Mocha Framework
- ~ Mocha Structure
- ~ Writing test suites using mocha framework using Describe and Context
- ~ Exclusive test in mocha using .only .skip
- ~ Hooks In Mocha before, beforeEach, after, afterEach
- ~ What is Chai (Assertion Library)
- ~ Mocha with Chai
- ~ Chai assertions using Should
- ~ Chai assertions using Expect
- ~ Chai assertions using Assert
- ~ Write your first Cypress Test - Electron
- ~ Difference between Cypress and cy
- ~ Executing Cypress Test On Chrome
- ~ Executing Cypress Test On Firefox
- ~ Executing Cypress Test On Edge
- ~ Logs In Cypress
- ~ ViewPort In Cypress
- ~ Browser Navigation Commands In Cypress - Back, Forward, Reload
- ~ Cypress Default Assertions for each command
- ~ How to verify urls, title In Cypress
- ~ Cypress Inspector to locate element
- ~ How to use SelectorsHub with Cypress
- ~ Locators In Cypress
- ~ How to write Xpath - Static Xpath, Dynamic Xpath, Xpath Axes, Xpath tricks
- ~ How to Write CSS Selectors in details
- ~ Get method vs Find methods vs Contains methods
- ~ Cypress defaults check before interacting with WebElements
- ~ Type method in detail and assertions
- ~ Type In Cypress with text content
- ~ Type In Cypress with Keyboard Events
- ~ Type In Cypress with force
- ~ Type In Cypress with timeout
- ~ Click in Cypress using locators
- ~ Click in Cypress using coordinates
- ~ Click in Cypress using positions
- ~ Click multiple elements
- ~ How to click when element is hidden or disabled
- ~ Click In Cypress with timeouts
- ~ Click with Key combination - Example - Control + Click, Shift + Click
- ~ How to Interact with Textbox and assertion
- ~ How to Interact with Buttons and assertions
- ~ How to Interact with Checkbox, Radio Buttons and assertions
- ~ How to work with disabled elements
- ~ How to work with hidden elements
- ~ Handling Single Select Dropdown and assertion
- ~ Select values from dropdown using Index
- ~ Select values from dropdown using value
- ~ Select values from disabled dropdown
- ~ Select values from hidden dropdown
- ~ Deselection of dropdown
- ~ How to Select values from Non Select Dropdowns- Bootstrap, AngularJS dropdown etc

- ~ Handle AutoSuggestions In Cypress
- ~ Handle Calendar Control In Cypress
- ~ How to capture and verify error messages in Cypress
- ~ Wait in Cypress
- ~ Pause in Cypress
- ~ Debug in Cypress
- ~ Difference between wait() vs pause() vs debug()
- ~ How to handle multiple webelements - Arrays of WebElements
- ~ Use of next(),last(),first(),eq()
- ~ Keyboard Events In Cypress Example- enter, ESC, alt , delete etc
- ~ Mouse Hover In Cypress using invoke
- ~ Mouse Hover In Cypress using trigger
- ~ Verifying values after mouse hover
- ~ DoubleClick in Cypress using locators
- ~ DoubleClick in Cypress using cordinates
- ~ DoubleClick in Cypress using positions
- ~ How to DoubleClick when element is hidden or disabled
- ~ DoubleClick In Cypress with timeouts
- ~ DoubleClick with Key combination - Example - Control + Click, Shift + Click
- ~ RightClick in Cypress using locators
- ~ RightClick in Cypress using cordinates
- ~ RightClick in Cypress using positions
- ~ How to RightClick when element is hidden or disabled
- ~ RightClick In Cypress with timeouts
- ~ RightClick with Key combination - Example - Control + Click, Shift + Click
- ~ Drag and Drop In Cypress
- ~ Handle Alert Box In Cypress
- ~ Handle Prompt Box In Cypress
- ~ Handle Confirmation Box In Cypress
- ~ Handle Frames in Cypress
- ~ Handle Child tabs in Cypress
- ~ Handle Shadow DOM in Cypress
- ~ How to handle file upload in Cypress
- ~ How to handle file downloads in Cypress
- ~ What is plugins in Cypress and List of plugin in Cypress
- ~ How to use plugin in Cypress
- ~ How to create custom commands in Cypress
- ~ What is Cypress CLI
- ~ How to execute single test from CLI
- ~ How to execute multiple test from CLI
- ~ How to change browser from CLI
- ~ Generating HTML Reports Using Cypress
- ~ Capture videos and screenshots In Cypress
- ~ Running Test in headless mode
- ~ Fixtures in Cypress
- ~ Page Object Model In Cypress
- ~ Framework Implementation
- ~ Cypress Dashboard Service
- ~ Cypress Parallel Test
- ~ How to run Cypress Test on Cloud using BrowserStack
- ~ Push your code from local to github
- ~ Github pull request process
- ~ Creating Branches and merge branches
- ~ What is Jenkins
- ~ Setting Up Jenkins with Email Configurations
- ~ Running cypress test from Jenkins CI
- ~ Creating Jenkins Pipelin for Cypress Test Execution
- ~ What is Github Actions
- ~ Executing Cypress Test using Github Actions

#### => API Automation Using Cypress :

- ~ What is API and why API testing is important
- ~ REST vs SOAP
- ~ Status Code
- ~ API term and keyword before starting API Testing
- ~ Write First API Test Using Cypress
- ~ How to make Post API request
- ~ How to make Put API request
- ~ How to make Patch API request
- ~ How to make Delete API request
- ~ Difference between put and patch
- ~ What is Swagger and how to use Swagger
- ~ JSON Object
- ~ JSON Array
- ~ Nested JSON Object Nested JSON Array
- ~ How to verify response
- ~ Handle Authentication and Authorization In Cypress
- ~ Cypress Interview Questions
- ~ Cypress Cheatsheet
- ~ Soure Code
- ~ Automating multiple application

#### => Playwright :

- ~ What is Playwright and Features of playwright
- ~ Playwright Architecture
- ~ Selenium Vs Playwright
- ~ Cypress vs Playwright
- ~ Download Node.J and Configure on windows
- ~ Download and Install Visual Code



- ~ Installation of Playwright
- ~ Understanding Playwright folder structure
- ~ Execute sample test from Playwright
- ~ Execute sample test from Playwright in headed mode
- ~ Understanding Configuration file
- ~ Execute test in specific browser
- ~ Locators In Playwright
- ~ Text based search
- ~ Based on CSS
- ~ Find by test-id
- ~ Find multiple web elements
- ~ Filter locators
- ~ Type into elements with text
- ~ Type into elements with keyboard events
- ~ Type into elements with delay
- ~ Type into elements with force
- ~ Click using locator
- ~ Click using coordinates
- ~ Click using positions
- ~ Click with force
- ~ Click with keyboard events
- ~ Capture error message and assert in different ways
- ~ Difference between `textContent()` and `allTextContent()`
- ~ Why `await` does not apply to `allTextContent()`
- ~ Check and Uncheck
- ~ Disable WebElements
- ~ Hidden WebElements
- ~ Selecting values from dropdown using index
- ~ Selecting values from dropdown using value
- ~ Selecting values from dropdown using text
- ~ Deselecting values from dropdown
- ~ Debug from CLI
- ~ Debug from code
- ~ Playwright Inspector
- ~ Step By Step Execute / Resume Execution
- ~ Check logs for each activity
- ~ Playwright Explorer
- ~ Assertion in details with page and element
- ~ Assertion for elements to be checked /unchecked
- ~ Assertion for elements to be disabled / enabled
- ~ Assertion for elements to be Editable
- ~ Assertion for elements to be visible / invisible
- ~ Assertion for text contains
- ~ Assertion for class contains
- ~ Assertion for attribute contains
- ~ URL Assertions
- ~ Title Assertions
- ~ Negating Assertions
- ~ Soft Assertions
- ~ How to deal with multiple web elements
- ~ How to handle calendar controls
- ~ How to handle autosuggestions
- ~ How to handle flaky test
- ~ How to handle waits in Playwright
- ~ What is `AutoWaiting` In Playwright
- ~ Condition checked in `AutoWait`
- ~ Why Playwright fails even after `AutoWait`
- ~ Add `waitFor` condition
- ~ Different `waitFor` conditions for pages and locators
- ~ Modify existing wait timeout for expect and locators
- ~ What is Promise in Playwright
- ~ How to handle `Promise.all`
- ~ Handle multiple tabs in playwright
- ~ How to handle frames in playwright
- ~ Handle alert window in playwright
- ~ `pause` method in playwright
- ~ How to generate report in Playwright
- ~ Attach screenshot in report for each step, on failure
- ~ How to generate pdf in Playwright
- ~ What is codegen
- ~ How to record and play your test in codegen
- ~ Analyse test recorded by codegen
- ~ Execute auto generated scripts from codegen
- ~ Drawbacks of auto generated scripts from codegen
- ~ Mouse Hover events in Playwright
- ~ Keyboard Events in Playwright
- ~ Handle Drag and Drop In Playwright
- ~ How to handle Shadow Dom
- ~ How to change view port in Playwright
- ~ How to emulate devices in Playwright
- ~ How To Perform Visual Testing Using Playwright
- ~ What is Playwright fixture
- ~ Browser Fixture
- ~ Page Fixture
- ~ Context Fixture
- ~ Request Fixture
- ~ What is Playwright Annotations and how to use them
- ~ `test.skip()`

- ~ test.fail()
- ~ test.slow()
- ~ test.only()
- ~ What is Trace Viewer In Playwright
- ~ How To Generate trace for single test
- ~ How To Generate trace for multiple test
- ~ How To analyse trace debugging
- ~ Different flags for tracing
- ~ What is Cross Browser Testing
- ~ How to perform cross browser test in Playwright -sequence
- ~ How to perform cross browser test in Playwright in parallel
- ~ Apply Retry options to execute test again
- ~ What is Test Tagging and how to add tags to your test
- ~ Group of Testcases
- ~ What is design pattern
- ~ What is POM - Page Object Model
- ~ Implement POM in Playwright
- ~ Why not to provide static data in test script
- ~ How to pass test data from json file
- ~ Data Driven Test In Playweight
- ~ What is Git and Github
- ~ Push your code from local to github
- ~ Github pull request process
- ~ Creating Branches and merge branches
- ~ What is Jenkins
- ~ Setting Up Jenkins with Email Configurations
- ~ Running Playwright test from Jenkins CI
- ~ Creating Jenkins Pipelin for Playwright Test Execution
- ~ What is Github Actions
- ~ Executing Playwright Test using Github Actions
- ~ Playwright Interview Questions
- ~ Playwright Cheatsheet
- ~ Soure Code
- ~ Automating multiple application

=> WebdriverIO :

- ~ What is WebDriverIO - WDIO
- ~ Why WebdriverIO
- ~ Selenium vs WDIO
- ~ Cypress vs WDIO
- ~ Playwright vs WDIO
- ~ Components of WDIO
- ~ Service offered by WDIO
- ~ Download Node.J and Configure on windows
- ~ Download and Install Visual Code
- ~ Install WDIO
- ~ Folder Structur
- ~ Configuration file
- ~ What is WDIO test runner
- ~ Execute the sample test
- ~ Reports in WDIO - Different Reporters
- ~ Different CLI commands
- ~ Write first WDIO script in Chrome
- ~ Write first WDIO script in Firefox
- ~ Write first WDIO script in Edge Browser
- ~ Verify URL and title
- ~ Async vs Sync
- ~ Locators In WDIO
- ~ Xpath in detail
- ~ CSS in detail
- ~ What is \$ and \$\$ and when to use and differences
- ~ How to intereact with Webelements
- ~ Handle textbox
- ~ Handle Button
- ~ Handle Links
- ~ Handle radio button and chekboxes
- ~ Handle Dropdown
- ~ How to handle non select dropdown
- ~ Handle autosuggestion
- ~ Handle Calendar
- ~ Verify element states
- ~ REPL Interface
- ~ Handle Shadow DOM
- ~ RightClick
- ~ Double Click
- ~ MouseHover
- ~ ScrollIntoView
- ~ Drag and Drop
- ~ Default matcher
- ~ Page Matcher
- ~ Element Matchers
- ~ waitFor Conditions in WDIO
- ~ waitUntil
- ~ waitForEnabled
- ~ waitForDisplayed
- ~ How to handle frames
- ~ How to handle alerts
- ~ How to handle different tabs/windows
- ~ How to include and exclude test in WDIO

- ~ *Capture Screenshots in WDIO*
- ~ *Retry failed testcases in WDIO*
- ~ *HTML Report in WDIO*
- ~ *PDF In WDIO*
- ~ *Allure Reports*
- ~ *TestData In WDIO*
- ~ *Passing different CLI flags from cmd*
- ~ *Cross Browser Testing In WDIO*
- ~ *Sequential Execution In WDIO*
- ~ *Parallel Execution In WDIO*
- ~ *Hooks In WDIO*
- ~ *before,beforeSuite,beforeHook, beforeTest,beforeCommands*
- ~ *after,afterSuite,afterHook,afterTest,afterCommand*
- ~ *What is design pattern*
- ~ *What is POM - Page Object Model*
- ~ *Adding Custom Commands*
- ~ *Implement POM in WDIO*
- ~ *Why not to provide static data in test script*
- ~ *How to pass test data in WDIO*
- ~ *What is Git and Github*
- ~ *Push your code from local to github*
- ~ *Github pull request process*
- ~ *Creating Branches and merge branches*
- ~ *What is Jenkins*
- ~ *Setting Up Jenkins with Email Configurations*
- ~ *Running Playwright test from Jenkins CI*
- ~ *Creating Jenkins Pipeline for WDIO Test Execution*
- ~ *What is Github Actions*
- ~ *Executing WDIO Test using Github Actions*
- ~ *WDIO Interview Questions*
- ~ *WDIO Cheatsheet*
- ~ *Source Code*
- ~ *Automating multiple application*

# The Ultimate Guide To OpenAI GPT-3 & Fine Tune with Custom Data

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP PROJECT

Course link : <https://ineuron.ai/course/The-Ultimate-Guide-To-OpenAI-GPT-3-&-Fine-Tune-with-Custom-Data>

## Course Description :-

Generative Pre-trained Transformer 3 (GPT-3; stylized GPT3) is an autoregressive language model that uses deep learning to produce human-like text. Given an initial text as prompt, it will produce text that continues the prompt. The architecture is a standard transformer network (with a few engineering tweaks) with the unprecedented size of 2048-token-long context and 175 billion parameters (requiring 800 GB of storage). The training method is "generative pretraining", meaning that it is trained to predict what the next token is. The model demonstrated strong few-shot learning on many text-based tasks.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access
- => Course Materials
- => Assignments

## What you will learn :-

- => Real Time Projects
- => OpenAI GPT-3 & Fine Tune with Custom data
- => What is GPT-3?
- => Demo: Build a paraphraser, Chatbot, Summarization
- => OpenAI playground to develop prompts.
- => Understand various engines and GPT-3 prompt parameters.
- => Create novel datasets with GPT-3 and Streamlit UI.
- => Zero-shot and few-shot prompts.
- => Fine tuning GPT3

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Boktiar Ahmed Bappy :

~ This is Bappy. I aim for simplicity in Data Science. Real Creativity won't make things more complex. Instead, I will simplify them, Interested in a Data Science Career and so develop myself accordingly. Data Scientist and lecturer with working experience in Machine Learning, Deep Learning, Microcontrollers and Electronics systems. Hands-on experience in classification, regression, clustering, computer vision, natural language processing and transfer learning models to solve challenging business problems. I have a huge interest in Robotics. I have innovated a lot of innovations, ideas, projects & robots and got a lot of achievements.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- The Ultimate Guide To OpenAI GPT-3 & Fine Tune with Custom Data :
  - ~ What is GPT-3?
  - ~ Demo: Build a paraphraser, Chatbot, Summarization
  - ~ OpenAI playground to develop prompts.
  - ~ Understand various engines/models and GPT-3 prompt parameters.
  - ~ Create novel datasets with GPT-3 and Streamlit UI.
  - ~ Zero-shot and few-shot prompts.
  - ~ Fine tuning GPT3
  - ~ Conclude the project
  - ~ Assignments & External Resources

# Dart Programming

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : DART

Course link : <https://ineuron.ai/course/Dart-Programming>

## Course Description :-

Learn how to write Dart programmes from the ground up. This course is designed for those who have never programmed before. Dart is a strong and expressive language with a simple learning curve. This makes it an excellent first language. Dart provides a client-optimized language, rich and powerful frameworks, and flexible tools to help you create attractive, high-quality experiences across all screens.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Dart fundamentals
- => Data types in Dart
- => Arrays
- => Maps
- => Constants
- => Operators
- => Conditionals
- => Functions
- => Object Oriented Programming in Dart
- => Asynchronous subroutines

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to Dart programming language :
  - ~ Introduction to dart
- => Hello World on MAC in Dart :
  - ~ Dart installation on MAC
- => Hello World on Windows in Dart :
  - ~ Dart installation for WINDOWS
- => Basics of Dart :
  - ~ Introduction to variables
  - ~ datatypes in dart
  - ~ array in dart
  - ~ Maps in dart
  - ~ Constants and operations
- => Conditionals in Dart :
  - ~ Introduction to if and else in dart
  - ~ Advance if else statement in dart
  - ~ Switch and case in dart

=> Functions and loops in Dart :

- ~ *basics of functions*
- ~ *Creating a calculator in dart*
- ~ *advance functions in dart*
- ~ *While loop in dart*
- ~ *Do While loop in dart*
- ~ *Solution for assignment in dart*
- ~ *for and for in loop in dart*

=> Intermediate Dart :

- ~ *More on Arrays in dart*
- ~ *More on maps in dart*
- ~ *for each loop for map in dart*

=> OOPS in Dart :

- ~ *Classes in dart*
- ~ *Objects in dart*
- ~ *constructor in Dart*
- ~ *getters and setters in dart*
- ~ *inheritance in dart*
- ~ *Interface in dart*
- ~ *Using multiple files in classes*

=> Advance Dart :

- ~ *Generics in dart*
- ~ *Code cascading in dart*
- ~ *Custom exception handling*
- ~ *Dart libraries*
- ~ *future and async in dart*
- ~ *Web server in dart*
- ~ *Making an API request in dart*

# AIOps Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS PROJECT

Course link : <https://ineuron.ai/course/AIOps-Projects>

## Course Description :-

Learn how to create a machine learning system from start to finish. Develop skills in training, deploying, scaling, and monitoring your machine learning model's performance in production. This course is specifically designed for deploying and scaling machine learning and deep learning applications.

## Course Features :-

- => Challenges
- => Various project implementation
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Design end-to-end machine learning system
- => Monitor and visualize the performance of apps
- => Build CI/CD pipelines
- => Optimizing the model training & prediction pipelines

## Requirements :-

- => Minimum System requirement: Intel Core i3 processor and 4GB RAM or Higher
- => A system with a decent internet connection
- => AWS, Azure, GCP, Digital Ocean accounts
- => Your dedication
- => Interest to learn

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

=> Linux :

- ~ Deploying flask app in EC2 Preview
- ~ Deploy Unicorn app in EC2
- ~ Configuring Nginx for Deployment
- ~ Configuring Elastic IP & SSL certificates for deployment
- ~ Deploy ML application on EC2

=> DVC :

- ~ AIOps project DVC NLP usecase part 01
- ~ AIOps project DVC NLP usecase part 02 Preview
- ~ AIOps project DVC NLP usecase part 03
- ~ AIOps project DVC NLP usecase part 04
- ~ AIOps project DVC NLP usecase part 05
- ~ AIOps project DVC NLP usecase part 06
- ~ AIOps project DVC NLP usecase part 07
- ~ AIOps project DVC NLP usecase part 08
- ~ AIOps project DVC NLP usecase part 09
- ~ AIOps project DVC NLP usecase part 10
- ~ AIOps project DVC NLP usecase part 11
- ~ AIOps project DVC NLP usecase part 12
- ~ AIOps project DVC NLP usecase part 13
- ~ AIOps project DVC NLP usecase part 14
- ~ AIOps project DVC NLP usecase part 15
- ~ Deploy ML application using DVC
- ~ Deploy computer vision application using DVC
- ~ Deploy DL application using DVC with Tensorflow
- ~ Deploy DL application using DVC with Pytorch

=> Docker :

- ~ Dockerize Python Application

- ~ Dockerize Machine Learning Application
- ~ Dockerize computer vision Application
- ~ Dockerize NLP Application
- ~ Docker Compose for multi-container deployments
- ~ Dockerize DL application build with Tensorflow
- ~ Dockerize DL application build with Pytorch

=> MLFlow :

- ~ Deploy ML application using MLFlow
- ~ Deploy vision application using MLFlow
- ~ Deploy NLP application using MLFlow
- ~ Deploy DL application on MLFlow with Tensorflow
- ~ Deploy DL application on MLFlow with Pytorch

=> Kubernetes :

- ~ Deploy ML application using Kubernetes
- ~ Deploy vision application using Kubernetes
- ~ Deploy NLP application using Kubernetes
- ~ Deploy DL application on Kubernetes with Tensorflow
- ~ Deploy DL application on Kubernetes with Pytorch

=> Kubeflow :

- ~ Deploy ML end-to-end application using Kubeflow
- ~ Deploy vision end-to-end application using Kubeflow
- ~ Deploy NLP end-to-end application using Kubeflow
- ~ Deploy DL end-to-end application on Kubeflow with TensorFlow
- ~ Deploy DL end-to-end application on Kubeflow stack with Pytorch

=> AWS MLOps :

- ~ Deploy ML application using AWS AI stack
- ~ Deploy computer vision application using AWS AI stack
- ~ Deploy NLP application using AWS AI stack
- ~ Deploy DL application on AWS AI stack with TensorFlow
- ~ Deploy DL application on AWS AI stack with Pytorch

=> Azure MLOps :

- ~ Deploy computer vision application using Azure AI stack
- ~ Deploy computer vision application using GCP AI stack
- ~ Deploy NLP application using Azure AI stack
- ~ Deploy DL application on Azure AI stack with TensorFlow
- ~ Deploy DL application on Azure AI stack with Pytorch

=> GCP MLOps :

- ~ Deploy ML application using GCP AI stack
- ~ Deploy vision application using GCP AI stack
- ~ Deploy NLP application using GCP AI stack
- ~ Deploy DL application on GCP AI stack with TensorFlow
- ~ Deploy DL application on GCP AI stack with Pytorch

=> Digital Ocean MLOps :

- ~ Deploy ML application using Digital Ocean AI stack
- ~ Deploy computer vision application using Digital Ocean AI stack
- ~ Deploy NLP application using Digital Ocean AI stack
- ~ Deploy DL application on Digital Ocean AI stack with Tensorflow
- ~ Deploy DL application on GCP AI stack with Pytorch



# Mern Stack Projects

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : WEB DEVELOPMENT PROJECT

Course link : <https://ineuron.ai/course/Mern-Stack-Projects>

## Course Description :-

Build Project with Full Stack Web Development using MERN. Build your MERN stack projects with ReactJS, Redux, Hooks and Context, NodeJS, express and MongoDB, real-time projects. Learn how to build a powerful and fully functional social media website using MERN.

## Course Features :-

- => Project Build
- => Different ways to handle design
- => Async benefits
- => Quizzes
- => Assignments
- => Completion certificate

## What you will learn :-

- => End to End Project Building
- => Starting of Project
- => Learn both Frontend & Backend
- => Testing
- => Deployment

## Requirements :-

- => Prior Knowledge in MERN Stack
- => A system with a decent internet connection
- => Your dedication

## Instructors :-

- => Syed Ashraf :
  - ~ Full Stack Engineer at TensorGo Technologies

## Curriculum details :-

- => CRUD in MERN :
  - ~ *CRUD in MERN Preview*
- => Social Media Project :
  - ~ *Introduction*
  - ~ *Initial Setup*
  - ~ *Creating Routes & Controllers*
  - ~ *Setting up Redux*
  - ~ *Handling Forms*
  - ~ *Building Post's*
  - ~ *Adding Functionalities*
  - ~ *Authentication*
  - ~ *Deployment*

# Talend

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Talend>

## Course Description :-

This Talend course will teach you how to use Talend Open Studio to make Big Data Integration easier. Top industry experts curate the Talend Big Data course. This course is based on a practical rather than a theoretical paradigm.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to Talend
- => Talend features
- => Install Talend open studio
- => Adding Lookup
- => Configuring lookup
- => Adding Database
- => Configuring db
- => sorting file
- => Joining datasources
- => Conditional based filtering

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Course Introduction :

- ~ Syllabus overview
- ~ Introduction to Talend
- ~ Talend features
- ~ Install Talend open studio

=> Practical:- Data integration task :

- ~ Overview
- ~ Configuring job
- ~ Adding metadata
- ~ Adding Lookup
- ~ Configuring lookup
- ~ Adding Database
- ~ Configuring db
- ~ sorting file
- ~ Joining datasources
- ~ Conditional based filtering
- ~ Conditional based filtering implementation

# Class 7th Physics

---

Topic Name : K12

Sub-topic Name : CLASS7

Course link : <https://ineuron.ai/course/Class-7th-Physics>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. The Physics section focuses on various concepts related to Heat, Winds, Storms, Cyclones, Electric current, etc.

## Course Features :-

=> Self Paced Videos

## What you will learn :-

=> Light

=> Motion and Time

=> Wind Storm and Cyclone

=> Heat

=> Electric Current And its Effect

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Jawala Prakash :

~

## Curriculum details :-

=> Light :

- ~ Lecture 1 : Introduction Preview
- ~ Lecture 2 : What is Light, Rectilinear Propagation of Light, Image Formation Preview
- ~ Lecture 3 : Image Formation in Plane Mirror and Spherical Mirror
- ~ Lecture 4 : Lenses, Concave and Convex Lens, Image formation in Concave and Convex Lens
- ~ Lecture 5 : Sunlight, Newton's Disc

=> Motion and Time :

- ~ Lecture 1 : Introduction Preview
- ~ Lecture 2 : Different Types of Motion Preview
- ~ Lecture 3 : Speed
- ~ Lecture 4 : Uniform and Non Uniform Motion Periodic Motion Simple Pendulum
- ~ Lecture 5 : Distance Time Graph

=> Wind Storm And Cyclone :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Air Exerts Pressure
- ~ Lecture 3 : Understanding How Winds are Produced
- ~ Lecture 4 : Thunderstorm and Cyclone

=> Heat :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Hot and Cold, Temperature
- ~ Lecture 3 : Reading a Thermometer, Clinical and Laboratory Thermometer
- ~ Lecture 4 : Conduction Convection and Radiation
- ~ Lecture 5 : Land Breeze and
- ~ Lecture 6 : Types of Clothes we Wear in Summer and Winter
- ~ Lecture 7 : NCERT Question Discussion

=> Electric Current and Its Effect :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Electric Circuit, Symbols of Electric Components, Circuit Diagram
- ~ Lecture 3 : Heating Effect of Electric Current Fuse
- ~ Lecture 4 : Electric Bell
- ~ Lecture 5 : NCERT Question Discussion

# HTML and CSS

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : HTML

Course link : <https://ineuron.ai/course/HTML-and-CSS>

## Course Description :-

This course will help you to grab the fundamentals of HTML and CSS for web development.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => HTML and DOM
- => We do not write without emmet
- => Starting with CSS
- => Mobile responsive webpages
- => CSS animation and libraries
- => Flexbox in CSS

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started :

- ~ Introduction to web dev
- ~ Get the development tools
- ~ your first hello world

=> We do not write without emmet :

- ~ Getting started with emmet
- ~ Emmet and speedy html
- ~ Parent child and grouping
- ~ emmet in css

=> HTML and DOM :

- ~ An old style blog
- ~ Why DOM is important
- ~ Inline vs Block and bring in images
- ~ Lists and interlinking pages
- ~ Getting a video on service page
- ~ 3 Plans in a table
- ~ GET and POST forms
- ~ Types of input forms

=> Starting with CSS :

- ~ Secret to learn CSS
- ~ Explore and bring in fonts
- ~ Bring in colors and styles
- ~ Transition and box shadow DOCS
- ~ Margin and padding
- ~ Button gets all and assignment

=> Working on coming soon template :

- ~ Introduction to CSS variables and new project
- ~ Browser defaults and variables
- ~ Getting more control over elements

=> Mobile responsive webpages :

- ~ *What are media queries*
- ~ *Media query in action*
- ~ *App landing page - setup*
- ~ *Navigation bar for website*
- ~ *Bring content in columns*
- ~ *Cover image in css*
- ~ *Start with flexbox*
- ~ *Buttons and columns in flexbox*
- ~ *Absolute position in CSS*
- ~ *Media query for 2 screens*

=> Register a new account :

- ~ *Handle conflict in CSS*
- ~ *Strategy and placing html*
- ~ *Classes and ID for testers*
- ~ *Where to use z index*
- ~ *Bootstrap style of CSS*
- ~ *Fixing CSS on form*
- ~ *Doing your assignment*

=> CSS animation and libraries :

- ~ *Animation and keyframes*
- ~ *Third party animation library*
- ~ *Razorpay style clipping*
- ~ *Not a payment gateway integration*

=> Flexbox in CSS :

- ~ *What is flexbox*
- ~ *Get to know the power of flexbox*
- ~ *Flexbox series-Axis and Flex direction*
- ~ *Flexbox series -justify content*
- ~ *Align items in flexbox*
- ~ *Flexbox series ordering the elements*
- ~ *Flex grow in flexbox*

# Complete ReactJS Developer Bootcamp

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : REACT

Course link : <https://ineuron.ai/course/Complete-ReactJS-Developer-Bootcamp>

## Course Description :-

This course will teach you React.js in a hands-on manner, utilizing all of the most up-to-date patterns and best practices. To become a React.js developer, you will master all of the foundations as well as advanced ideas and associated subjects. This course will provide you with a wealth of essential material and expertise, whether you are new to React or have some basic React experience.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => React templates
- => React states
- => React hooks
- => React context API
- => Firebase integration
- => Redux Apps

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction :
  - ~ Into to REACTJS course
- => After launch updates :
  - ~ React router v6
- => Getting started with ReactJS :
  - ~ How to use exercise files
  - ~ What is react and myths
  - ~ Tools that we need
- => Going All classic :
  - ~ Section 2 introduction
  - ~ Finishing the hello world task
  - ~ Delete and recreate everything
  - ~ Adding CSS to our Hello World
  - ~ Everything in its own file
  - ~ Reusable components
- => Create a react template :
  - ~ Section 3 introduction
  - ~ Understand the existing template
  - ~ Move navbar and understand the errors
  - ~ Convert the HTML template into React App
  - ~ Reusable Card and Assignment
- => Getting friendly with states :
  - ~ Section 4 introduction
  - ~ What are props and states

- ~ *Preparing the state based applications*
- ~ *Complete counter application*
- ~ *Assignment for counter app*

=> Building a Tic Tac Toe :

- ~ *section 5 Introduction*
- ~ *Your need to study first*
- ~ *Preparing the Tic Tac Toe*
- ~ *Sending icons from components*
- ~ *Setup layout for tictactoe*
- ~ *Game is almost working*
- ~ *Finishing tictactoe and assignment*

=> Learn React Context API with projects :

- ~ *Section 6 Introduction*
- ~ *The problem that contextAPI solves*
- ~ *Detail on Context and Provider*
- ~ *Detail on Consumer in contextAPI*
- ~ *Understand the working of dark and light mode*
- ~ *Creating a theme Toggler with Context API*
- ~ *Finishing the theme switcher app*

=> App with Context API with reducers and actions :

- ~ *Section 7 introduction*
- ~ *What are we building here*
- ~ *Create brain of the application*
- ~ *useReducer for our app*
- ~ *Add an input form*
- ~ *Sending a dispatch*
- ~ *Display the context data and dispatch*

=> Local storage and useEffect hooks :

- ~ *Section 8 introduction*
- ~ *Introducing the Effect hook*
- ~ *A form to submit the data*
- ~ *Looping through all the values*
- ~ *Hooks and local storage in action*

=> Learn to handle API :

- ~ *Section 9 introduction*
- ~ *Learn to read docs for API*
- ~ *lets read Axios docs*
- ~ *Drill down the API*
- ~ *Extracting information from API*

=> Designing a shopping cart API :

- ~ *Section 10 introduction*
- ~ *A walk through Pexels and JSON*
- ~ *Add item to the cart*
- ~ *Buy item and remove item*
- ~ *Fetching photos from API*
- ~ *Store everything in state*
- ~ *Card for every product*
- ~ *Create cart section*
- ~ *Bring the shop together*
- ~ *Removing the duplicate*

=> Firebase with Github App :

- ~ *Section 11 introduction*
- ~ *What we are about to build*
- ~ *React Router crash course*
- ~ *Your tour to configure firebase*
- ~ *Read firebase docs with me*
- ~ *Creating components for firebase app*
- ~ *Bring in the react router*
- ~ *Headers and Footers*
- ~ *Conditional rendering in Navbar*
- ~ *Adding firebase configuration*
- ~ *User Signup in firebase*
- ~ *Logout and signin user*
- ~ *User card component*
- ~ *Repo component*
- ~ *Home page and finish the app*

=> Firebase real time database :

- ~ *Section 12 introduction*
- ~ *A challenge application*
- ~ *Firebase real time database*
- ~ *Setting context and actions*
- ~ *Creating reducers for contact*
- ~ *Header and Footer tasks*
- ~ *How to upload image in firebase storage*
- ~ *Add and update contact in firebase*
- ~ *Add or update finder*
- ~ *Update star and delete contact*
- ~ *Use dispatch and FIXME*
- ~ *Get all data from firebase*
- ~ *Loop through firebase object*
- ~ *Firebase finale and assignment*

=> Bonus-Redux App :

- ~ 3 Principles of redux
- ~ Bring in the central state
- ~ Actions make redux simpler
- ~ Reducer - brain part of app
- ~ Component dispatching the info
- ~ 2 most important method for Redux
- ~ Provider to give access of store
- ~ Finally creating that store

=> More bonus stuff -Extra production tips :

- ~ Axios optimise API calls

=> Bonus updates :

- ~ React 18 updates



# Machine Learning And Deep Learning Masters

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-And-Deep-Learning-Masters>

## Course Description :-

This is Machine Learning masters and Deep Learning, where you will learn various things from beginning like python , API , deployment in Aws , azure , GCP , Heroku , database , various modules in statistics ,all machine learning algorithm , four mode of Chabot live Dialog flow , Amazon Lex , Azure Luis and RASA NLU , and 15+ live project all together in live instructor led class along with various mode of support and services and doubt clearing session.

## Course Features :-

- => Machine Learning in depth from beginning to advance discussion and implementation with Deployment.
- => Deep learning in-depth topic wise discussion and implementation with the project.
- => Docker and Kubernetes end to end with CI/CD pipeline for machine learning.
- => End to End Model Deployment in Azure, GCP, AWS, and Pivotal Cloud.
- => Python spark implementation with the project.
- => Time Series end to end implementation in machine learning and deep learning.
- => 26 + hands-on industry real-time projects.
- => Power BI and Tableau self-placed course.
- => Machine Learning Deep Learning Masters Certificate
- => 200 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access.
- => Doubt clearing one to one
- => Doubt clearing through mail and support team
- => Assignment in all the module
- => 20+ use case of Machine learning
- => A live project with real-time implementation
- => Resume building
- => career guidance
- => interview Preparation
- => Regular assessment
- => Job alerts
- => Online Instructor-led learning: Live teaching by instructors
- => Product Demo

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Data analytics
- => Mock interview
- => Interview preparation
- => Resume building

## Requirements :-

- => Dedication
- => Laptop with internet connectivity

## Instructors :-

- => krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course Introduction :

- ~ Introduction of Data science and its application in Day to Day life
- ~ Course overview and Dashboard description

=> Python Core :

- ~ Introduction of python and comparison with other Preview
- ~ Programming language
- ~ Installation of Anaconda Distribution and other python
- ~ IDE Python Objects, Number & Booleans, Strings Preview
- ~ Container objects, Mutability of objects
- ~ Operators Arithmetic, Bitwise, Comparison and Assignment operators, Operators Precedence and associativity
- ~ Conditions(if else, if elif else) Loops(While ,for)
- ~ Break and Continue statement and Range Function.

=> String Objects and collections :

- ~ String object basics
- ~ String methods
- ~ Splitting and Joining Strings
- ~ String format functions
- ~ List object basics
- ~ List as stack and Queues
- ~ List comprehensions

=> Tuples, Set, Dictionaries Functions :

- ~ Tuples, Sets Dictionary Object basics, Dictionary Object methods, Dictionary View Objects.
- ~ Functions basics, Parameter passing, Iterators Generator functions
- ~ Lambda functions
- ~ Map, Reduce, Filter functions

=> OOPS concepts Working with Files :

- ~ OOPS basic concepts
- ~ Creating classes and Objects Inheritance
- ~ Multiple Inheritance
- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other File methods

=> Exception Handling :

- ~ Exceptions Handling with Try except

=> Api :

- ~ Flask introduction
- ~ Flask Application
- ~ Open link Flask
- ~ App Routing Flask
- ~ URL Building Flask
- ~ HTTP Methods Flask

=> Database :

- ~ Mongo DB SQL
- ~ Lite python SQL

=> Python pandas Modules :

- ~ Python Pandas Series
- ~ Python Pandas DataFrame
- ~ Python Pandas Panel
- ~ Python Pandas Basic functionality

=> Python Numpy :

- ~ NumPy Narray Object
- ~ NumPy Data Types
- ~ NumPy Array Attributes
- ~ NumPy Array Creation Routines
- ~ NumPy Array from Existing
- ~ Data Array From Numerical Ranges
- ~ NumPy Indexing & Slicing
- ~ NumPy Advanced Indexing
- ~ NumPy Broadcasting
- ~ NumPy Iterating Over Array
- ~ NumPy Array Manipulation
- ~ NumPy Binary Operators
- ~ NumPy String Functions
- ~ NumPy Mathematical Functions
- ~ NumPy Arithmetic Operations
- ~ NumPy Statistical Functions
- ~ Sort, Search & Counting Functions
- ~ NumPy Byte Swapping
- ~ NumPy Copies Views
- ~ NumPy Matrix Library
- ~ NumPy Linear Algebra

=> Exploratory Data Analysis :

- ~ Feature Engineering and Selection
- ~ Building Tuning and Deploying Models

- ~ *Analyzing Bike Sharing Trends*
- ~ *Analyzing Movie Reviews Sentiment*
- ~ *Customer Segmentation and Effective Cross Selling*
- ~ *Analyzing Wine Types and Quality*
- ~ *Analyzing Music Trends and Recommendations*
- ~ *Forecasting Stock and Commodity Prices*

#### => Statistics :

- ~ *Descriptive Statistics*
- ~ *Sample vs Population statistics Random Variables*
- ~ *Probability distribution function Expected value*
- ~ *Binomial Distribution*
- ~ *Normal Distribution z score*
- ~ *Central limit Theorem*
- ~ *Hypothesis testing Z Stats vs T stats*
- ~ *Type 1 type 2 error*
- ~ *Confidence interval*
- ~ *Chi Square test*
- ~ *ANOVA test*
- ~ *F stats*

#### => Machine Learning 1 :

- ~ *Introduction*
- ~ *Supervised , Unsupervised, Semi supervised, Reinforcement Train , Test, Validation Split*
- ~ *Performance Overfitting , underfitting OLS.*
- ~ *Linear Regression assumption.*
- ~ *R square adjusted*
- ~ *R square Intro to Scikit learn*
- ~ *Training methodology*
- ~ *Hands on linear regression*
- ~ *Ridge Regression*
- ~ *Logistics regression*
- ~ *Precision Recall ROC curve*
- ~ *F Score*

#### => Machine Learning 2 :

- ~ *Decision Tree Cross*
- ~ *Validation Bias vs Variance*
- ~ *Ensemble approach Bagging*
- ~ *Boosting Random*
- ~ *Forest Variable Importance*

#### => Machine Learning 3 :

- ~ *XGBoost*
- ~ *Hands on XgBoost*
- ~ *K Nearest Neighbour*
- ~ *Lazy learners*
- ~ *Curse of Dimensionality*
- ~ *K NN Issues*
- ~ *Hierarchical clustering K Means*
- ~ *Performance measurement*
- ~ *Principal Component analysis*
- ~ *Dimensionality reduction*
- ~ *Factor Analysis*

#### => Machine Learning4 :

- ~ *SVR*
- ~ *S V M*
- ~ *Polynomial Regression*
- ~ *Ada boost*
- ~ *Gradient boost*
- ~ *Gaussian mixture*
- ~ *Anomaly detection*
- ~ *Novelty detection algorithm Stacking*
- ~ *K NN regressor*
- ~ *Decision tree regressor DBSCAN*

#### => Natural Language Processing :

- ~ *Text Analytics*
- ~ *Tokenizing , Chunking*
- ~ *Document term*
- ~ *Matrix TFIDF*
- ~ *Sentiment analysis hands on*

#### => Spark :

- ~ *Spark overview.*
- ~ *Spark installation.*
- ~ *Spark RDD.*
- ~ *Spark dataframe .*
- ~ *Spark Architecture.*
- ~ *Spark ML lib.*
- ~ *Spark Nlp*
- ~ *Spark linear regression.*
- ~ *Spark logistic regression.*
- ~ *Spark Decision Tree.*
- ~ *Spark Naive Bayes*
- ~ *Spark xg boost*
- ~ *Spark time series.*
- ~ *Spark Deployment in local server*
- ~ *Spark job automation with scheduler.*

=> Deep Learning :

- ~ Deep Learning Introduction.
- ~ Neural Network Architecture.
- ~ Loss Function.
- ~ Cost Function.
- ~ Optimizers.
- ~ CNN architecture.
- ~ Build First Classifier in CNN.
- ~ Deploy Classifier over cloud.
- ~ RNN overview.
- ~ GRU.
- ~ LSTM.
- ~ Time Series using RNN LSTM.
- ~ Customer Feedback analysis using RNN LSTM.

=> Time Series :

- ~ Arima
- ~ Sarima .
- ~ Auto Arima
- ~ Time series using RNN LSTM .
- ~ Prediction of NIFTY stock price.

=> Deployment :

- ~ Deployment of all the project In cloudfoundary , AWS AZURE and Google cloud platform
- ~ Expose api to web browser and mobile application retraining a pproach of Machine learning model
- ~ Devops infrastructure for machine learning model
- ~ Data base integration and scheduling of machine learning model and retraining c ustom machine learning training approach.
- ~ AUTO ML
- ~ Discussion on infra cost and data volume
- ~ P rediction based on streaming data

=> Extra session :

- ~ Discussion on project explanation in interview
- ~ Data scientist roles and responsibilities
- ~ Data scientist day to day work
- ~ Companies which hire a data scientist
- ~ Resume discussion with our team one to one

=> Tableau and power Bi self placed session :

- ~ Business Intelligence (BI) Concepts.
- ~ Microsoft Power BI (MSPBI) introduction.
- ~ Connecting Power BI with Different Data sources.
- ~ Power Query for Data Transformation.
- ~ Data Modelling in Power BI.
- ~ Reports in Power BI Reports and Visualisation types in Power BI.
- ~ Dashboards in Power BI.
- ~ Data Refresh in Power BI.
- ~ Traditional Visualisation(Excel) vs Tableau.
- ~ About Tableau.
- ~ Tableau vs Other BI Tool Pricing.

=> Tableau Interview Questions.

## Project details :-

=> Python project :

- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis
- ~ Integration with web portal
- ~ Integration with rest a A pi W eb portal and Mongo DB on Azure
- ~ Deployment on web portal on Azure
- ~ Text mining
- ~ Social media data churn

=> Chatbot Project :

- ~ Chatbot using Microsoft Luis
- ~ Chatbot using google Dialog flow
- ~ Chatbot using Amazon Lex
- ~ Chatbot using Rasa NLU
- ~ Deployemnt of chatbot with web , Telegram , Whatsapp , Skype

=> Machine learning project :

- ~ Healthcare analytics prediction of medicines based on FIT BITband
- ~ Revenue forecasting for startups
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ Anamoly detection in inventory packaged material.
- ~ Fault detection in wafferes based on sensordata
- ~ Demand forecasting for FMCG product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with Zomato dataset.
- ~ Fault detection in wafferes based on sensor data.
- ~ Cement\_Strength \_ reg.
- ~ Credit Card Fraud.
- ~ Forest\_Cover\_ Classification .
- ~ Fraud Detection.
- ~ Income Prediction.
- ~ Mushroom classifier., Phising Classifier , Thyroid\_Detection .
- ~ Visibility climate.

=> Deep Learning projects :

- ~ Customer Feedback analysis using RNN LSTM.

- ~ *Family member detection.*
- ~ *Industry financial growth prediction.*
- ~ *Speech recognition based attendance system.*
- ~ *Vehicle Number plate detection and recognition system.*

=> Tableau and power Bi Projects :

- ~ *Project 1. Project Sales.*
- ~ *Project 2. Financial Report.*
- ~ *Project 3. HealthCare.*
- ~ *Project 4. Procurement Spend Analysis.*
- ~ *Project 5. Human Resource Tableau*

# Complete Angular Developer Bootcamp

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : ANGULAR JS

Course link : <https://ineuron.ai/course/Complete-Angular-Developer-Bootcamp>

## Course Description :-

Do you want to develop front-end applications more quickly and with less code that are robust, manageable, and testable? Then think about enrolling in this course to learn about AngularJS, one of the most widely used Single Page Application (SPA) frameworks available today. Because it was created and is supported by Google, AngularJS is extremely marketable expertise.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Getting started with angular
- => TypeScript that you need for Angular
- => Let's Build counter
- => Core foundation of angular apps
- => Generator - user input
- => Game- passing info to parent
- => RxJs fundamentals
- => Project:- Form service and pipe in angular project
- => Project :- SignUp reactive form in Angular
- => Project:- Web request and API in Angular
- => Project:- Fire base login and github searcher

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with angular :

- ~ Angular Section 1 Intro
- ~ Tools and installation for Angular
- ~ NG command line utility
- ~ Create your first angular application
- ~ Angular file structure
- ~ Official Hello to Angular app
- ~ Online editors -A word of caution

=> TypeScript that you need for Angular :

- ~ Angular section 2 intro
- ~ Types in TypeScript
- ~ Conditionals in TypeScript
- ~ Looping through array
- ~ Functions and Parameters in TypeScript
- ~ Interface in TypeScript
- ~ Class constructor and Interface
- ~ Decorators- Best explanation you will see

=> Let's Build counter :

- ~ Angular section 3 intro

- ~ Counter app assets and naming
- ~ Creating HTML interface for the counter app
- ~ Handling counter logic
- ~ Injecting class into template
- ~ Event binding and interpolation

=> Core foundation of angular apps :

- ~ Angular section 4 intro
- ~ Delete everything in project
- ~ What is main ts file
- ~ Creating a module in Angular
- ~ Inject decorator in Module
- ~ Inject decorator in Component
- ~ Polyfills and running the app
- ~ File separation for component
- ~ Injecting second component

=> Generator - user input :

- ~ Angular section 5 intro
- ~ Basics about user input and routing
- ~ Load CSS, Bootstrap and Custom assets
- ~ Logic part of word application
- ~ Finish word generator and assignment

=> Game- passing info to parent :

- ~ Angular section 6 intro
- ~ Prototype and reading docs
- ~ Creating a mistake and custom CSS
- ~ Winning logic and package config
- ~ Input decorator in Angular
- ~ Angular switch and case
- ~ Winning and reset logic in Angular
- ~ Detect clicks and custom messages in Angular
- ~ Reset the game
- ~ Ng For loop and property binding in Angular
- ~ Small CSS fix - optional

=> RxJs fundamentals :

- ~ Angular section 7 intro
- ~ Problem that RxJs is trying to solve
- ~ Comparing regular Js and RxJs
- ~ Understand the flow in RxJs
- ~ What is observable in RxJs
- ~ What are observers in RxJs
- ~ Subscribe and Unsubscribe to events
- ~ Pipe and operators in RxJs

=> Project:- Form service and pipe in angular project :

- ~ Angular section 8 intro
- ~ Building todo - form and service in angular
- ~ Creating app structure for todo in Angular
- ~ Creating model for todo in Angular
- ~ Angular pipe in Action
- ~ Service - Business logic of Angular app
- ~ Life cycle hooks in Angular
- ~ NgClass and NgFor todo Angular
- ~ Reactive form and template driven form
- ~ Adding form in module Angular
- ~ 2 way binding syntax in Angular
- ~ Angular wrapper elements

=> Project :- SignUp reactive form in Angular :

- ~ Angular section 9 intro
- ~ Building a signup form and validations
- ~ Adding reactive form and bootstrap
- ~ Brain part of reactive form in Angular
- ~ Building custom form validators
- ~ Understand the basic signup form template
- ~ Connect form with validators
- ~ Render error messages to users

=> Project:- Web request and API in Angular :

- ~ Angular section 10 Intro
- ~ HTTP module in Angular
- ~ Generating components and services for users
- ~ Adding httpClientModule to app
- ~ Injecting httpClient in Angular
- ~ Make a web request in Angular
- ~ Accept data from parent as user
- ~ Getting API response and displaying it
- ~ One more thing about ngContent

=> Project:- Fire base login and github searcher :

- ~ Angular section 11 intro
- ~ reading routing docs and layout in Angular
- ~ Understand the project structure
- ~ Reading fire base docs
- ~ Creating a new firebase project
- ~ Config project to firebase
- ~ Generating file structure for github app
- ~ Bring everything in Module for Angular

- ~ signup and sign in and getUser from firebase
- ~ Working with GitHub service API
- ~ Footer for github Angular
- ~ Fixing bugs and header brain
- ~ Header template with router
- ~ Detect changes in grand child
- ~ User card for git
- ~ Finishing home component
- ~ Page not found
- ~ Signup with firebase for git
- ~ Protecting routes and routing
- ~ Sign In with firebase for git
- ~ Minor debugging and Final github searcher

=> Project :- Social Media and - Insta Inspired :

- ~ Angular section 12 intro
- ~ Social media mockup -intro
- ~ Understand database and storage
- ~ Understand database and architecture
- ~ Installing tools that we need
- ~ Generating all components for travelgram
- ~ Building firebase services for travelgram
- ~ Header and footer of the application
- ~ Signup with DB entry in travelgram
- ~ How to upload images or any resources in database
- ~ Conditional rendering of signup template
- ~ Setting up routing for travelgram
- ~ Router and lots of debugging
- ~ Signs in is easy now
- ~ Add post by user
- ~ Home component with a BUG
- ~ Adding list of users
- ~ Like and dislike the post
- ~ Like and dislike with changes



# NLP Interview Questions

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP INTERVIEW

Course link : <https://ineuron.ai/course/NLP-Interview-Questions>

## Course Description :-

If you are looking for a place to be prepared for your upcoming interviews in Natural Language Processing, then this course is for you. We will discuss all kinds of NLP-based questions that can be asked in an interview and, it will also help you in boosting your confidence. Artificial Intelligence Operations is the most in-demand technical skill (AIOps). It facilitates the use of DevOps techniques in the creation of AI products. This course will cover a variety of approaches to implementing AIOps methodology in machine learning and deep learning projects, including implementation on AWS, Azure, Google Cloud Platform, and DigitalOcean.

## Course Features :-

- => Interview based question discussions
- => Completion certificate
- => Downloadable resources

## What you will learn :-

- => Learn to solve and answer different NLP based Interview questions
- => Tackle any Interview
- => Project Pipeline Algorithm , Selection Building Solutions

## Requirements :-

- => Prior knowledge in NLP
- => A System with internet connection
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Interview Questions :
  - ~ NLP Question Discussion part 1 Preview
  - ~ NLP Question Discussion part 2

# C++ Bootcamp for Beginners

---

Topic Name : PROGRAMMING

Sub-topic Name : C++

Course link : <https://ineuron.ai/course/C++-Bootcamp-for-Beginners>

## Course Description :-

Ultimate modern C++ Bootcamp. A modern approach to understand C++. By mastering the fundamentals of the language, you can start writing C++ programmes right away. Additionally, you will hear several methods and viewpoints on using C++ professionally.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to CPP
- => Getting Started with CPP
- => Basics but indepth of CPP
- => Functions in CPP
- => Object Oriented Programming
- => Smart Pointers in CPP

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction to CPP :

- ~ CPP20 A note
- ~ CPP20 section1
- ~ Welcome to Cpp bootcamp
- ~ Prerequisite and tools for cpp
- ~ Understand the entry point of hello world
- ~ Compare the 2 hello world
- ~ Version history and official documentation of cpp

=> Getting Started with CPP :

- ~ CPP20 section2
- ~ Return type and comments
- ~ Redefining program in cpp
- ~ What is namespace in cpp
- ~ First iteration of program
- ~ Can I name that
- ~ Get the color and assignment

=> A Little fast pace CPP :

- ~ CPP20 section3
- ~ Your first introduction to pointers
- ~ Reference is the actual tough thing in cpp
- ~ Cpp array are different with pointers
- ~ A formal introduction to integers
- ~ Conditionals and ternary
- ~ Conditionals as switch
- ~ While and do while loops
- ~ Introduction to for and range based for loops
- ~ Loop with pointers and shortcuts

=> Basics but indepth of CPP :

- ~ CPP20 section4

- ~ Always use float with caution
- ~ Why always divide by zero for try catch block
- ~ Sneek peek to functions in cpp
- ~ linkers qualifiers prefix and postfix
- ~ Basics of operations on cpp
- ~ Logical AND OR and NOT
- ~ bitwise operation in cpp
- ~ Memory leaks in cpp

=> More datatypes in CPP :

- ~ CPP20 section5
- ~ Get started with structs in cpp
- ~ Enums and Preprocessors
- ~ A challenge to strongly types language
- ~ Heap and Stack memory with a version discussion

=> Functions in CPP :

- ~ CPP20 section6
- ~ Detailed introduction to functions
- ~ How to create a header file in cpp
- ~ Your first introduction to templates
- ~ What are functional pointers
- ~ nullptr saves the day
- ~ Factorial and recursion are close friend
- ~ Lets talk about MACROS
- ~ Variadic templates and recursion

=> Object Oriented Programming :

- ~ CPP20 section7
- ~ A design example
- ~ Get started with class and objects
- ~ Getters and Setters for a data member
- ~ Method separation and const qualified methods
- ~ Constructor destructor and rule of 3
- ~ Disable the constructor
- ~ THIS is not easy in cpp

=> Little more OOPS :

- ~ CPP20 section8
- ~ Inheritance is my favourite
- ~ Base class Derived class and overriding
- ~ Friend keyword come with caution
- ~ Multiple Inheritance
- ~ polymorphism and virtual

=> Smart Pointers in CPP :

- ~ CPP20 section9
- ~ What are smart pointers
- ~ Unique pointers and issues
- ~ Shared pointers in smart pointers
- ~ Weak pointers in smart pointers

=> Move Semantics file & lambda :

- ~ CPP20 section10
- ~ Move semantics Lvalue and Rvalue
- ~ Vectors - Dynamic array from STD template library
- ~ Lambda - a small hello
- ~ Create, rename and delete files
- ~ Reading and writing into files and MODES

=> STL - Standard Template Library :

- ~ CPP20 section11
- ~ Introduction to STL and generic programming
- ~ Main components in STL
- ~ Functors in STL
- ~ SORT algorithms in STL
- ~ SEARCH algorithms in STL
- ~ Partition and Stable partition in STL

=> STL - a little more :

- ~ CPP20 section12
- ~ Revisiting vectors in STL
- ~ List in STL
- ~ Queue and priority queue in STL
- ~ Deque in STL
- ~ Stack in STL and assignment
- ~ Sets and MultiSets in STL
- ~ MAPS and assignment

# Linux

---

Topic Name : DEVOPS

Sub-topic Name : LINUX

Course link : <https://ineuron.ai/course/Linux>

## Course Description :-

This Linux course looks at the tools and techniques that Linux system administrators and end-users use on a daily basis to complete their tasks in a Linux environment.

## Course Features :-

- => Source Code
- => Downloadable Resources
- => Quiz Questions
- => Completion Certificate

## What you will learn :-

- => Linux Introduction
- => Setting up Our Linux Space
- => Linux Concepts
- => Package Management
- => Linux Commands
- => Working with Terminal
- => Permissions & Security

## Requirements :-

- => A system with Internet Connection
- => Your dedication

## Instructors :-

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Linux Introduction :

- ~ Introduction to Linux Preview
- ~ What is Linux
- ~ Important Pieces in Linux
- ~ Features of Linux
- ~ Evolution of Linux
- ~ Differences between Windows and Linux

=> Setting up Our Linux Space :

- ~ Downloading Necessary tools Preview
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH Clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting Up Passwordless SSH

=> Linux Concepts :

- ~ What is Kernel Preview
- ~ Types of Kernel
- ~ What is Shell
- ~ Types of Shell
- ~ Distro in Linux
- ~ Linux Boot Process
- ~ File System
- ~ RunLevels in Linux
- ~ Filetypes of Linux

=> Package Management :

- ~ Package Management
- ~ Package Managers & DPKG
- ~ Working with APT & APT GET
- ~ Apt-get Advanced Part 1
- ~ Apt-get Advanced Part 2

=> Linux Commands :

- ~ *Linux Commands Part1*
- ~ *Linux Commands Part2*
- ~ *Linux Commands Part3*
- ~ *Linux Commands Part4*
- ~ *Cat Command Usages*

=> Working with Terminal :

- ~ *File Archival*
- ~ *File Compression*
- ~ *Files and Patterns Search*
- ~ *Input-Output Redirection*
- ~ *Working with Vi Editor*
- ~ *Advanced Vi Editor Part 1*
- ~ *Advanced Vi Editor Part 2*

=> Permissions & Security :

- ~ *Types of Account in Linux*
- ~ *User Management*
- ~ *Group Management*
- ~ *Files Access Controls*
- ~ *Linux File Permissions*
- ~ *Modifying File Ownership*
- ~ *Sudoers in Linux*
- ~ *Cronjobs*
- ~ *SCP*
- ~ *Special Permissions*
- ~ *System Management*
- ~ *System tools*
- ~ *Hard link and Soft link*
- ~ *Aliasing in Linux*
- ~ *Creating users in Multiple ways*

=> Linux in AWS Cloud- Deploy an App in EC2 :

- ~ *Launching an Ubuntu VM and SSH Setup*
- ~ *Package installation in VM*
- ~ *Running our Calculator App*
- ~ *Gunicorn & Nginx Setup*
- ~ *Creating a Gunicorn Service*
- ~ *Attaching an Elastic IP*
- ~ *Attaching OpenSSL Certificates for HTTPS*

# Apache Atlas

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Apache-Atlas>

## Course Description :-

Apache Atlas provides open metadata management and governance capabilities for organizations to build a catalog of their data assets, classify and govern these assets and provide collaboration capabilities around these data assets for data scientists, analysts, and the data governance team.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduces Apache Atlas
- => Apache Atlas Installation
- => Walkthrough of Apache Atlas Console
- => Terminologies in Apache Atlas
- => Data Lineage in Apache Atlas
- => Classification in Apache Atlas
- => Basic and Advanced Search
- => Glossary in Apache Atlas
- => REST APIs in Apache Atlas
- => Practical use of REST APIs
- => Apache Atlas Internals
- => Apache Atlas at ING

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shruti Mantri :

~ Shruti Mantri is a well-known software architect, instructor and mentor in the industry. She has 10+ years of experience in the software industry, and has worked with different organizations like Oracle, Flipkart, Amazon, Myntra and Twitter. She is known for her expertise in the data engineering field, and has a sound knowledge on the latest technologies in this domain. She has helped develop data platform at organizations, and guided several mentees in understanding data engineering and how to get better at it.

## Curriculum details :-

=> Introduction :

- ~ Introduction
- ~ Course Objectives

=> Installation :

- ~ Atlas Installation
- ~ Loading Sample Data

=> Terminologies :

- ~ Types and Entities
- ~ Relationships
- ~ Attributes
- ~ System specific types
- ~ Data lineage
- ~ Classification
- ~ Classification propagation

=> Atlas UI :

- ~ *Basic Search in Atlas UI*
- ~ *Advanced search in Atlas UI*
- ~ *Glossary*

=> **REST APIs :**

- ~ *REST APIs in Atlas*
- ~ *Precap to Hands-on*
- ~ *Creating Entity Type Definitions*
- ~ *Creating Relationship Type Definitions*
- ~ *Creating Entities*
- ~ *Creating Relationships*
- ~ *Creating Data Lineage*
- ~ *Creating Classification*

=> **Internals :**

- ~ *Apache Atlas Internals*

=> **Use-cases :**

- ~ *Industry Use-case: Apache Atlas at ING*

# Youtube Mastery Course in Hindi Tech Neuron

---

Topic Name : DIGITAL MARKETING

Sub-topic Name : YOUTUBE MARKETING

Course link : <https://ineuron.ai/course/Youtube-Mastery-Course-in-Hindi-Tech-Neuron>

## Course Description :-

Video Mastery Course Is To Learn About How You can Work On Youtube And Grow Your channel To earn money through your videos, you will have to have a Google AdSense account. Dont have a Google AdSense account? You are able to create a new Google AdSense account with your YouTube channel.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Creating Your Videos Content
- => Useful Apps for Youtubers
- => Kinemaster Expert
- => Photoshop Graphic Designing
- => Youtube Short Videos
- => Filmora
- => Canva
- => INSTAGRAM
- => Photoshop Graphic Designing
- => FACEBOOK

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Why YouTube As a platform & How Youtube Works? :
  - ~ What is YouTube and how to earn through it?
  - ~ What is Amazon Affiliate Through Youtube?
  - ~ Youtube as a profession and business
  - ~ Sponsorship through Youtube & another Multiple way of learning
- => Channel Creation and Learn Seo Identify your Target Audience :
  - ~ Creating Youtube Channel
  - ~ How to Verify your Youtube Channel and apply custom Thumbnail?
  - ~ Decide Your Unique name of YouTube Channel
  - ~ Making Professional Channel Art
  - ~ How To Make Your Channel Brand New Logo?
  - ~ Complete Youtube Creators Studio
  - ~ Youtube A to Z Complete Step by Step Settings
  - ~ How To Upload Youtube Video With 100% SEO?
  - ~ Best Mic For Your Video & How To Use it
  - ~ Best Camera for Video
  - ~ How To Decide Your Profitable Niche?
- => Creating Your Videos Content :
  - ~ How To Write an Effective Script for your videos?
  - ~ How do the best creators produce content consistently?
  - ~ well-researched and steadily consistent videos
  - ~ Learn the secrets of high-quality channel
- => Important Websites & Tools for Youtube :
  - ~ How to get Copyright free videos and IMAGES?
  - ~ Learning to use Google Fonts
  - ~ Using Google Ads to promote Youtube Videos



- ~ Complete tutorial of Social Blade, TubeBuddy tutorial, VidIQ tutorial
- ~ Creative Commons Licence

#### => Useful Apps for Youtubers :

- ~ Best Video Editing App
- ~ Best Audio Editing App
- ~ Best Screen Recorder App
- ~ Youtube Video Analytics
- ~ Best Tag Finder For Youtube Video
- ~ Lower Third For Youtube Video
- ~ How To Make Professional Thumbnail for Your Youtube Video?
- ~ How To Make Professional Intro or Outro For YOUTUBE Video?
- ~ TubeBuddy For Tags

#### => Free Youtube Video, Audio etc. :

- ~ Download Audacity software ,install and use it
- ~ Type of Mic
- ~ YouTube Equipment Setup
- ~ How to set Chroma Properly

#### => 50 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Teaching
- ~ Elementary class
- ~ Middle class
- ~ High School
- ~ Professional Skills-
- ~ Cooking,
- ~ Yoga
- ~ Sales
- ~ Marketing
- ~ Startups
- ~ Digital Marketing
- ~ Affiliate Marketing
- ~ Fitness Channels
- ~ Product Fit Without Gym
- ~ Comedy Channels
- ~ Vines
- ~ Group
- ~ Unboxing Videos
- ~ Bollywood Gossip Video
- ~ Sports Video
- ~ Fashion
- ~ Mens
- ~ Womens
- ~ Music and music tutorial channel
- ~ Food Challenges
- ~ Vlogs channel
- ~ Timelapse channel
- ~ Copy Paste channel
- ~ Interviews channel
- ~ Property channel
- ~ Facts Video - Day to Day
- ~ Life Hacks
- ~ Beauty product video
- ~ Product Comparison
- ~ Video Editing channel
- ~ Graphic Designing channel
- ~ Film Your Podcast
- ~ Give Business Advice
- ~ Hair Tutorial channel
- ~ Clothes Tips channel
- ~ Health Fruits channel
- ~ New Channel-Shot Film
- ~ Food Vlogging channel
- ~ Startup news-biopic
- ~ Case Studies
- ~ Charts/Graphs
- ~ Ebooks reader
- ~ Cartoons/Illustrations
- ~ Book Summaries
- ~ Tool Reviews
- ~ Online earning application review channel
- ~ Share market
- ~ Personal finance
- ~ Infographics
- ~ Mind Maps
- ~ Online Game
- ~ Podcasts
- ~ Quotes
- ~ Quizzes
- ~ Cricket News Channel
- ~ Cricket Prediction
- ~ Baby Care Channel

#### => Youtube Short Videos :

- ~ Understand Youtube Shorts Algorithm
- ~ Youtube Shorts Video Uploading
- ~ How To Shoot & Edit Youtube Shorts Videos?
- ~ How To Upload Youtube Shorts Video?
- ~ Youtube Shorts Video Complete 100% Seo

=> Get 1000 subscribers in 90 days and many more tips :

- ~ *How to get 1000 subscribers and 4000 hours watch time in 2022?*
- ~ *Rank as no. 1 Youtuber- Youtube seo step by step tutorial*
- ~ *Making videos trending in less views*
- ~ *Becoming digital business growth consultant*
- ~ *Engaging people in youtube channels*
- ~ *How to Increase watch time?*
- ~ *Starting from 0 subscribers*
- ~ *Why youtubers fail to accomplish?*

=> Pre Production Work Editing From Mobile Apps Power Director :

- ~ *Power Director Tutorial*
- ~ *Power Director Tutorial part 1*
- ~ *Power Director Tutorial part 2*

=> Kinemaster Expert :

- ~ *Basic of Kinemaster Expert*
- ~ *Important Setting of Kinemaster Expert*
- ~ *Learning Text Tool in Kinemaster Expert*
- ~ *Use transition Tool in Kinemaster Expert*
- ~ *Learning to use Pan and Zoom in Kinemaster Expert*
- ~ *Remove Green Screen in Kinemaster Expert*
- ~ *Learning to Save Video in High Quality*
- ~ *How to add Subscribers button in Channel?*
- ~ *How to use colour filters and adjustment tools?*
- ~ *How to make 3D Mockup book?*
- ~ *Kinemaster Latest update explained 720p*
- ~ *Editing Transitional type video 1080 p*
- ~ *Tutorial of Biography Channel*
- ~ *Kinemaster Latest update 5.0.0*
- ~ *Learning to animated videos like GIGL and Seeken*
- ~ *Learning to make pdf and slideshow in mobile*
- ~ *Recording and editing educational videos*
- ~ *Learning to import and export Kinemaster Video*
- ~ *Learning to edit Chroma Video in mobile phone*
- ~ *Audio Settings in Kinemaster*
- ~ *Learn to change video speed in Kinemaster*
- ~ *How to do Voiceover in Kinemaster?*
- ~ *Learn to make scrolling text*
- ~ *How to use Keyframe Animation Tool and Handwriting Tool in Kinemaster?*
- ~ *How to use Sticker Tool in Kinemaster?*

=> Editing Through Laptop Filmora Tutorial :

- ~ *Overview*
- ~ *Begin your Video Editing Journey with Filmora*
- ~ *How to download and install Filmora X?*
- ~ *Selection of Aspect Ratio*
- ~ *Overview of Filmora X Interface*
- ~ *Add and Adjust Background Music*
- ~ *Speed up and Slow down Video*
- ~ *Adding multiple video at single screen*

=> Functions of Filmora X :

- ~ *How to add text and filters?*
- ~ *How add filters?*
- ~ *Adding Transitions between videos*
- ~ *Adding own logo*
- ~ *How to crop video?*
- ~ *How to use pan and zoom tool?*
- ~ *Removing green screen*
- ~ *How to use Keyframing?*

=> Filmora Advance Lust pack and colour grading :

- ~ *How to use colour matching?*
- ~ *Learn to colour grade Video*

=> Filmora Advance Tutorials :

- ~ *How to add scrolling text?*
- ~ *Learn to blur face in video*
- ~ *Learning to screen record in Filmora X*

=> FilmoraGo App Tutorials :

- ~ *FilmoraGo App Tutorials*

=> Whiteboard Animation Video Tutorial :

- ~ *How to make Animated Video on Android?*
- ~ *Make Animation Video through Mobile Phone*
- ~ *Learn to script for Animation video*
- ~ *Learn to script for Animation video*
- ~ *How to make Animation Video Full Tutorial?*
- ~ *How to make cartoon tutorial?*

=> Make Thumbnail Like Successful Youtubers :

- ~ *How To Make Professional Thumbnail?*
- ~ *How To Make Attractive & Eye Catchy Thumbnail?*
- ~ *How To make Clickbait Thumbnails?*

=> Canva Complete Tutorial :

- ~ *Canva Complete Tutorial*

=> Youtube Advance Class Session { Live Sessions } With Famous Youtuber :

- ~ *How To Complete 1000 Subscribers & 4000 Hour Watch Time In 90 Days?*
- ~ *Google Adsense Complete Process Setup & Detailed Class*
- ~ *How To Grow 10x Your Youtube Channel?*
- ~ *Always Rank On Top Your Video*
- ~ *How To Compete With Your Competitor?*

#### **=> FACEBOOK :**

- ~ *Facebook related niche*
- ~ *Monetization policy*
- ~ *Types of niche on facebook*
- ~ *How to make money through facebook?*
- ~ *How To Create & Optimize A Facebook Page?*
- ~ *What Kind of Facebook Page Do You Want To Create?*
- ~ *Top 5 SEO Tips For Your Facebook Page*
- ~ *What Makes Your Facebook Page Awesome?*
- ~ *How To Create A Facebook Group For Your Business?*

#### **=> INSTAGRAM :**

- ~ *How to create Instagram Account?*
- ~ *How to find niche for Instagram Account?*
- ~ *How to make money through Instagram?*
- ~ *How To Get 1000 Followers In 50 Days?*
- ~ *Which Niche Has More Money?*
- ~ *Personal Branding Instagram Page*
- ~ *Types of Account on Instagram*
- ~ *How to make instagram reels?*
- ~ *How to set up your Instagram profile Basic To Advance?*
- ~ *How to convert your personal profile to business creator one?*

#### **=> Photoshop Graphic Designing :**

- ~ *Introduction & Workspace*
- ~ *Photoshop All Tools In Detail*
- ~ *Advance Editing In Photoshop*
- ~ *How To Create Mockups?*
- ~ *How To Create Best Quality Thumbnails?*
- ~ *Logo Designing & Channel art*

# R Shiny

---

Topic Name : PROGRAMMING

Sub-topic Name : R

Course link : <https://ineuron.ai/course/R-Shiny>

## Course Description :-

This program is specialized in providing theoretical as well as practical understanding of the RShiny package from R which allows you to learn how to build web app using Shiny package. Course curriculum includes concepts about shiny package features, handson, sharing apps and much more!

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => RShiny features
- => Architecture
- => Installation
- => User interface
- => R scripts
- => Tutorial
- => Sharing app

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Course Introduction :

- ~ Syllabus overview Preview
- ~ Installation Preview

=> Handson :

- ~ Hello World
- ~ Shiny architecture
- ~ user interface
- ~ Widgets
- ~ server/display output
- ~ R Scripts
- ~ reactive expression
- ~ Sharing app

# Java Community Class

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Java-Community-Class>

## Course Description :-

This Java Community session is intended to assist tech enthusiasts in learning the most popular programming language in the technology sector. This course covers the majority of Java Language features in great detail. These classes don't require any prerequisites.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Variables and Operators
- => Conditional statements and Loops
- => Arrays
- => Class and Objects
- => Inheritance
- => Abstraction
- => Package and Interface
- => Exception Handling
- => Multithreading
- => Collection API
- => Lambda Expression
- => TreeSet
- => HashSet
- => LinkedHashSet
- => Collection Hierarchy

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hyder Abbas :

~ Corporate Software Development Trainer with a demonstrated track record of success in the IT and Ed-tech industries. I started my career as a software developer and have since taught Java, Python, Javascript to hundreds of IT enthusiasts, including corporate professionals, throughout the years. I have been developing software for over 6 years.

## Curriculum details :-

=> Day - 1 :

- ~ Basic Understanding of Computer
- ~ Basic feature of Java

=> Day - 2 :

- ~ Main method
- ~ Classes and Objects(Basics)
- ~ Statically typed vs Dynamically types PL
- ~ Variables and data types
- ~ Naming Convention

=> Day - 3 :

- ~ Operators in Java
- ~ Incrementation and Decrementation

~ Conditional statement

=> Day - 4 :

~ Loops intro  
~ for - while - do while  
~ Nested loops

=> Day - 5 :

~ Object creation  
~ instance variable vs local variables  
~ Methods

=> Day - 6 :

~ Why array?  
~ What is an Array?  
~ How to create an array  
~ Buffer over run and ArrayIndexOutOfBoundsException  
~ Disadvantages of Array in Java

=> Day - 7 :

~ String Introduction  
~ Types of string  
~ Immutable string  
~ Ways to compare and memory map String constant pool  
~ Inbuilt methods in String class  
~ Concatination  
~ Mutable String  
~ String Buffer vs String Builder  
~ Inbuilt Methods

=> Day - 8 :

~ Need of Encapsulation  
~ What is Encapsulation?  
~ Private members  
~ Shadowing problem and this keyword  
~ Setters , Getters  
~ Constructor,  
~ this()

=> Day - 9 :

~ Inheritance intro  
~ extends keyword  
~ Types of Inheritance  
~ Types of methods Inherited, overridden, specialized  
~ Abstract keyword and Abstraction  
~ Abstract class and Abstract method  
~ Final keyword in Java

=> Day - 10 :

~ What is interface  
~ Need of Interface  
~ Different use cases of Interface  
~ Abstract vs interface  
~ Functional Interface  
~ What is Lambda Expression  
~ Different ways to create Lambda Expression

=> Day - 11 :

~ What is an Exception?  
~ try catch  
~ Multiple catch block  
~ Handling vs Ducking an Exception  
~ Hierachy of an Exception class

=> Day - 12 :

~ Why Collection ?  
~ ArrayList  
~ LinkedList  
~ PriorityQueue  
~ ArrayDeque  
~ TreeSet  
~ HashSet  
~ LinkedHashSet  
~ Collection Hierarchy

# Machine Learning Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING PROJECT

Course link : <https://ineuron.ai/course/Machine-Learning-Projects>

## Course Description :-

The science of getting computers to act without being explicitly programmed is known as machine learning. This course will teach you about the most effective machine learning techniques and give you practice putting them into practice. More significantly, you'll master not only the theoretical foundations of learning but also the practical know-how required to apply these strategies to new challenges quickly and effectively.

## Course Features :-

- => End to end model deployment in azure, GCP, AWS, and pivotal cloud
- => Time series end to end implementation in machine learning
- => Completion certificate
- => A live project with real-time implementation

## What you will learn :-

- => Machine learning end-to-end project
- => Problem solving skills and approach

## Requirements :-

- => Basic understanding of Machine Learning
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Machine Learning Projects :
  - ~ Revenue forecasting for startups
  - ~ Anomaly detection in inventory packaged material.
  - ~ Demand forecasting for FMCG products.
  - ~ Defect detection in the vehicle engine.
  - ~ Fault detection in wafers based on sensor data.
  - ~ Credit card fraud.
  - ~ Fraud detection. Preview
  - ~ Mushroom classifier, phishing classifier, thyroid detection. Preview

# MS SQL Server and T-SQL

---

Topic Name : DATABASE

Sub-topic Name : MSSQL

Course link : <https://ineuron.ai/course/MS-SQL-Server-and-T-SQL>

## Course Description :-

In this intro to SQL Server course, you gain the introductory skills to maximize SQL Server's benefits. Learn how to use the various administrative and development tools within SQL Server including the Microsoft BI stack of Integration Services, Analysis Services, and Reporting Services. SQL Server offers a platform for enterprise data management, robust development, and implementation of modern Business Intelligence solutions. Attend this hands-on course and learn to leverage these features.

## Course Features :-

- => Practical Implementation
- => Downloadable resources
- => Class Recordings
- => Quiz Questions
- => Completion Certificate

## What you will learn :-

- => Working with SQL Server
- => Architecture
- => Datatypes
- => Keys in Database

## Requirements :-

- => Prior Knowledge of Databases
- => Interest to learn
- => Your dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => MS SQL Server & T-SQL Fundamentals :
  - ~ Advantages of stored procedure
  - ~ Creating and executing stored procedures with output parameters part 1
  - ~ Creating and executing stored procedures with output parameters part 2
  - ~ Stored Procedure in SQL server part 3
  - ~ Stored Procedure in SQL server part 2
  - ~ Stored Procedure in SQL server part 1
  - ~ What is SQL Preview
  - ~ What is SQL Server
  - ~ What is DBMS Preview
  - ~ Types of DBMS
  - ~ TCL Commands
  - ~ sql server installation
  - ~ SQL Server Data Types
  - ~ SQL Server Command Categories
  - ~ SQL Server Architecture
  - ~ Operators
  - ~ Nested Queries
  - ~ Keys in Database
  - ~ Joins
  - ~ Features of SQL Server
  - ~ Exception Handling
  - ~ DML Commands
  - ~ Difference between sql and T sql
  - ~ DDL Commands Preview
  - ~ DCL Commands
  - ~ Constraints in Database



# Fivetran

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Fivetran>

## Course Description :-

In this course you will learn about Fivetran platform which allows for efficient collection of business processes and customer data from related applications, websites, and servers. The data collected is then transferred to other tools for analytics, marketing, and data warehousing purposes.

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Account management
- => Fivetran integrations
- => Salesforce connector

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Fivetran :
  - ~ Introduction to Fivetran Preview
  - ~ How fivetran works Preview
  - ~ Fivetran integrations
  - ~ Getting started
  - ~ Pricing plans
  - ~ Account management
  - ~ Architecture
  - ~ Setup connectors part 1
  - ~ Setup connectors part 2
  - ~ Setup connectors part 3
  - ~ Data Transformations on Fivetran // ETL vs ELT
  - ~ Salesforce connector

# DevOps Master

---

Topic Name : DEVOPS

Sub-topic Name : DEVOPS MASTERS

Course link : <https://ineuron.ai/course/DevOps-Master>

## Course Description :-

This course is specifically created to help you master many areas of software development, operations, continuous integration, continuous delivery, automated build, test, and deployment. You will learn DevOps tools such as Git, Jenkins, Docker, Ansible, Kubernetes, and more. This will teach you how to design Continuous Integration (CI) and Continuous Delivery (CD) pipelines to deliver applications to various environments such as testing, staging, and production.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable Resources

## What you will learn :-

- => Linux
- => Git
- => Docker
- => Kubernetes
- => Openshift
- => Ansible
- => Terraform

## Requirements :-

- => Prior Knowledge of Linux & Bash Scripting will be helpful
- => A system with a decent internet connection
- => Dedication

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

=> Ritesh Yadav :

~ Ritesh is truly passionate about data science, machine learning and DevOps in general, he likes what he does, and is keen to learn. Currently, He is working as a Jr. Data Scientist at Ineuron.ai. He also loves to contribute to Open Source Projects, which are mainly under CNCF Landscape. Ritesh loves to work in Cloud-Native technologies and Golang ( Go ). Apart from this, Ritesh has been actively involved in the open-source community for over a year, helping many open-source DevOps tools and CNCF Projects like Porter, Meshery, Keptn, TensorFlow, and Thanos through his contributions.

## Curriculum details :-

=> Linux :

- ~ Introduction to Linux Preview
- ~ What is Linux Preview
- ~ Important Pieces in Linux
- ~ Features of Linux
- ~ Evolution of Linux
- ~ Differences between Windows and Linux
- ~ Downloading necessary tools
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting Up passwordless SSH
- ~ What is Kernel
- ~ Types of Kernel
- ~ What is Shell
- ~ Types of Shell
- ~ Distros in Linux

- ~ Linux Boot Process
- ~ File System
- ~ Run Levels in Linux
- ~ File types of Linux
- ~ Package Mangement
- ~ Package Mangers & DPKG
- ~ Working with APT & APT GET
- ~ Apt-get Advanced Part 1
- ~ Apt-get Advanced Part 2
- ~ Linux Commands Part1
- ~ Linux Commands Part2
- ~ Linux Commands Part3
- ~ Linux Commands Part4
- ~ cat command usages
- ~ File Archival
- ~ File Compression
- ~ Files and Patterns Search
- ~ Input Output Redirection
- ~ Working with Vi Editor
- ~ Advanced Vi Editor Part 1
- ~ Advanced Vi Editor Part 2
- ~ Types of Account in Linux
- ~ User Management
- ~ Group Management
- ~ Files Access Controls
- ~ Linux File Permissions
- ~ Modifying File Ownership
- ~ Sudoers in Linux
- ~ Cronjobs
- ~ SCP
- ~ Special Permissions
- ~ System Management
- ~ System tools
- ~ Hard link and Soft link
- ~ Aliasing in Linux
- ~ Creating users in Multiple ways
- ~ Launching an Ubuntu VM and SSH Setup
- ~ Package installation in VM
- ~ Running our Calculator App
- ~ Unicorn & Nginx Setup
- ~ Creating a Unicorn Service
- ~ Attaching an Elastic IP
- ~ Attaching OpenSSL Certificates for https

=> Git :

- ~ Git Introduction Preview
- ~ What is Version Control? Preview
- ~ Types of Version Control Preview
- ~ What is Git?
- ~ Why Git?
- ~ Git Installation in Windows
- ~ Git Installation in Linux
- ~ Git Setup
- ~ Git Terminologies
- ~ Repositories in GIT
- ~ Creating Repository
- ~ Checking Repository History
- ~ Doing Commits
- ~ git diff
- ~ git restore
- ~ gitignore
- ~ Tagging
- ~ Branching
- ~ Branching Practicals
- ~ Merging
- ~ Merge Conflicts
- ~ Remote repository
- ~ Cloning Repository
- ~ Working with Remote Repository
- ~ Pushing to Remote Failed in Github
- ~ Personal Access Token Setup in Windows
- ~ Personal Access Token Setup in Linux
- ~ Pull Request
- ~ git Fetch & Pull
- ~ Fork
- ~ Rebasing
- ~ Interactive Rebasing
- ~ Git Rewrite History
- ~ Git Rewrite History continued
- ~ Cherry Picking
- ~ Modify Recent Commits
- ~ Git Revert

=> Docker :

- ~ Docker Introduction Preview
- ~ What is Docker? Preview
- ~ Why Docker? Preview
- ~ Benefits of Docker
- ~ What is Container?

- ~ Containers vs VM
- ~ Containers vs Image
- ~ Docker Editions
- ~ What Docker is not?
- ~ Important Terminologies
- ~ Docker Setup in Windows
- ~ Docker Setup in Linux
- ~ Docker Setup in Mac
- ~ Docker Basic Commands part 1
- ~ Docker Basic Commands part 2
- ~ Docker Run Part 1
- ~ Docker Run Part 2
- ~ Docker Images
- ~ Creating a new image
- ~ Environment variables
- ~ Commands & Entrypoints
- ~ Docker Compose
- ~ Voting Application Understanding
- ~ Docker Compose Versions
- ~ Docker Compose Networks
- ~ Voting Application with Docker Run
- ~ Voting Application with Docker Compose

=> Kubernetes :

- ~ What is Kubernetes ?
- ~ Introduction to Kubernetes
- ~ Kubernetes Architecture
- ~ Install minikube and kubectl
- ~ Components
- ~ ConfigMap & Secret
- ~ Volumes
- ~ Deployment & StatefulSet
- ~ Change the pod/deployment configuration
- ~ What is a Namespace?

=> Ansible :

- ~ What is Ansible?
- ~ Getting Started with Ansible
- ~ Ansible Components
- ~ Ansible Concepts
- ~ Ansible Controller Node Setup

=> Openshift :

- ~ Openshift on AWS and Openshift CMD
- ~ Openshift
- ~ OpenShift Vs Kubernetes

=> Terraform :

- ~ Course Requirements
- ~ What is Terraform and IaC
- ~ Terraform Workspaces
- ~ Terraform Variables, Input and Output
- ~ Terraform TtVars and Autovars
- ~ Terraform State and Statefiles
- ~ Terraform Providers
- ~ Terraform Commands
- ~ More on Terraform Commands
- ~ Launch EC2 using Terraform
- ~ Installing Terraform CLI
- ~ Installing AWS CLI

=> DevOps Hybrid Pipeline Project :

- ~ Introduction to DevOps Pipelines & its Tools Preview

=> GCP Pipeline project :

- ~ Problem Statement Preview
- ~ Requirement Gathering and Analysis Preview

# Complete Backend Development with Nodejs

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : NODE JS

Course link : <https://ineuron.ai/course/Complete-Backend-Development-with-Nodejs>

## Course Description :-

Javascript is being used for much more than it originally intended. All backend work may now be done through javascript. In this course , we will learn how to use current javascript to develop comprehensive backend code. To begin, we'll use VSCode to set up some tools. Then we'll learn how to build our own web server without using any third-party modules. After that, we'll learn express . We will also learn how to send web-based and JSON-based responses. We'll go into body parser, middleware, and templating in more detail later.

## Course Features :-

- => Course material
- => Course resources
- => On-demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Javascript
- => ES6
- => Web servers
- => Express Js
- => Body parsers
- => Middlewares
- => View Engines
- => Multer
- => Passport JS
- => Big Stack
- => MongoDB integration with Node JS

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction to NodeJS course :

- ~ Tools to be downloaded
- ~ Setting up VSCode and reading docs
- ~ Our very first project - web server
- ~ How to get exercise files

=> A web server - Manual work and reading docs :

- ~ Reading Docs for next project
- ~ Configuration for html js and css files
- ~ Finding the file on the server
- ~ Handling server error response
- ~ Finishing up node server project

=> Learn Express :

- ~ What is ExpressJs and templating
- ~ Installing express and detail about package file

- ~ Creating routes using express
- ~ Get Post and Delete requests - Postman
- ~ Routing in express
- ~ Server response and status code

=> Body parser, middleware and view engines :

- ~ What is a middleware
- ~ Applying bodyparser with express
- ~ Serving static files and form data
- ~ Using template engine

=> Multer - Upload a user profile photo :

- ~ Overview of multer and documentation
- ~ Setting up multer
- ~ configuring multer for uploads
- ~ Change profile pic using multer

=> PassportJS - Facebook Authentication :

- ~ Authentication in nodeJs
- ~ Setting up facebook app
- ~ Installing dependencies
- ~ Create all views
- ~ Configuring middleware
- ~ Configuring our routes
- ~ http and https problem in facebook auth
- ~ A demo on Heroku - Not a heroku tutorial

=> Moving to Big Stack Project :

- ~ Why we are using MongoDB
- ~ Setting up an Amazon instance using Mlab
- ~ Take time and read these npm docs
- ~ Design scalable folder structure
- ~ Creating home route and setup

=> Move to MongoDB :

- ~ Connect your project with mongoDB
- ~ Creating auth API and a challenge
- ~ Solution of challenge
- ~ Creating a person schema
- ~ Our first query in MongoDB
- ~ Creating new object from Mongo model
- ~ Generating salt and hash to save password
- ~ Using postman for testing

=> Bigstack Major Project - login routes and tokens :

- ~ Setting up login route
- ~ Validation of password in login route
- ~ Creating a Strategy using Passport
- ~ Creating tokens with information
- ~ Fixing errors and profile route

=> Bigstack Major Project Working on User Profile :

- ~ Creating model for UserProfile
- ~ Creating route for profile
- ~ Collecting user profile values
- ~ Update the profile values and save them
- ~ Debugging routes part 1
- ~ Debugging application - part 2

=> Bigstack Major Project Unique Username and other routes :

- ~ Unique username and url based access
- ~ Getting all users from database
- ~ Deleting a user from database
- ~ Workrole - Pushing array in database
- ~ Testing array based routes
- ~ Writing and testing delete route in array

=> Bigstack Major Project - Questions and Upvotes :

- ~ Question Model - Challenge
- ~ Creating question model
- ~ Creating post question routes and debugging
- ~ How to take help from stackoverflow and get route
- ~ Posting answers for questions
- ~ Upvotes routes and some assignments

# MSBI

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/MSBI>

## Course Description :-

The MSBI course is meant to give you an overview of the various tools in the Microsoft Business Intelligence Suite, such as SQL Server Integration Services, SQL Server Analysis Services, and SQL Server Reporting Services.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to MSBI Fundamentals
- => Downloading sql server
- => Installing visual studio
- => Data Flow,Control Flow, ETL demo
- => Parameters and Debugging
- => Packaging, Deployment and Running SSIS package
- => Fact ,Dimensions and star schema
- => Database Daigram
- => Shared Connection Manager and Package tasks
- => SCD, Type 0 and Type 1 Attribute
- => LookUp and Updating SSIS Package
- => Sort,Merge and Merge Joins

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => MSBI :
  - ~ Introduction to MSBI Fundamentals
  - ~ Downloading sql server
  - ~ Installing visual studio
  - ~ Data Flow,Control Flow, ETL demo
  - ~ Conditional split, data conversion and error handling
  - ~ For Loop, Variables
  - ~ Parameters and Debugging part 1
  - ~ Parameters and Debugging part 2
  - ~ Parameters and Debugging part 3
  - ~ Packaging, Deployment and Running SSIS package part 1
  - ~ Packaging, Deployment and Running SSIS package part 2
  - ~ Packaging, Deployment and Running SSIS package part 3
  - ~ Packaging, Deployment and Running SSIS package part 4
  - ~ Packaging, Deployment and Running SSIS package part 5
  - ~ Fact ,Dimensions and star schema Part 1
  - ~ Fact ,Dimensions and star schema Part 2
  - ~ Fact ,Dimensions and star schema Part 3
  - ~ Database Daigram
  - ~ Shared Connection Manager and Package tasks part 1
  - ~ Shared Connection Manager and Package tasks part 2
  - ~ SCD, Type 0 and Type 1 Attribute part 1

- ~ *SCD, Type 0 and Type 1 Attribute part 2*
- ~ *SCD, Type 0 and Type 1 Attribute part 3*
- ~ *LookUp and Updating SSIS Package*
- ~ *Sort, Merge and Merge Joins*
- ~ *Creating SSAS CUBE (SSAS) part 1*
- ~ *Creating SSAS CUBE (SSAS) part 2*
- ~ *SSAS Time series and Display result in Excel part 1*
- ~ *SSAS Time series and Display result in Excel part 2*
- ~ *Transactions and Checkpoints*
- ~ *SSRS report and implementing Matrix, Tabular Report*



# Yolo X and Yolo R

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/Yolo-X-and-Yolo-R>

## Course Description :-

This course will help you to learn the practical implementations of YoloX & YoloR.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => YoloX
- => YoloR
- => Installation \_ Setup
- => Get your custom data \_ format it
- => Doing Annotation data
- => Handle your custom labels
- => Get pretrained weights

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Boktiar Ahmed Bappy :

~ This is Bappy. I aim for simplicity in Data Science. Real Creativity won't make things more complex. Instead, I will simplify them, Interested in a Data Science Career and so develop myself accordingly. Data Scientist and lecturer with working experience in Machine Learning, Deep Learning, Microcontrollers and Electronics systems. Hands-on experience in classification, regression, clustering, computer vision, natural language processing and transfer learning models to solve challenging business problems. I have a huge interest in Robotics. I have innovated a lot of innovations, ideas, projects & robots and got a lot of achievements.

## Curriculum details :-

=> YoloX :

- ~ Introduction to YOLOX
- ~ Installation \_ Setup
- ~ Get your custom data \_ format it
- ~ Doing Annotation data
- ~ Handle your custom labels
- ~ Get pretrained weights
- ~ Training YOLOX
- ~ Evaluation \_ Visualize
- ~ Export Model \_ Tasks

=> YoloR :

- ~ Introduction to YOLOR
- ~ Installation \_ Setup
- ~ Custom Data
- ~ Data Annotation
- ~ Getting pretrained model
- ~ Model Training
- ~ Evaluation \_ Saving models

# Bash Scripting

---

Topic Name : PROGRAMMING

Sub-topic Name : SHELL SCRIPTING

Course link : <https://ineuron.ai/course/Bash-Scripting>

## Course Description :-

This course teaches you how to automate processes on UNIX systems using shell scripts. On a UNIX system, bash shell scripts allow you to automate almost any task. They combine all of the UNIX userland utilities with a robust scripting language. Shell scripts will increase your productivity whether you're a system administrator, a developer, or a power user.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Unique Characters
- => Characters Hash SemiColon
- => Variable and Parameters
- => Internal Commands
- => Regular Expressions

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Ermin Kreponic :

~ Strongly motivated young IT expert, Linux enthusiast with a passion for troubleshooting network related problems. With an exceptional eye for details and a sense of urgency when it comes down to problem solving.

## Curriculum details :-

=> Introduction :

~ Intro

=> Setting up the environment :

~ Set up part 1

~ Set up part 2

=> Unique Characters :

~ Terminal customization and hash character

~ Characters Hash SemiColon

~ Characters Dot Double quotes Single quotes Comma Double comma

~ Backslash Slash Backquote Null command

~ [exclamation mark] [asterix] [question mark]

~ Parenthesis and Curly braces

~ Characters [OR] [AND] [Dash] [Modulo] [Tilde Plus] [Tilde Minus] [Caret] [Double Caret]

=> Variable and Parameters :

~ Variables and Parameters part 1

~ Variables and Parameters part 2

~ Variables and Parameters part 3

~ Variables and Parameters part 4

=> Return Values :

~ Return Values part 1

~ Return Values part 2

=> Conditional Statements :

~ Conditional Statements part 1

~ Conditional Statements part 2

=> Variables Continued :

~ Built in variables part 1

~ Built in variables part 2

- ~ *Built in variables part 3*
- ~ *Modifying the properties of variables*
- ~ *Random number generation*

=> Loops :

- ~ *For Loop*
- ~ *Until Loop*
- ~ *While Loop*
- ~ *Break and Continue Part 1*
- ~ *Break and Continue Part 2*
- ~ *Case Construct*
- ~ *Select Construct*

=> Internal Commands :

- ~ *Printf*
- ~ *Read*
- ~ *Eval and Set*
- ~ *More on set + unset*
- ~ *getopts*
- ~ *shopt type jobs disown*
- ~ *fg kill and command*

=> Regular Expressions :

- ~ *Grep Demo*
- ~ *Demo for sed Extended regular expressions*
- ~ *Globing*

=> Input Output Redirection :

- ~ *STDOUT, STDIN, STDERR part 1*
- ~ *STDOUT, STDIN, STDERR part 2*

=> Functions :

- ~ *Functions part 1*
- ~ *Functions part 2*

=> Arrays :

- ~ *Arrays part 1*
- ~ *Arrays part 2*
- ~ *Arrays part 3*

=> Lists :

- ~ *OR and AND lists*

=> Debugging :

- ~ *Debugging part 1*
- ~ *Debugging part 2*
- ~ *Debugging part 3*

# Data Processing using Dask

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Data-Processing-using-Dask>

## Course Description :-

In this course, students will learn how to scale data analysis tasks using a very powerful Python library called Dask. You will use Dask to run distributed data analysis tasks, from processing the data to visualizing it. After successfully completing this course, you will be able to put data analysis ideas into practice using a variety of approaches on datasets.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction about Dask library
- => Distributed computing
- => Applications of Dask
- => Introduction to Google colab
- => Manipulation of Dask Data Frames
- => Statistics overview
- => Overview about Descriptive statistics
- => Overview of Data visualization
- => Importance of data visualization
- => Different types of Plots Using Seaborn

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Course Introduction :
  - ~ Dashboard overview
  - ~ Course overview
  - ~ Who is this course for?
  - ~ What is a Data Science?
  - ~ Why should you learn Data Science?
  - ~ History of Data Science
  - ~ What is distributed computing?
  - ~ What is Dask?
  - ~ Why should you learn Dask?
  - ~ Applications of Dask
- => Introduction to Jupyter Notebook and Python programming :
  - ~ An overview of Python programming language
  - ~ Understanding the Google colab structure
  - ~ Understanding jupyter notebook structure
  - ~ Saving and loading jupyter notebooks
  - ~ Why should you learn Python programming?
  - ~ Applications of Python programming
  - ~ Installing Dask library in colab
- => Introduction to Dask :
  - ~ Introduction to distributed computing
  - ~ Why do we use distributed computing?

- ~ What are DataFrames?
- ~ Why should we use DataFrames?
- ~ Dask vs Pandas

=> Dask fundamentals :

- ~ Reading data from text files using dask
- ~ Overview of csv files
- ~ Reading data from csv files
- ~ Selecting columns from dask dataframe
- ~ Dropping columns from dask dataframe
- ~ Renaming columns in dask dataframe
- ~ Selecting rows from a dataframe
- ~ Counting missing values from a dataframe
- ~ Dropping rows with missing data

=> Assignment 1 :

- ~ Reading and analysing a geographical dataset

=> Analyzing dataframes :

- ~ What are descriptive statistics?
- ~ Overview of descriptive statistics
- ~ Calculating descriptive statistics with dask
- ~ Using describe method for descriptive statistics
- ~ What is correlation in data?
- ~ Calculating correlations in dask dataframes
- ~ What is numerical data?
- ~ What is categorical data?

=> Assignment 2 :

- ~ Program to find descriptive statistics of a pokemon dataset

=> Data Visualization with Dask :

- ~ What is data visualization?
- ~ Importance of data visualization
- ~ Introduction to Seaborn
- ~ Installing Seaborn
- ~ Creating a scatterplot
- ~ Creating bar graph
- ~ Creating pie charts
- ~ Creating line plots

=> Assignment 3 :

- ~ Building interactive charts and dashboard on pokemon dataset

=> Course summary :

- ~ Course Outro
- ~ Future scope and references

# Dask

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Dask>

## Course Description :-

Dask is a flexible library for parallel computing in Python. It can easily handle large data which enables users to perform ml related tasks at scale.

## Course Features :-

- => Self-Paced Classes
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Dask Arrays
- => Dask Dataframes
- => Dask Bags
- => ML with Dask

## Requirements :-

- => Little bit of Python Knowledge
- => Dedication
- => Internet Connection

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Introduction :
  - ~ The course Overview Preview
  - ~ Introduction to Dask Preview
  - ~ Dask Alternatives
  - ~ Advantages of using dask
  - ~ Limitations of task
  - ~ Dask Setup Preview
- => Understanding dask arrays :
  - ~ Introduction to blocked algorithms
  - ~ Hands on with DASK Arrays
  - ~ Digging deeper into dask arrays
  - ~ performance comparison with numpy arrays Preview
  - ~ creating universal numpy functions with dask
  - ~ Limitations of Dask
- => Parallelizing python code with DASK :
  - ~ Lazy Evaluation
  - ~ using `dask.delayed`
  - ~ understand task graphs
- => Understanding Dask Dataframes :
  - ~ Introduction to dask dataframes
  - ~ exploring dask dataframes
  - ~ creating dask dataframes
  - ~ loading large datasets with dask dataframes
  - ~ analyzing data with dask dataframes
  - ~ limitations of dask dataframes
- => Exploring Dask Bags :
  - ~ Introduction to dask bags
  - ~ creating and storing dask bags
  - ~ manipulating dask bags
  - ~ word count example using dask bags
  - ~ Limitations of Dask Bags
- => Distributed computing with dask :
  - ~ overview of distributed computing with dask
  - ~ setting up your dask cluster
  - ~ understanding dask schedulers

*~ Exploring dask dashboard UI*

**=> Machine Learning with Dask :**

*~ Introduction to dask ML Preview*

*~ using dask ML for regression*

*~ using dask ML for Classification*

# DSA with Python

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH PYTHON

Course link : <https://ineuron.ai/course/DSA-with-Python>

## Course Description :-

This Python course on Data Structures and Algorithms covers data structures such as linked lists, stacks and queues, binary search trees, heaps, searching, and hashing. This course covers a variety of sorting algorithms, as well as their implementation and analysis. The following topics are covered with Python implementation in this Data Structures in Python course. Analysis of Algorithms, Big O notation, Time Complexity, Singly Linked List, Doubly linked list, Trees, Heaps, Hashing and Sorting algorithms.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Big O notation
- => Time and space complexity
- => DSA problem solving
- => Stacks and heaps
- => Physical and logical structures
- => Abstract data types
- => Recursion
- => Linked Lists
- => Stacks
- => Queues
- => Trees
- => Hashing
- => AVL trees
- => Heaps
- => Sorting

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to DSA :
  - ~ Why we need Data structures and algorithms Preview
  - ~ Time based approach
  - ~ Concept of Big O and graphs Preview
  - ~ Data Structures and Algorithms HB
- => Problem Solving :
  - ~ Start with a challenge - reverse string
  - ~ Reverse a string - solution
  - ~ Interview approach to solve a problem Preview
  - ~ Classic interview steps for DSA problems



## => Data Structure Introduction :

- ~ Memory process - Stack and Heap Preview
- ~ Physical and logical data structures
- ~ Abstract Data Types - ADT

## => Recursion in depth :

- ~ Introduction to recursion
- ~ Tracing the recursion tree
- ~ Trace tree assignment
- ~ Trace tree solution
- ~ Types of Recursion Preview
- ~ Complex recursion tree
- ~ What is Factorial
- ~ Factorial program in Python
- ~ Fibonacci series THEORY
- ~ Fibonacci series and its version Python Code
- ~ What is Power Program
- ~ Power Program Python code
- ~ What is a Combination Program
- ~ Combination Program Python code
- ~ Classic Tower of Hanoi problem
- ~ Classic Tower of Hanoi Python code

## => Linked List in depth :

- ~ Introduction to Linked List
- ~ Add value in linked list - cases
- ~ Push Append and insert in LinkedList - Python code
- ~ Deletion of linked list THEORY.
- ~ Deletion in linked list Python code
- ~ Delete complete linked list Python code
- ~ Count all nodes in linked list python code
- ~ Reversing a linked list THEORY
- ~ Reversing a linked list Python code

## => Circular Linked List in Depth :

- ~ Circular linked list THEORY
- ~ Circular Linked List push Python code
- ~ Traverse a circular linked list Python code
- ~ Deletion in circular linked list Python code
- ~ count nodes in circular linked list Python code
- ~ convert linked list to circular linked list Python code

## => Doubly Linked List in Depth :

- ~ Theory for doubly linked list
- ~ Doubly linked list push Python code
- ~ Insert After in doubly linked list Python code
- ~ add to last in doubly linked list Python code
- ~ Traverse a doubly linked list Python code
- ~ Deleting a node in doubly linked list Python code

## => Stack and Queue :

- ~ Stack - Push and Pop operation THEORY
- ~ Stack operations with Python code
- ~ Queue concept THEORY
- ~ Queue implementation in Python code
- ~ Circular queue THEORY
- ~ Circular queue Python code

## => Binary Search Tree :

- ~ What is Binary Search tree and creation THEORY update
- ~ Insertion and Deletion in BST THEORY
- ~ InOrder Traversal of BST THEORY
- ~ Pre Order traversal in BST THEORY
- ~ Post order traversal in BST THEORY
- ~ Creating a Binary Search tree Python code
- ~ search a key in BST Python code
- ~ Insertion in BST Python code
- ~ deletion of key in BST Python code
- ~ inorder preorder and postorder traversal in BSTPython code

## => Hashing :

- ~ What is Hashing THEORY
- ~ Hash chaining with linked list
- ~ Linear Hash Shifting
- ~ Square hash shifting

## => AVL Tree :

- ~ What is AVL tree and height Preview
- ~ Finding balance factor
- ~ Left Left and Right Right Rotation in AVL Tree
- ~ LR and RL rotation with 1 trick
- ~ Creating a AVL tree - Important
- ~ Deletion in AVL Tree.

## => HEAP :

- ~ Heap - Max and min Heap
- ~ Insertion and deletion in HEAP

## => Sorting algorithms :

- ~ Categories of sorts
- ~ Selection sort - Theory

- ~ *Selection sort - Python Code Preview*
- ~ *Bubble Sort - Theory*
- ~ *Bubble Sort - Python Code*
- ~ *Insertion sort - Theory*
- ~ *Insertion sort - Python Code*
- ~ *Quick Sort - Theory*
- ~ *Quick Sort - Theory part 2*
- ~ *Quick Sort - Python Code*
- ~ *Counting Sort - Theory*
- ~ *Merge Sort Theory*
- ~ *Merge sort Python code*
- ~ *Counting Sort - Python Code*

# Class 9th Math

---

Topic Name : K12

Sub-topic Name : CLASS9

Course link : <https://ineuron.ai/course/Class-9th-Math>

## Course Description :-

This course is useful for Grade 9 students. In this course, entire NCERT will be covered, Various questions from NCERT, NCERT exemplar and previous year will also be discussed. Dedicated doubt clearing sessions will also be conducted for helping the students regularly throughout their learning journey.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Algebra
- => Geometry
- => Statistics
- => Probability
- => Number Systems

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

### => NUMBER SYSTEMS :

- ~ Lecture 1 : Introduction Preview
- ~ Lecture 2 : Inserting Rational Numbers Between Two Rational Numbers Preview
- ~ Lecture 3 : NCERT solutions Ex1.1
- ~ Lecture 4 : Representation of Irrational Numbers
- ~ Lecture 5 : NCERT solutions Ex1.2
- ~ Lecture 6 : NCERT solutions
- ~ Lecture 7 : NCERT solutions Ex1.4
- ~ Lecture 8 : Theory
- ~ Lecture 9 : Representation of Root X Proof
- ~ Lecture 10 : NCERT solutions Ex1.5 Question 1&2
- ~ Lecture 11 : NCERT solutions Ex1.5 Question 2,3&4
- ~ Lecture 12 : NCERT solutions Ex1.6

### => POLYNOMIALS :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Types of Polynomials, Polynomials in One Variable
- ~ Lecture 3 : Problem Discussion
- ~ Lecture 4 : Value of a Polynomial, Zeroes / Roots of a Polynomial
- ~ Lecture 5 : Remainder Theorem, Factors and Multiples
- ~ Lecture 6 : Factor Theorem
- ~ Lecture 7 : Questions on Factor Theorem
- ~ Lecture 8 : Algebraic Identities,

### => COORDINATE GEOMETRY :

- ~ Lecture 1 : Understanding Cartesian Coordinates
- ~ Lecture 2 : Plotting Points on X Y Plane ( Cartesian Coordinate System)

### => LINEAR EQUATIONS IN TWO VARIABLES :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : General form of Linear Equation in Two Variables
- ~ Lecture 3 : Solution of a Linear Equation in Two Variable
- ~ Lecture 4 : Graph of Linear Equation in Two Variable
- ~ Lecture 5 : Some Other Important Graph

### => INTRODUCTION TO EUCLID'S GEOMETRY :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Euclid's Definition, Axioms and Postulates
- ~ Lecture 3 : Euclid's Five Postulates
- ~ Lecture 4 : Exercise 5.1

~ Lecture 5 : Equivalent Version of Euclid's 5th Postulates

## => LINES AND ANGLES :

~ Lecture1\_Chapter\_Contents  
~ Lecture2\_Introduction  
~ Lecture3\_Basic\_Terms\_&\_Definitions  
~ Lecture4\_NCERT\_Example\_Problems\_Discussion  
~ Lecture5\_NCERT\_EX6.1\_Problem\_Discussion  
~ Lecture6\_Parallel\_Lines\_&\_Transversal  
~ Lecture7\_NCERT\_Parallel\_Line\_Example\_Problem\_Discussion  
~ Lecture8\_NCERT\_EX6.2\_Problems\_Discussion  
~ Lecture9\_Triangle\_Angle\_Sum\_Property  
~ Lecture10\_NCERT\_EX6.3\_Problem\_Discussions  
~ Lecture11\_NCERT\_Exemplar\_Problem\_Discussion  
~ Lecture12\_HOTS\_Question\_Discussion

## => TRIANGLES :

~ Lecture 1 : Introduction Congruence  
~ Lecture 2 : Part1 SAS Rule  
~ Lecture 3 : NCERT solutions Ex7.1 Question 1&2  
~ Lecture 4 : NCERT solutions Ex7.1 Question 3&4  
~ Lecture 5 : NCERT solutions Ex7.1 Question 5,6&7  
~ NaN  
~ Lecture 7 : Theorem 7.2&7.3  
~ Lecture 8 : NCERT solutions Ex7.2 Question 1&2  
~ Lecture 9 : NCERT solutions Ex7.2 Question2&3  
~ Lecture 10 : NCERT solutions Ex7.2 Question 5,6,7,8  
~ Lecture 11 : SSS & RHS Introduction  
~ Lecture 12 : NCERT solutions Ex7.3 Question 1&2  
~ Lecture 13 : NCERT solutions Ex7.3 Question 3,4&5  
~ lec 14 : IntroductionTheorem 7.6,7.7, 7.8 & Ex7.4 Question 1  
~ Lecture 15 : NCERT solutions Ex7.4 Question1,2  
~ Lecture 16 : NCERT solutions Ex7.4 Question 3,4,5,6

## => QUADRILATERALS :

~ Lecture 1 : Understanding Quadrilaterals and related important terms , Different types of Quadrilaterals  
~ Lecture 2 : Theorem 8.1 : A diagonal of a parallelogram divides it into two congruent triangles. Theorem 8.3 : If each pair of opposite sides of a quadrilateral is equal, then it is a parallelogram.  
~ Lecture 3 : Theorem 8.5 : If in a quadrilateral, each pair of opposite angles is equal, then it is a parallelogram. Theorem 8.6 : The diagonals of a parallelogram bisect each other. Theorem 8.7 : If the diagonals of a quadrilateral bisect each other, then it is a parallelogram. Theorem 8.8 : A quadrilateral is a parallelogram if a pair of opposite sides is equal and parallel.  
~ Lecture 4 : Ex 8.1 Q 1,2,3,4  
~ Lecture 5 : Ex 8.1 Q 5,6,7  
~ Lecture 6 : Ex 8.1 Q 8,9,10,11,12  
~ Lecture 7 : Mid Point Theorem  
~ Lecture 8 : Question Discussion

## => AREAS OF PARALLELOGRAMS AND TRIANGLES :

~ Lecture 1 : Figures on the Same Base and Between the Same Parallels, Theorem 1 : Parallelograms on the same base and between the same parallels are equal in area.  
~ Lecture 2 : EX 9.2 / Q 1,2,3,4  
~ Lecture 3 : EX 9.2 / Q 5,6  
~ Lecture 4 : Theorem 2 : Two triangles having the same base (or equal bases) and equal areas lie between the same parallels.  
~ Lecture 5 : EX 9.3 / Q 1,2,3,4  
~ Lecture 6 : EX 9.3 / Q 5, 6, 7  
~ Lecture 7 : EX 9.3 / Q 8, 9, 10, 11  
~ Lecture 8 : EX 9.3 / Q 12, 13, 14  
~ Lecture 9 : EX 9.3 / Q 15, 16

## => CIRCLES :

~ Lecture1\_Course\_Contents\_&\_Introduction  
~ Lecture2\_Circle\_Terminologies  
~ Lecture3\_Angle\_Subtended\_By\_Chords\_At\_Centre\_Of\_Circle  
~ Lecture4\_Perpendicular\_From\_Centre\_To\_Chord  
~ Lecture5\_Circle\_Through\_Three\_Points  
~ Lecture6\_Equal\_Chords\_And\_Their\_Distances\_From\_The\_Centre\_Equal\_Chords\_And\_Distance\_From\_Centre  
~ Lecture7\_NCERT\_EX10.4\_PROBLEMS\_DISCUSSIONS  
~ Lecture8\_ANGLE\_SUBTENDEDCORD\_CENTRE\_CIRCLE  
~ Lecture9\_CYCLIC\_QUADRILATERAL  
~ LECTURE10\_NCERT\_EX10.5\_PROBLEMS\_DISCUSSION  
~ LECTURE11\_NCERT\_EXEMPLAR\_PROBLEMS\_DISCUSSIONS  
~ LECTURE12\_CIRCLE\_HOTS\_QUESTIONS

## => CONSTRUCTIONS :

~ Lecture 1 : Angle bisector & perpendicular bisector of a line segment  
~ Lecture 2 : 60 degree angle & triangle construction when sum of two sides base and one base angle is given  
~ Lecture 3 : Difference of two sides one base angle and base is given  
~ Lecture 4 : perimeter and two base anngles are given

## => HERONS FORMULA :

~ Lecture 1 : Area of a Triangle using Heron's Formula  
~ Lecture 2 : EXERCISE 12.1  
~ Lecture 3 : EXERCISE 12.1  
~ Lecture 4 : EXERCISE 12.2

## => SURFACE AREAS AND VOLUMES :

~ Lecture1\_Course\_Contents\_&\_Introduction  
~ Lecture2\_Surface\_Area\_Cuboid\_&\_Cube

- ~ Lecture3\_NCERT\_EX13.1\_Problem\_Discussion
- ~ Lecture4\_Right\_Circular\_Cylinder
- ~ Lecture5\_NCERT\_EX13.2\_PROBLEM\_DISCUSSIONS
- ~ Lecture6\_RIGHT\_CIRCULAR\_CONE
- ~ Lecture7\_NCERT\_EX13.3\_PROBLEMS\_DISCUSSIONS
- ~ Lecture8\_SURFACE\_AREA\_SPHERE\_&\_HEMISPHERE
- ~ Lecture9\_NCERT\_EX13.4\_PROBLEMS\_DISCUSSION
- ~ Lecture10\_VOLUME\_CUBOID\_&\_CUBE
- ~ Lecture11\_NCERT\_EX\_13.5\_PROBLEMS\_DISCUSSION
- ~ Lecture12\_VOLUME\_OF\_CYLINDER
- ~ Lecture13\_NCERT\_EX13.6\_PROBLEM\_DISCUSSIONS
- ~ LECTURE14\_VOLUME\_OF\_RIGHT\_CIRCULAR\_CONE
- ~ LECTURE15\_NCERT\_EX13.7\_PROBLEM\_DISCUSSION
- ~ LECTURE16\_VOLUME\_OF\_SPHERE\_&\_HEMISPHERE
- ~ LECTURE17\_NCERT\_EX13.8\_PROBLEM\_DISCUSSIONS
- ~ LECTURE18\_NCERT\_OPTIONAL\_PROBLEM\_DISCUSSION
- ~ LECTURE19\_HOTS\_QUESTION\_DISCUSSION

=> STATISTICS :

- ~ Lecture1\_Course\_Content\_Discussion
- ~ Lecture2\_Introduction
- ~ Lecture3\_Data\_Collection
- ~ Lecture4\_NCERT\_EX14.1\_Discussion
- ~ Lecture5\_Data\_Representation
- ~ Lecture6\_NCERT\_EX\_14.2\_Discussion
- ~ Lecture7\_Graphical\_Represntation\_Data\_&\_BarGraph
- ~ Lecture8\_Graphical\_Representation\_Histogram
- ~ Lecture9\_Data\_Representation\_Frequency\_Polygon
- ~ Lecture10\_NCERT\_EX14.3\_Discussion
- ~ Lecture11\_Measures\_Central\_Tendency\_Mean
- ~ Lecture12\_Measures\_Central\_Tendency\_Median
- ~ Lecture13\_Measures\_Central\_Tendency\_Mode
- ~ Lecture14\_Comparisons\_Mean\_Median\_Mode
- ~ Lecture15\_NCERT\_EXEMPLAR\_PROBLEM\_MCQ
- ~ Lecture16\_NCERT\_EXEMPLAR\_PROBLEM\_VERY\_SHORT\_ANSWER\_TYPE\_QUESTIONS
- ~ Lecture17\_NCERT\_EXEMPLAR\_PROBLEM\_SHORT\_ANSWER\_TYPE\_QUESTIONS
- ~ Lecture18\_NCERT\_EXEMPLAR\_PROBLEM\_LONG\_ANSWER\_TYPE\_QUESTIONS

=> PROBABILITY :

- ~ Lecture1\_Introduction
- ~ Lecture2\_Experimental\_Approach\_Probability
- ~ Lecture3\_NCERT\_Example\_Problem\_Discussion
- ~ Lecture4\_NCERT\_Problems\_Discussion
- ~ Lecture5\_NCERT\_Exemplar\_Problem\_Discussion

# Scala

---

Topic Name : PROGRAMMING

Sub-topic Name : SCALA

Course link : <https://ineuron.ai/course/Scala>

## Course Description :-

With actual executions and examples, this course will help you learn object-oriented parts of Scala, such as trait methods and XML. By the end of the course, you'll have a solid working knowledge of Scala and be able to apply it in real-world situations.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What is Scala and Why to Learn Scala?
- => Scala Setup
- => Scala First Code
- => Scala Defining Variable using Var & Val
- => Class and Object in Scala
- => Creating Methods in Scala
- => List & Lambda Expression in Scala
- => List Reverse , Drop & Take
- => Scala Type Hierarchy
- => List of Complex Objects in Scala
- => Tuples in Scala

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Navin Reddy :

~ I am Corporate Java trainer. Since past few years successfully trained many professionals at JP Morgan, Accenture, Polaris and L&T infotech. My youtube channel "Telusko" presently has 1.7 million subscribers. Passionate about Java Technology for over a decade and moved on as a corporate trainer. I am certified blockchain developer and Currently, building Applications running on Blockchain (dapps).

## Curriculum details :-

- => Scala :
  - ~ What is Scala and Why to Learn Scala?
  - ~ Scala Setup
  - ~ Scala First Code
  - ~ Scala Defining Variable using Var & Val
  - ~ Class and Object in Scala
  - ~ Creating Methods in Scala
  - ~ List & Lambda Expression in Scala
  - ~ List Reverse , Drop & Take
  - ~ Scala Type Hierarchy
  - ~ List of Complex Objects in Scala
  - ~ Tuples in Scala

# Digital Marketing and Youtube Niche Foundations

---

Topic Name : DIGITAL MARKETING

Sub-topic Name : DIGITAL MARKETING MASTERS

Course link : <https://ineuron.ai/course/Digital-Marketing-and-Youtube-Niche-Foundations>

## Course Description :-

May I will tell you in the coming 15th class on YouTube how you can use YouTube and digital marketing, You will be able to do your youtube and Digital Marketing and you will make good career in this filed, this will be a free program of 15 days. on youtube we make creators YouTube channel

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => How to find your niche
- => Youtube related Q&A
- => Social media Q&A
- => Facebook ads
- => Think About Your Interests and Passions
- => Solve Problems For Your Audience
- => Why You Want To Select This Niche
- => Carry Out Competitor Analysis
- => Choose a Niche That Gives You The Most YouTube Views.
- => Check Out the Competition for Your Niche
- => Do Keyword Research
- => Assess your Niche
- => Find your Unique Selling Point
- => Test your Idea
- => How to start monetize your skills

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Amresh Bharti :

~ Amresh Bharati is an inspiring entrepreneurial figure, Digitalised marketing & startup growth consultant, renowned YouTube personality, Online branding expert, highly stimulating coach, former teacher, video marketing trainer, online earning & kick startup trainer, author, Josh talk speaker and among one of the most leading digital entrepreneurs of todays time. this is Mahatmaji Technical website to learn Digital Marketing

## Curriculum details :-

- => How to find your niche :
  - ~ How to find your niche
  - ~ Youtube related Q&A
  - ~ Social media Q&A
  - ~ Facebook ads
  - ~ Think About Your Interests and Passions
  - ~ Solve Problems For Your Audience
  - ~ Why You Want To Select This Niche
  - ~ Carry Out Competitor Analysis
  - ~ Choose a Niche That Gives You The Most YouTube Views.
  - ~ Check Out the Competition for Your Niche
  - ~ Do Keyword Research
  - ~ Assess your Niche

- ~ *Find your Unique Selling Point*
- ~ *Test your Idea*
- ~ *How to start monetize your skills*



# Certified Ethical Hacker Bootcamp

---

Topic Name : CYBER SECURITY

Sub-topic Name : CYBERSECURITY MASTERS

Course link : <https://ineuron.ai/course/Certified-Ethical-Hacker-Bootcamp>

## Course Description :-

Ethical hacking is a topic that has grown increasingly essential in today's world, and it can assist individuals and companies in adopting safe IT practices and usage. This ethical hacking course will teach you those skills as well as prepare you for associated certification examinations, allowing you to demonstrate your competence.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to Ethical Hacking
- => Reconnaissance - Surveying the Attack Surface
- => Network Presence
- => Attacking
- => Web Hacking

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Joseph Delgadillo :

~ *The digital age is upon us. Would you like to build/protect the systems that shape our future? I am here on Teachable to produce valuable educational resources for students who wish to learn skills related to information technology, network security, programming languages and much more. Enroll in my course for a practical, down to earth approach to learning.*

## Curriculum details :-

=> Introduction to Ethical Hacking :

- ~ *What is an ethical hacker*
- ~ *Terminology crash course pt1*
- ~ *Terminology crash course pt2*
- ~ *Terminology crash course pt3*
- ~ *CIA*
- ~ *Legal considerations*

=> Reconnaissance - Surveying the Attack Surface :

- ~ *Surveying the attack surface*
- ~ *Recon types*
- ~ *Passive recon part 1*
- ~ *Passive recon part 2*
- ~ *Active recon*
- ~ *Recon walkthrough tools summary*
- ~ *Maltego demo*
- ~ *FOCA demo*
- ~ *Harvester demo*
- ~ *Reconng demo*

=> Scanning and Enumeration - Getting Down to Business :

- ~ *Scanning enumeration*
- ~ *Identifying active hosts pt1*
- ~ *Identifying active hosts pt2*
- ~ *Identifying active services*
- ~ *OS and services fingerprinting*
- ~ *Network mapping*
- ~ *Final thoughts*
- ~ *Nmap syntax pt1*
- ~ *Nmap syntax pt2*

- ~ Nmap hosts discovery
- ~ Nmap service discovery
- ~ Nmap scripts
- ~ masscan

#### => Network Presence :

- ~ Network insecurity
- ~ Sniffing and spoofing
- ~ Sniffing tools
- ~ Spoofing 2C crypto 2C and wifi
- ~ Tcpdump
- ~ Wireshark
- ~ Ettercap
- ~ SSL burp
- ~ Scapy

#### => Attacking :

- ~ Security overview windows architecture
- ~ Security overview credentials security
- ~ Security overview memory corruption and exploitation
- ~ Windows hacking basics
- ~ Local access and privilege escalation
- ~ Dumping hashes and cracking passwords
- ~ Linux attacking basics pt1
- ~ Linux attacking basics pt2
- ~ References
- ~ Windows msf exploit pt1
- ~ Windows msf exploit pt2
- ~ Post exploitation
- ~ Mimikatz
- ~ Mimikatz john the ripper
- ~ Hashcat
- ~ Konboot
- ~ Post cmd
- ~ Post powershell
- ~ Hydra ncrack pt1
- ~ Hydra ncrack pt2
- ~ Attacking Linux targets pt1
- ~ Attacking Linux targets pt2

#### => Web Hacking :

- ~ Introduction to web hacking
- ~ Web security architecture overview pt1
- ~ Web security architecture overview pt2
- ~ Attacking the web server pt1
- ~ Attacking the webserver pt2
- ~ Attacking the platform pt1
- ~ Attacking the platform pt2
- ~ Attacking the technology pt1
- ~ Attacking the technology pt2
- ~ OWASP top 10 pt1
- ~ OWASP top 10 pt2
- ~ Attacking the business logic pt1
- ~ Attacking the business logic pt2
- ~ Tools and methodology
- ~ References
- ~ OWASP
- ~ SQLI
- ~ SQL map intro
- ~ SQL map
- ~ Burpsuite
- ~ Burpsuite xsshunter
- ~ Mitmproxy
- ~ Skipfish pt1
- ~ Skipfish pt2

#### => Social Engineering - Hacking Humans :

- ~ Social engineering basics
- ~ Social engineering methods
- ~ Tools and techniques pt1
- ~ Tools and techniques pt2
- ~ Tools and techniques pt3
- ~ Physical security considerations
- ~ Final Thoughts
- ~ Intro demo
- ~ Toolkit prep
- ~ Credential harvesting
- ~ Website cloning
- ~ Automating an attack
- ~ Antivirus evasion pt1
- ~ Antivirus evasion pt2 UPDATED

# Business Analytics Masters with 3 month Internship

---

Topic Name : DATA ANALYTICS

Sub-topic Name : BUSINESS ANALYTICS MASTERS

Course link : <https://ineuron.ai/course/Business-Analytics-Masters-with-3-month-Internship>

## Course Description :-

Learn the power of using powerful visualization tools such as PowerBi and Tableau alongside advanced excel coupled with the most important fundamentals of Python

## Course Features :-

=> Business Analytics Masters

## What you will learn :-

=> Python

=> PowerBI

=> Tableau

=> Advanced Excel

=> Statistics

## Requirements :-

=> Laptop

=> Stable internet connection

## Instructors :-

=> Amit Bose :

~

## Curriculum details :-

=> Course Curriculum

=> Introduction to Analytics

=> Python for Data Analytics :

- ~ *Install setup and overview Preview*
- ~ *Ipython /Jupyter Notebook overview.*
- ~ *Intro to NUMPY.*
- ~ *Creating Arrays.*
- ~ *Using Arrays and Scalar.*
- ~ *Indexing Arrays.*
- ~ *Arrays transposition.*
- ~ *Universal arrays function.*
- ~ *Arrays processing.*
- ~ *Array input and output.*
- ~ *Series.*
- ~ *Data frames.*
- ~ *Index Objects.*
- ~ *Re index.*
- ~ *Drop entry.*
- ~ *Selecting entries.*
- ~ *Data alignment.*
- ~ *Rank and Sort.*
- ~ *Summary statistics.*
- ~ *Missing data.*
- ~ *Index Hierarchy.*
- ~ *Reading and writing text files.*
- ~ *JSON with Python.*
- ~ *HTML with Python.*
- ~ *Microsoft Excel files with Python.*
- ~ *Merge.*
- ~ *Merge on Index.*
- ~ *Concatenate.*
- ~ *Combining Data Frames.*
- ~ *Reshaping.*
- ~ *Pivoting.*
- ~ *Duplicates in DataFrames*
- ~ *Mapping.*
- ~ *Replace.*
- ~ *Rename index.*
- ~ *Binning.*
- ~ *Outliners.*
- ~ *Permutation.*
- ~ *GroupBy on DataFrames*
- ~ *GroupBy on Dict and Series.*
- ~ *Aggregation.*

- ~ Splitting, Applying and combining.
- ~ Cross Tabulation.
- ~ Installing Seaborn.
- ~ Histograms.
- ~ Kernel Density estimate plots.
- ~ Combining plot styles.
- ~ Box and Violin plots.
- ~ Regression Plots.
- ~ Heat maps and clustered matrices.
- ~ Introduction to SQL with Python.
- ~ SQL SELECT, DISTINCT, WHERE, AND & OR.
- ~ SQL WILDCARDS, ORDER BY, GROUP BY, and Aggregate Functions.

#### => SQL FOR DATA ANALYTICS :

- ~ Introduction.
- ~ ER Diagram.
- ~ Schema Design.
- ~ Normalization.
- ~ SQL SELECT statement.
- ~ SQL SELECT using common functions.
- ~ SQL JOIN overview.
- ~ INNER JOIN.
- ~ LEFT JOIN.
- ~ RIGHT JOIN.
- ~ FULL JOIN.
- ~ SQL best practice.
- ~ INNER JOIN Advanced.
- ~ INNER JOIN and LEFT JOIN combo.
- ~ SELF JOIN.
- ~ JOINS and AGGREGATION Subqueries.
- ~ Sorting.
- ~ Independent Subqueries.
- ~ Co related Subqueries.
- ~ Analytic function.
- ~ Set operations.
- ~ SQL views.
- ~ Create a view.
- ~ Create a view using DDL.
- ~ SQL insert Advanced Technique.
- ~ Insert to create table.
- ~ INSERT to new data on existing table 1.
- ~ INSERT to new data on existing table 2.
- ~ INSERT to new data on existing table 3
- ~ INSERT to new data on existing table 4.
- ~ SQL update Advance technique and TCL.
- ~ SQL delete and TCL.
- ~ SQL constraints.
- ~ SQL aggregations.
- ~ SQL programmability.
- ~ SQL query performance.
- ~ SQL Extras.

#### => Advance Excel :

- ~ Data wrangling with Excel
- ~ Microsoft Excel fundamentals.
- ~ Entering and editing texts and formulae.
- ~ Working with basic Excel functions.
- ~ Modifying an Excel worksheet.
- ~ Formatting data in an excel worksheet.
- ~ Inserting images and shapes into an Excel worksheet.
- ~ Creating Basic charts in Excel.
- ~ Printing an Excel worksheet.
- ~ Working with an Excel template.
- ~ Working with an excel list.
- ~ Excel list function.
- ~ Excel data validation.
- ~ Importing and exporting data.
- ~ Excel pivot tables.
- ~ Working with excels PowerPivot tools.
- ~ Working with large sets of Excel data.
- ~ Conditional function.
- ~ Lookup functions.
- ~ Text based functions.
- ~ Auditing and Excel worksheet.
- ~ Protecting Excel worksheets and workbooks.
- ~ MasteringMasteringExcelExcel"What"What--if?"if?"Tools?Tools?
- ~ Automating Repetitive Tasks in Excel with Macros.Automating Repetitive Tasks in Excel with Macros.
- ~ Macro Recorder Tool.Macro Recorder Tool.
- ~ Excel VBA Concepts.Excel VBA Concepts.
- ~ Advance VBA.Advance VBA.
- ~ Preparing and Cleaning Up Data withPreparing and Cleaning Up Data withVBA.VBA.
- ~ VBA to Automate Excel Formulas.VBA to Automate Excel Formulas.
- ~ Preparing Weekly Report.Preparing Weekly Report.
- ~ Working with Excel VBA User Forms.Working with Excel VBA User Forms.
- ~ Importing Data from Text Files.Importing Data from Text Files.

#### => Business Statistics :

- ~ Descriptive Analytics.
- ~ Inferential Statistics.

- ~ Hypothesis Test 1 & 2.
- ~ Covariance
- ~ Correlation.
- ~ Regression.
- ~ Conjoint & Discriminant Analysis.
- ~ Discrete Uniform Distribution.
- ~ Continuous Uniform Distribution.
- ~ Binomial Distribution.
- ~ Poisson Distribution.
- ~ Normal Distribution.
- ~ Sampling Techniques.
- ~ T Distribution.
- ~ Hypothesis Testing and Confidence Intervals.
- ~ Chi Square Test and Distribution.
- ~ Bayes Theorem.

=> Visual Analyst :

- ~ Talking about Business Intelligence.
- ~ Tools and Methodologies used in BI.
- ~ Why Visualization is getting more popular.
- ~ Why Tableau?
- ~ Gartner Magic Quadrant of Market Leaders.
- ~ Future business impact of BI.
- ~ Let's Explore
- ~ Tableau Products.
- ~ Tableau Architecture.
- ~ BI Project Execution.
- ~ Tableau Installation in local system.
- ~ Introduction to Tableau Prep.
- ~ Tableau Prep Builder User Interface.
- ~ Data Preparation techniques using Tableau Prep Builder tool
- ~ How to connect Tableau with different data source.
- ~ Visual Segments.
- ~ Visual Analytics in depth.
- ~ Filters, Parameters & Sets.
- ~ Tableau Calculations using functions.
- ~ Tableau Joins.
- ~ Working with multiple data source (Data Blending).
- ~ Building Predictive Models.
- ~ Dynamic Dashboards and Stories.
- ~ Sharing your Reports.
- ~ Tableau Server.
- ~ User Security.
- ~ Scheduling
- ~ PDF File.
- ~ JSON File.
- ~ Spatial File.
- ~ Statistical File.
- ~ Microsoft SQL Server.
- ~ Salesforce
- ~ AWS
- ~ Azure.
- ~ Google Analytics.
- ~ R
- ~ Python
- ~ Hadoop
- ~ OneDrive
- ~ Microsoft Access.
- ~ SAP HANA.
- ~ SharePoint.
- ~ Snowflake.
- ~ Subject
- ~ Planning.
- ~ Pen & Paper approach.
- ~ Tools
- ~ Color theme.
- ~ Shapes
- ~ Fonts
- ~ image Selection.
- ~ text position.
- ~ visual placing.
- ~ Story layout & design.
- ~ Dashboard planning.
- ~ Power BI introduction and overview. Preview
- ~ Key Benefits of Power BI.
- ~ Power BI Architecture.
- ~ Power BI Process.
- ~ Components of Power BI.
- ~ Power BI Building Blocks.
- ~ Power BI vs other BI tools.
- ~ Power Installation.
- ~ Overview of Power BI Desktop.
- ~ Data Sources in Power BI Desktop.
- ~ Connecting to a data Sources.
- ~ Query Editor in Power BI.
- ~ Views in Power BI.
- ~ Field Pane.
- ~ Visual Pane.
- ~ Custom Visual Option.

- ~ Filters.
- ~ Introduction to using Excel data in Power BI.
- ~ Exploring live connections to data with Power BI.
- ~ Connecting directly to SQL Azure, HD Spark, SQL
- ~ Server Analysis Services/ My SQL.
- ~ Introduction to Power BI Development API.
- ~ Import Power View and Power Pivot to Power BI.
- ~ Power BI Publisher for Excel.
- ~ Content packs.
- ~ Introducing Power BI Mobile.
- ~ Power Query Introduction.
- ~ Query Editor Interface.
- ~ Clean and Transform your data with Query Editor.
- ~ Data Type.
- ~ Column Transformations vs Adding Columns.
- ~ Text Transformations.
- ~ Cleaning irregularly formatted data Transpose.
- ~ Date and Time Calculations.
- ~ Advance editor: Use Case.
- ~ Query Level Parameters.
- ~ Combining Data Merging and Appending.
- ~ Data Modelling.
- ~ Calculated Columns.
- ~ Measures/New Quick Measures.
- ~ Calculated Tables.
- ~ Optimizing Data Models.
- ~ Row Context vs Set Context.
- ~ Cross Filter Direction.
- ~ Manage Data Relationship.
- ~ Why is DAX important?
- ~ Advanced calculations using Calculate functions
- ~ DAX queries.
- ~ DAX Parameter Naming.
- ~ Time Intelligence Functions.
- ~ Types of visualization in a Power BI report.
- ~ Custom visualization to a Power BI report.
- ~ Matrixes and tables.
- ~ Getting started with color formatting and axis properties.
- ~ Change how a chart is sorted in a Power BI report.
- ~ Move, resize, and pop out a visualization in a Power BI report.
- ~ Drill down in a visualization in Power BI.
- ~ Drill Through.
- ~ Histograms
- ~ Basic Area chart.
- ~ Combo Chart in Power BI.
- ~ Customize visualization title, background, and legend.
- ~ Doughnut charts in Power BI.
- ~ Scatter Charts in Power BI.
- ~ Funnel charts in Power BI.
- ~ KPI Visuals.
- ~ Radial Gauge charts in Power BI.
- ~ Bookmarks in Power BI.
- ~ Slicers in Power BI.
- ~ Filters
- ~ Report Level Parameters.
- ~ Z Order.
- ~ Waterfall charts in Power BI.
- ~ Create a Power BI dashboard.
- ~ Dashboard tiles in Power BI.
- ~ Pin a tile to a Power BI dashboard from a report.
- ~ Pin an entire report page to a Power BI dashboard.
- ~ Data alerts in Power BI service.
- ~ Add an image, text box, video, hyperlink or web code to your dashboard.
- ~ Configuring a Dashboard.
- ~ Power BI Q&A.
- ~ Display a dashboard tile in Focus mode.
- ~ Power BI embedded.
- ~ Row Level Security in Power BI.
- ~ Report Server Basics.
- ~ Refresh a dataset created from a Power BI Desktop file local.
- ~ Refresh a dataset created from a Power BI Desktop file cloud.
- ~ Web Portal.
- ~ Paginated Reports.
- ~ Data Gateways.
- ~ Scheduled Refresh.
- ~ Resources (Rest API/ SOAP APIs/ URL Access).
- ~ R Integration in Power BI Desktop.
- ~ R Powered Custom Visuals.
- ~ Creating R visuals in Power BI.
- ~ R Visuals in Power BI Service.
- ~ R Scripts Security.
- ~ Creating visual using Python.

# Data Science Interview

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/Data-Science-Interview>

## Course Description :-

This course is designed for an individual trying to transition towards various data science careers in the industry. Keeping all the hurdles in mind that we generally face during your transition so that your journey will be smooth and without losing any opportunity, you will be able to transition in the industry. Discuss, Collaborate, Participate and Win the Race.

## Course Features :-

- => Online Instructor-led learning
- => Meet with Achiever
- => Proper Roadmap
- => One-One Resume Building
- => Lifetime Dashboard access
- => Doubt clearing
- => Quiz in every module
- => Career Counselling
- => Assessments
- => Mock Interview
- => Certificate
- => 850 + interview question live discussion
- => AI leader talk(Panasonic, EY, Verizon, Apple, and many)

## What you will learn :-

- => Profile Building
- => System Designing
- => Domain Understanding
- => Common Mistakes
- => Project Management
- => ML Interview Questions
- => DL Interview Questions
- => NLP Interview Questions
- => Stats Interview Questions
- => Python Interview Questions
- => Computer Vision Interview Questions
- => Mock Interview

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Basic Understanding of Python
- => Basic Understanding of ML
- => Basic Understanding of DL

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Introduction about Data science industry and does and don't in your profile and public profile building with iNeuron team. :

- ~ *Induction & Course Introduction*
- ~ *Impact of Data Science in today's world & Roles in Data Science*

=> Python Interview Questions :

- ~ *50 Interview Questions Day1*
- ~ *50 Interview Questions Day2*

=> Stats Interview Questions :

- ~ *60 Interview Questions Day1*
- ~ *60 Interview Questions Day2*

=> Machine Learning Interview Question and Solution Design :

- ~ *40 Interview Questions Day1*
- ~ *40 Interview Questions Day2*
- ~ *40 Interview Questions Day3*
- ~ *40 Interview Questions Day4*
- ~ *40 Interview Questions Day5*
- ~ *40 Interview Questions Day6*
- ~ *40 Interview Questions Day7*

=> Deep Learning & Computer Vision Interview Questions :

- ~ *40 Interview Questions Day1*
- ~ *40 Interview Questions Day2*
- ~ *40 Interview Questions Day3*
- ~ *40 Interview Questions Day4*
- ~ *40 Interview Questions Day5*

=> NLP Interview Questions :

- ~ *40 Interview Questions Day1*
- ~ *40 Interview Questions Day2*

=> Project deployment & Solution Design Life Cycle Interview Questions :

- ~ *50 Interview Questions Day1*
- ~ *50 Interview Questions Day2*

=> Meet with multiple people who has made recent transition :

- ~ *Ask anything*
- ~ *Get suggestion and roadmap*

=> Generic Project architecture design for interview :

- ~ *How Project start in Industry?*
- ~ *Business Expectation*
- ~ *Data Sharing Agreement*
- ~ *Proof of Concept*
- ~ *Master Data Management*
- ~ *High Level Architecture Design*
- ~ *Low Level Architecture Design*
- ~ *Project Wireframe*
- ~ *Data Accusation*
- ~ *Code Level Architecture*
- ~ *Tech Identification*
- ~ *Team Building*
- ~ *Project Delivery Methodology*
- ~ *Project Timeline Calculation*
- ~ *Infrastructure Setup*
- ~ *Project Cost Estimation*
- ~ *Project Kickoff*

=> Resume Design and projects by iNeuron one to one resume building :

- ~ *Resume Template Selection*
- ~ *Tech Stack Involvement*
- ~ *Project Selection & Alignment as per your Experience*
- ~ *Project Details*
- ~ *Your Involvement in Project*
- ~ *Tech Stack for Project*
- ~ *Fine Tuning of your Resume*
- ~ *Proof Reading*
- ~ *LinkedIn & GitHub Update*
- ~ *Applying for Job*
- ~ *Resume Finalization based on Job Description*
- ~ *One to One Discussion with iNeuron Team*

=> Mock interview with Krish and Sudhanshu one to one live/Offline :

- ~ *Fact Check*
- ~ *All round Interview*
- ~ *Review*
- ~ *Feedback*
- ~ *Suggestions*

=> Interaction with many achievers who has done a recent transition in data science on all level :

- ~ *Interaction with Ineuron Achievers of all Ages*
- ~ *Ask Anything*
- ~ *Expert Advice*
- ~ *Doubt Clarification*

=> Final touch of everything for next journey and launch :

- ~ *Check the JOSH!!!*



# Mern stack job preparation

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : WEB DEVELOPEMENT INTERVIEW

Course link : <https://ineuron.ai/course/Mern-stack-job-preparation>

## Course Description :-

This section is a Step by step guide to common questions, how to solve them, optimize, and present them during tech interviews.

Learn exactly how you need to answer difficult questions and the framework you need for ANY kind of questions they throw at you.

Become a better MERN developer by mastering interview fundamentals.

## Course Features :-

- => Challenges
- => Interview Questions
- => Solving Problems
- => Multiple Technologies
- => Quizzes
- => Completion Certificate

## What you will learn :-

- => Interview Questions
- => Handling Counter Questions
- => Solving the Questions in Real Time
- => How to answer smartly

## Requirements :-

- => Prior Knowledge in Mern Stack
- => A System with Internet Connectivity

## Instructors :-

- => Syed Ashraf :  
~ Full Stack Engineer at TensorGo Technologies

## Curriculum details :-

### => HTML :

- ~ What is Doctype ? Preview
- ~ Complete Boilerplate of HTML
- ~ What is the difference between Elements, Attributes and Tags ?
- ~ What are self-closing tags ?
- ~ How to create a link ?
- ~ What are the common tags in Table ?
- ~ How to create a nested webpage in HTML ?
- ~ What are empty elements ?
- ~ How to create Forms ?
- ~ Describe HTML layout
- ~ What is the difference between Progress Bar and Meter ?
- ~ What is the difference between anchor and link tag ?

### => CSS :

- ~ How can you use CSS in a webpage ? Preview
- ~ Dominance between inline, internal and external
- ~ How to use image in HTML ?
- ~ What are some CSS Selectors ?
- ~ What is CSS Box Model ?
- ~ What is Embedded Style Sheet ?
- ~ What is Margin ?
- ~ What is Padding ?
- ~ What is Z-index ?
- ~ What is opacity in image ?
- ~ What is the difference between Class and Id ?
- ~ What is the difference between background color and color ?
- ~ What is the difference between display hidden and visibility hidden ?
- ~ What is universal selector ?
- ~ Dominance between inline, internal and external

### => Javascript :

- ~ What is Javascript ? Preview
- ~ What are some Data Types in Javascript ?
- ~ What are different types of Popups ?
- ~ How to write a function in javascript ?
- ~ What are named and anonymous functions ?

- ~ What are some basic loops in Javascript ?
- ~ What are some disadvantages of javascript ?
- ~ What are the different ways to Console ?
- ~ How to Comment in Javascript ?
- ~ What is Null ?
- ~ What is the difference between == and === ?
- ~ What is the difference between Undeclared and Undefined variables ?
- ~ What is type of operator ?
- ~ Write a Sum of 2 numbers using function ?

=> Node.js :

- ~ What is Node.js ?
- ~ What are the benefits of Node.js ?
- ~ Where can be Node.js used ?
- ~ What are the types of API's in Node.js ?
- ~ What is npm ?
- ~ What are modules ?
- ~ Why is Node.js preferred over other languages ?
- ~ What is package.json ?
- ~ What is Express.js ?
- ~ Create a GET API in Express.js.
- ~ What are Streams ?
- ~ How to uninstall, install and update npm package ?
- ~ Create a server in Node.js and return "Hello World".
- ~ Write a code to get post a query in Express.js.
- ~ Which are the arguments available to an Express JS route handler function?
- ~ How can you allow CORS in Express.js?
- ~ How can you deal with error handling in Express.js ? Explain with an example.
- ~ Write the code to start serving static files in Express.js.
- ~ How can we render a plain HTML?
- ~ What is the purpose of module.exports?

=> React.js :

- ~ What is React.js ?
- ~ What are the advantages of using React?
- ~ What is JSX?
- ~ What are the differences between functional and class components?
- ~ What are the differences between controlled and uncontrolled components?
- ~ Explain Strict Mode in React.
- ~ Explain React state and props.
- ~ Explain React Hooks.
- ~ What are the different ways to style a React component?
- ~ What are keys in React?
- ~ How to pass data between react components?
- ~ What is prop drilling in React?
- ~ What are the features of React?
- ~ What are the limitations of React?
- ~ Why cant browsers read JSX?
- ~ In React, everything is a component. Explain.
- ~ What is the purpose of render() in React
- ~ How can you embed two or more components into one?
- ~ What are the different phases of React components lifecycle?
- ~ What do you understand by refs in React?
- ~ How are forms created in React?
- ~ What is Redux?
- ~ What are the three principles that Redux follows?
- ~ What do you understand by Single source of truth?
- ~ List down the components of Redux.
- ~ What is React Router?
- ~ Why do we need a Router in React?

# C Coding Interview Preparation

---

Topic Name : PROGRAMMING

Sub-topic Name : C

Course link : <https://ineuron.ai/course/C-Coding-Interview-Preparation>

## Course Description :-

This course is designed mostly for C test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> C Theoretical Test

=> C Practical Test

=> C Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> C Coding Test :

~ C Test 1

~ C Test 2

~ C Test 3

~ C Test 4

~ C Test 5

~ C Test 6

~ C Test 7

~ C Test 8

~ C Test 9

~ C Test 10

~ C Test 11

~ C Test 12

~ C Test 13

~ C Test 14

~ C Test 15

~ C Test 16

~ C Test 17

~ C Test 18

~ C Test 19

~ C Test 20

# Digital Marketing Foundation

---

Topic Name : DIGITAL MARKETING

Sub-topic Name : DIGITAL MARKETING MASTERS

Course link : <https://ineuron.ai/course/Digital-Marketing-Foundation>

## Course Description :-

Grow your digital marketing results faster through the power of growth hacking! In this industry-leading course, you'll discover the extraordinary benefits of digital metrics, including lean analytics, web traffic, digital conversion funnels, and LTV and CAC calculations.

## Course Features :-

- => Course for pre launch business owners who have no idea where to get started
- => For starting a freelancing techniques in Marketing field

## What you will learn :-

- => From Scratch grow business online
- => Work from home as a Freelancer Marketer
- => Make money as an Affiliate Marketer

## Requirements :-

- => No Experience required
- => Computer with Internet connectivity
- => Basic Programming understanding

## Curriculum details :-

- => Digital Marketing Class 1 - Introduction to Digital Marketing :  
~ Introduction Preview
- => Digital Marketing Class 2 - Basics of Websites, Selecting Domain, and Activating Free Hosting
- => Digital Marketing Class 3 - Plugins and Integrations (Part - 1)
- => Digital Marketing Class 4 - Plugins and Integrations (Part - 2)
- => Digital Marketing Third Party Website Integrations Google and Bing
- => Digital Marketing - Market Research - With No Tools
- => Digital Marketing 6.2 - Market Research - No Tools
- => Digital Marketing Class 7.1 - Introduction to SEO and Basic Research
- => Digital Marketing Class 7.2 - Research With Free Tools - Google Trends
- => Digital Marketing 8 1 Free Tool
- => Digital Marketing Class 8.2 - Paid Tools
- => Digital Marketing Class 9.1 - Intro to Google Analytics
- => Digital Marketing Class 9.2 - All Tabs of Google Analytics
- => Digital Marketing Class 10.1 - Goals, Reports, Dashboard
- => Digital Marketing Class 10.2 - Search Console
- => Digital Marketing Class 11.1 - Website Optimization Tools
- => Digital Marketing Class 11.2 - Email Marketing
- => Digital Marketing Conclusion

# System Design

---

Topic Name : SYSTEM DESIGN

Sub-topic Name : SYSTEM DESIGN

Course link : <https://ineuron.ai/course/System-Design>

## Course Description :-

A comprehensive course on System Design covering different case studies. This course provides both theoretical and step by step approaches to solving a particular case study.

## Course Features :-

- => Self-Paced Classes
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => System Design Concepts
- => What are Proxies ?
- => Caching
- => Rest API
- => Scaling
- => Database Replication

## Requirements :-

- => Prior knowledge in data structures

## Instructors :-

- => Vaibhav Kumar :
  - ~ Keen Problem solver and a Competitive coder, Ex-Gainsight

## Curriculum details :-

### => Theory :

- ~ Introduction to System Design Preview
- ~ Components of System Design Preview
- ~ Client Server Architecture
- ~ What are Proxies ? Preview
- ~ Introduction to Data and Data Flow
- ~ Database and Types of Database
- ~ Anatomy of Applications and Services
- ~ API - (Application Programming Interface)
- ~ Caching
- ~ Rest API Preview
- ~ Message Queue
- ~ Publisher Subscriber Model
- ~ Performance Metrics
- ~ Performance Metrics of components
- ~ Fault and Failure
- ~ Scaling
- ~ Database Replication
- ~ CAP theorem
- ~ Database Sharding
- ~ Key, Range and Directory based sharding
- ~ Hashing
- ~ Consistent Hashing
- ~ Foundation of System Design Interview

### => Case Study :

- ~ System Design Tiny Url
- ~ System Design Api Rate Limiter
- ~ System Design a text sharing service like Pastebin
- ~ System Design Chat Messenger
- ~ System Design Twitter
- ~ System Design Distributed Web Crawler
- ~ System Design a Movie Ticket Booking System
- ~ System Design Uber

# Web Designing using Wordpress

---

Topic Name : K12

Sub-topic Name : CLASS9

Course link : <https://ineuron.ai/course/Web-Designing-using-Wordpress>

## Course Description :-

If you are interested in building websites, WordPress is what you need to learn and here we introduce website design and creation by utilizing different kinds of themes and templates available in WordPress. Students who complete this course will gain hands-on practical experience in building various interactive and beautiful websites by using WordPress.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to WordPress
- => WordPress Themes
- => WordPress Plugins
- => WordPress to create Blog
- => Creating site pages using WordPress
- => WordPress Hosting
- => Projects on WordPress

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shivan Kumar :

~ Associate Data Scientist, Mentor, and Kaggle Master with 2 Years of Experience. Experience in building models that translate data points into business insights. Highly accurate at collecting, analyzing, and interpreting large datasets, developing machine learning, deep learning, Chatbots models, and deploying the solutions on the cloud. I also love to participate in Hackathons. In my free time, I like to mentor students to learn Data Science and achieve their goals.

## Curriculum details :-

=> Course Introduction :

- ~ Course Introduction
- ~ Who is this course for?
- ~ Course Overview
- ~ Course Outcome
- ~ What is Web Development?
- ~ Why Web development?
- ~ Web development using Wordpress

=> WordPress Introduction :

- ~ What is WordPress?
- ~ Features of WordPress
- ~ Applications of WordPress

=> Installing WordPress :

- ~ Download WordPress
- ~ Installing in System
- ~ Verifying the Installation
- ~ How to use WordPress?
- ~ General site settings

#### => Wordpress Themes :

- ~ *What is Theme?*
- ~ *Select, install and activate a theme*
- ~ *WordPress theme directory*
- ~ *Practical:- Theme customization*

#### => Wordpress Plugins :

- ~ *What is Plugins?*
- ~ *Use of Plugins in web development*
- ~ *Installation of Plugins*
- ~ *How to create Plugin files?*
- ~ *Practical:- Play with different different WordPress Plugin*

#### => WordPress to Create Blog :

- ~ *What is Blog?*
- ~ *Steps to create blog using WordPress*
- ~ *WordPress blog examples*
- ~ *Career in Blogging*
- ~ *Practical:- Creation of blog using WordPress*

#### => Assignment 1: :

- ~ *Write a blog related to your favorite history chapter.*

#### => Creating Site Pages using WordPress :

- ~ *What is page in WordPress?*
- ~ *Difference between page and post.*
- ~ *How do WordPress pages work?*
- ~ *WordPress page template*
- ~ *Adding a page*
- ~ *Practical:- Steps to create Pages in WordPress*

#### => WordPress Hosting :

- ~ *What is Hosting?*
- ~ *What is Domain?*
- ~ *Why hosting is required?*
- ~ *Difference between Web hosting and WordPress hosting?*

#### => Projects :

- ~ *Design your own portfolio using WordPress.*
- ~ *Create a blog using WordPress.*

#### => Summary :

- ~ *Course Conclusion*
- ~ *Future Scope of WordPress*

# Job Guaranteed Big Data Bootcamp

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA MASTERS

Course link : <https://ineuron.ai/course/Job-Guaranteed-Big-Data-Bootcamp>

## Course Description :-

By using the tools and processes that big data scientists and engineers use on a daily basis, you will be intelligent enough to understand the insights that big data may offer. With a general understanding of how large data is arranged, examined, and evaluated, you can make better business judgments. This unique industry program will help to learn the entire stack of Big Data and be ready to crack jobs in leading organizations.

## Course Features :-

- => High Quality Premium Big Data Labs Included
- => Full stack Big Data certification
- => Job guarantee otherwise refund
- => One year of internship Anytime
- => 1:1 Personalized Mentorship
- => Revision Classes
- => Online Instructor-led learning: Live teaching by instructors
- => 20 + hands-on industry real-time projects.
- => 200 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Doubt clearing through mail and skype support team
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building Anytime
- => Career guidance Anytime
- => Interview Preparation Anytime
- => Regular assessment
- => Job Fair and Internal Hiring
- => Mock Interview Anytime

## What you will learn :-

- => High Quality Premium Big Data Labs Included
- => Big Data
- => Hadoop
- => HDFS
- => YARN
- => Linux
- => AWS EC2
- => AWS IAM
- => AWS S3
- => AWS SNS
- => AWS DMS
- => AWS RDS
- => AWS Redshift
- => Hbase
- => Sqoop
- => Confluent



- => Atlas
- => Ambari
- => Databricks

### Requirements :-

- => Premium Big Data Labs Included
- => System with Internet Connection
- => Interest to learn
- => Dedication

### Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Shashank Mishra :

~ Experienced Data Engineer with a demonstrated history of working in service and product companies. Solved data mysteries for different domains like Aviation, Pharmaceutical, FinTech, Telecom and Employee Services. Have designed scalable & optimized data pipelines to handle PetaBytes of data, with Batch & Real Time frequency. Got good exposure on different BigData frameworks (Hadoop, Spark, Hive, Sqoop, Flume, Flink, Kafka, Docker), Databases (MySQL, HBase, Cassandra, Redshift, Elastic Search), AWS Services (S3, Lambda, EMR, Glue, Cloudwatch, Redshift, SNS, SQS, Athena, Appflow), Dashboarding Tools (Grafana, Kibana, QuickSight, DataDog, Data Studio), Monitoring Tools (Airflow, Azkaban), Web Development (HTML, CSS, Scala Play, Django, Rest API, JavaScript, Ajax, JQuery), Good command over programming languages (Python, Java, Scala, Shell Scripting) and strong Data Structures & Algorithm fundamentals.

### Curriculum details :-

=> Big Data Introduction: Introduction :

- ~ What is Big Data?
- ~ Evolution of Big Data
- ~ Why to learn Big Data technologies?
- ~ Examples of Big Data
- ~ Who is using Big Data?
- ~ Why is Data so important?
- ~ Characteristics of Big Data
- ~ Challenges of Big Data
- ~ Data scale
- ~ Manage, store and process Big Data
- ~ 5 Vs of Big Data
- ~ Sources of Data flood
- ~ Exploding data problem
- ~ OLTP and OLAP
- ~ Operational vs Analytical Big Data
- ~ Possible solutions: scaling up vs. scaling out
- ~ Challenges of scaling up and scaling out

=> Hadoop fundamentals :

- ~ What is Hadoop?
- ~ Hadoop in layman's term
- ~ History and timeline of Hadoop
- ~ Evolutionary features of Hadoop
- ~ Why hadoop in demand?
- ~ Components of Hadoop ecosystem
- ~ Hadoop architecture
- ~ How hadoop solve data explosion problem?
- ~ Differences between Hadoop 1.X and Hadoop 2.X and Hadoop 3.X
- ~ Hadoop 1.x 2.x 3.x architecture, components and working of those Components

=> HDFS :

- ~ Design of HDFS
- ~ HDFS architecture
- ~ HDFS features
- ~ Name node and data node
- ~ Secondary name node
- ~ Job tracker
- ~ Task tracker
- ~ Client nodes
- ~ Explain master-slaves
- ~ Pseudo-distributed
- ~ Fully-distributed
- ~ Data replication
- ~ How does a file read and write work?
- ~ Local file system and HDFS
- ~ Rack awareness
- ~ Arrangement of racks
- ~ Arrangement of machines and racks
- ~ Checkpointing in Hadoop
- ~ Benefits of replica placement and rack awareness
- ~ URL And URN
- ~ HDFS commands
- ~ HDFS web interface
- ~ Fault tolerance
- ~ Name node failure management
- ~ Anatomy of file read and write from HDFS
- ~ Important java classes to write data to HDFS

- ~ Inputsplit and data blocks difference
- ~ Why Is the block size 128 MB?
- ~ Recordreader
- ~ Inputformat
- ~ Default Inputformat: TextInputformat
- ~ Outputformat
- ~ What is partitioner?
- ~ Using partitioner
- ~ Map only job
- ~ Flow of operations in MapReduce
- ~ Serialization in MapReduce

#### => HDFS Operations :

- ~ Start HDFS
- ~ Listing files in HDFS
- ~ Writing a file into HDFS
- ~ Reading data from HDFS
- ~ Shutting down HDFS
- ~ Listing contents of directory
- ~ Displaying and printing disk usage
- ~ Moving files & directories
- ~ Copying files and directories
- ~ Displaying file contents

#### => YARN :

- ~ What is Yarn?
- ~ Why Yarn?
- ~ Classic MapReduce v/s Yarn
- ~ Yarn architecture
- ~ Resource Manager
- ~ Node manager
- ~ Application master
- ~ Node manager containers
- ~ Resource manager components
- ~ Advantages & disadvantages of Yarn
- ~ Yarn applications
- ~ Scheduling in Yarn
- ~ Fair Scheduler
- ~ Fault Tolerance
- ~ Schedulers in Yarn
- ~ FIFO scheduler
- ~ Capacity scheduler
- ~ Fair scheduler

#### => Setting up Our Linux Space :

- ~ Downloading necessary tools
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH Clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting Up Passwordless SSH

#### => Linux Commands :

- ~ Linux Commands Part1
- ~ Linux Commands Part2
- ~ Linux Commands Part3
- ~ Linux Commands Part4
- ~ Cat Command Usages

#### => Introduction :

- ~ What is Hive?
- ~ Hive Vs Map Reduce
- ~ Hive Vs Relational databases
- ~ Installation and setup of Hive
- ~ Introduction to CouchDB
- ~ Why CouchDB?
- ~ History of CouchDB
- ~ Features of CouchDB
- ~ Advantages of CouchDB
- ~ Disadvantages of CouchDB
- ~ What is Neo4j?
- ~ Why Neo4j?
- ~ Features of Neo4j
- ~ Advantages of Neo4j
- ~ Neo4j Architecture
- ~ Applications of Neo4j
- ~ Data model of Neo4j
- ~ Building Blocks of Neo4j

#### => Hive Architecture :

- ~ Hive architecture
- ~ Different modes of Hive
- ~ Hive Functions: Built-in & UDF
- ~ Datatypes in Hive
- ~ Operators in Hive
- ~ How to create and drop databases?
- ~ Hive create table: internal table, external table, alter, drop

#### => DDL and DML commands in Hive :

- ~ Hive DDL

- ~ Create
- ~ Show
- ~ Describe
- ~ Use
- ~ Drop
- ~ Alter
- ~ Truncate
- ~ Hive DML
- ~ Load
- ~ Select
- ~ Insert
- ~ Delete
- ~ Update
- ~ Export
- ~ Import
- ~ Hive view and index
- ~ What is Hive metastore?
- ~ How to install and configure Hive metastore?
- ~ What is Hive data modeling?

=> Hive partitioning and bucketing :

- ~ Partitioning in Hive
- ~ Static and dynamic partitioning
- ~ Bucketing in Hive
- ~ Bucketing vs Partitioning
- ~ What is Hive query language(HQL)?

=> HQL language :

- ~ HiveQL- Where
- ~ HiveQL- Order By
- ~ HiveQL- Group By
- ~ HiveQL- Joins and types
- ~ HiveQL- SubQuery
- ~ Hive ETL: loading JSON, XML, text data
- ~ Working with arrays
- ~ Sort by and order by
- ~ Distribute by and cluster by
- ~ Bucket-map join
- ~ Sort-Merge-Bucket-Map join
- ~ Left semi join

=> Different File formats in Hive :

- ~ File formats in Hive
- ~ Text files
- ~ Input formats in Hive
- ~ Sequence files in Hive
- ~ RC file in Hive
- ~ Sequencefile
- ~ ORC files in Hive
- ~ Avro files
- ~ Parquet file
- ~ Inline index in ORC files
- ~ ORC file configurations in Hive
- ~ SerDe in Hive
- ~ Demo: CSVSerDe
- ~ JSONSerDe
- ~ RegexSerDe
- ~ Analytic and windowing in Hive
- ~ Demo: analytics.hql
- ~ Hcatalog in Hive
- ~ Demo: using\_HCatalog
- ~ Accessing Hive with JDBC
- ~ Demo: HiveQueries.Java
- ~ HiveServer2 and beeline
- ~ Demo: beeline
- ~ Demo: ToUpper.Java and working\_with\_UDF
- ~ Optimizations in Hive
- ~ Demo: Optimizations

=> Introduction of HBase :

- ~ What is HBase?
- ~ HDFS and HBase
- ~ HBase vs RDBMS
- ~ HBase vs HIVE
- ~ HBase storage mechanism
- ~ Feature of HBase
- ~ Applications of HBase

=> HBase installation setup :

- ~ Apache HBase setup
- ~ Hardware recommendations
- ~ Software recommendations
- ~ Installation using cloudera manager
- ~ Basic static configuration

=> HBase architecture :

- ~ Architecture of HBase
- ~ Components of HBase architecture
- ~ Client library
- ~ Zookeeper

- ~ HMaster server
- ~ HBase regions servers

=> HBase commands :

- ~ General commands
- ~ status
- ~ table\_help
- ~ version
- ~ whoami
- ~ Data definition commands
- ~ alter
- ~ alter\_async
- ~ alter\_status
- ~ create
- ~ drop
- ~ drop\_all
- ~ enable
- ~ enable\_all
- ~ exists
- ~ get\_table
- ~ is\_disabled
- ~ is\_enabled
- ~ show\_filters
- ~ Data manipulation commands
- ~ append
- ~ count
- ~ delete
- ~ deleteall
- ~ get\_table
- ~ get\_counter
- ~ put
- ~ truncate
- ~ truncate\_preserve
- ~ Other HBase shell commands
- ~ Admin commands
- ~ Replication commands
- ~ Snapshot commands
- ~ Visibility labels commands
- ~ Security commands

=> CRUD operations using HBase shell :

- ~ What is HBase shell?
- ~ HBase shell usage
- ~ Starting HBase shell
- ~ Creating table
- ~ Inserting a row
- ~ Updating a row
- ~ Retrieving a row
- ~ Retrieving a range of rows
- ~ Deleting a row
- ~ Deleting a table
- ~ Retrieve rows within a time range
- ~ Filter by column value - SingleColumnValueFilter
- ~ Filter by Row id - RowFilter
- ~ Apply multiple conditions - Filterlist

=> Understanding the troubleshooting in HBase :

- ~ Understand the troubleshooting
- ~ Trouble shooting distributed clusters
- ~ Administration from the command line
- ~ How to use the HBase UI?
- ~ How to use the Metrics and the logs?

=> Basic Introduction :

- ~ Challenges with traditional RDBMS
- ~ What is Nosql database?
- ~ History behind the creation of Nosql databases
- ~ Features of Nosql database
- ~ Different types of Nosql databases
- ~ When Nosql should be used?
- ~ Advantages of Nosql
- ~ Disadvantages of Nosql
- ~ Why Nosql database?

=> Introduction and overview of cassandra :

- ~ What is Apache Cassandra?
- ~ History of Cassandra
- ~ Cassandra Database vs Relational Database
- ~ Apache Cassandra features
- ~ Cassandra use cases and applications
- ~ Advantages of Cassandra
- ~ Disadvantages of Cassandra

=> Setup, installtion and configuration :

- ~ Cassandra configuration with datastax
- ~ Understanding different ways to communicate with cassandra
- ~ Using cq/sh

=> Cassandra Architecture :

- ~ Cassandra architecture
- ~ Cassandra data model

- ~ *Cassandra as a distributed database*
- ~ *Node*
- ~ *Data center*
- ~ *Cluster*
- ~ *Commit log*
- ~ *Mem-table*
- ~ *SSTable*
- ~ *Data replication*
- ~ *Write operation*
- ~ *Read operation*
- ~ *Data compaction*

#### => Cassandra - Shell Commands :

- ~ *Help*
- ~ *Capture*
- ~ *Consistency*
- ~ *Copy*
- ~ *Describe tabel*
- ~ *Describe keyspaces*
- ~ *Expand*
- ~ *Exit*
- ~ *Show*
- ~ *Source*

#### => Cassandra Query Language(CQL) :

- ~ *CQL Data Definition Commands*
- ~ *Cassandra CQL Data Types*
- ~ *Creating Database*
- ~ *Creating Keyspace*
- ~ *Use Keyspace*
- ~ *Alter Keyspace*
- ~ *Drop Keyspace*
- ~ *Create Table*
- ~ *Alter table*
- ~ *Drop table*
- ~ *Truncate*
- ~ *Create Index*
- ~ *Drop Index*
- ~ *CQL Data Manipulation Commands*
- ~ *Insert*
- ~ *Update*
- ~ *Delete*
- ~ *Batch*
- ~ *CQL Clauses*
- ~ *Select*
- ~ *Cassandra Where Clause*
- ~ *Cassandra Order by Clause*

#### => Cassandra CRUD Operation :

- ~ *Create data*
- ~ *Update data*
- ~ *Read data*
- ~ *Delete data*
- ~ *Maps*
- ~ *Sets*
- ~ *Lists*
- ~ *Key and indexing*

#### => Introduction to MongoDB :

- ~ *Introduction*
- ~ *key charcristic of MongoDB*
- ~ *Understanding MongoDB ecosystem*
- ~ *Advantages & disadvantages of using MongoDB*

#### => MongoDB installtion and setup :

- ~ *MongoDB installation in local*
- ~ *Setup MongoDB server*
- ~ *Setup MongoDB compass*
- ~ *Exploring the MongoDB compass*
- ~ *MondoDB local server and compass setup*
- ~ *MongoDB atlas setup*

#### => Architecture :

- ~ *Architecture of MongoDB*
- ~ *Understanding databases, collections & documents*
- ~ *Creating databases & collections*
- ~ *Understanding JSON Data*
- ~ *Comparing JSON & BSON*
- ~ *Storage engines*
- ~ *Read path*
- ~ *Write path*
- ~ *Working set*
- ~ *Capped collection*
- ~ *Oplog collection*
- ~ *TTL index*
- ~ *Gridfs*

#### => CRUD opearations :

- ~ *MongoDB data types*
- ~ *Finding, Inserting, Deleting & Updating elements*
- ~ *Querying the documents*

- ~ Bulk insert operations
- ~ Updating multiple document
- ~ Limiting documents
- ~ Understanding insertOne vs insertMany()
- ~ Updateone() vs updateMany()
- ~ Understanding find() & fetchall()
- ~ Understanding "deleteOne()" & "deleteMany()"
- ~ Filtering documents

=> Schema design and data modeling :

- ~ Why do we use Schemas?
- ~ What is data modeling?
- ~ RDBMS and MongoDB data modeling difference
- ~ Embedding document
- ~ Reference document
- ~ Structuring documents
- ~ Understanding relations
- ~ One To One
- ~ One To Many
- ~ Many To Many

=> Database administration in MongoDB :

- ~ Database status
- ~ Troubleshooting issues
- ~ Current operations
- ~ Rotating log files
- ~ Users and roles
- ~ Copy and clone database
- ~ DB and collection stats
- ~ Explain plan
- ~ Profiling
- ~ Changing configuration files
- ~ Upgrading the database

=> Working with python driver :

- ~ Splitting work between the Driver & the Shell
- ~ Preparing our project
- ~ Installing Visual Studio Code or Pycharm
- ~ Installing the Python
- ~ Connecting Python & the MongoDB cluster
- ~ Storing products in the database
- ~ Fetching data from the database
- ~ Getting a single product
- ~ Editing & deleting products
- ~ Implementing pagination
- ~ Adding an index
- ~ Adding an index to make the Email unique
- ~ Adding user sign-in

=> Replication in MongoDB :

- ~ Concept of replication
- ~ Replicaset member roles
- ~ Voting and electing primary
- ~ Role of oplog in replication
- ~ Read and write concern
- ~ Arbiter, Hidden and Delayed replica node
- ~ Priority settings
- ~ Replicaset nodes health check
- ~ Concept of resyncing the nodes
- ~ Rollbacks during failover
- ~ Keyfile authentication

=> MongoDB scalability :

- ~ Concept of scalability
- ~ Sharding concept
- ~ Shardkey and chunks
- ~ Choosing shardkey
- ~ Sharding components
- ~ Types of sharding
- ~ Balanced data distribution
- ~ Sharded and non-sharded collection
- ~ Sharded replicaset
- ~ Tag aware sharding

=> MongoDB Monitoring :

- ~ MMS manager
- ~ Ops manager
- ~ MongoDB utility commands
- ~ MongoDB developer tools
- ~ MongoDB client drivers

=> Installation :

- ~ Installation of CouchDB on Windows
- ~ Installation of CouchDB on Ubuntu

=> Neo4j CQL :

- ~ Introduction to Neo4j CQL
- ~ Neo4j CQL clauses
- ~ Neo4j CQL Functions
- ~ Neo4j CQL Data Types
- ~ Neo4j CQL operators

- ~ *Neo4j CQL Boolean operators*
- ~ *Neo4j CQL Comparison operators*
- ~ *Node Creation in Neo4j CQL*
- ~ *Relationship creation in Neo4j CQL*

#### => Introduction to Kafka :

- ~ *Introduction to Apache Kafka*
- ~ *History of Apache Kafka*
- ~ *Why Apache Kafka?*
- ~ *What is messaging system?*
- ~ *Kafka message flow*
- ~ *Committed vs uncommitted messages*
- ~ *Kafka operations*
- ~ *Kafka communication*
- ~ *Advantages of Kafka*
- ~ *Kafka use-cases*

#### => Architecture of kafka :

- ~ *Kafka architecture*

#### => Installation of kafka :

- ~ *Installation of Kafka in local system*
- ~ *kafka setup on cloud*
- ~ *Kafka - Confluent*
- ~ *Kafka - Confluent platform*

#### => Kafka CLI :

- ~ *Introduction to Kafka CLI*
- ~ *Creating Kafka topic*
- ~ *Listing topics in Kafka CLI*
- ~ *Deleting topics in Kafka CLI*
- ~ *Getting details of Kafka topic*
- ~ *Producing data to Kafka topic*
- ~ *Consuming data to Kafka topic*
- ~ *Purging a Kafka topic*

#### => Zookeeper in Kafka :

- ~ *Why Zookeeper is used in Kafka?*
- ~ *Role of Zookeeper in Kafka*

#### => Kafka APIs :

- ~ *Introduction to Kafka API*
- ~ *Different types of Kafka API*
- ~ *Producer API*
- ~ *Consumer API*
- ~ *Streams API*
- ~ *Connector API*
- ~ *Kafka integration with Spark*

#### => Introduction to Nifi :

- ~ *What is Apache NiFi?*
- ~ *Architecture of Apache NiFi*
- ~ *Characteristics of Apache NiFi*
- ~ *Advantages of Apache NiFi*

#### => Installation of Apache NiFi :

- ~ *Environment Setup*
- ~ *Setting up Windows Developer Environment*
- ~ *Setting up Linux Developer Environment*
- ~ *Setting up Mac Developer Environment*

#### => Apache NiFi Repository :

- ~ *Flowfile Repository*
- ~ *Content Repository*
- ~ *Provenance Repository*

#### => Apache NiFi User Interface :

- ~ *Introduction to Apache NiFi User Interface*
- ~ *NiFi Canvas*
- ~ *NiFi Processors*
- ~ *Process Groups and Templates*
- ~ *Apache NiFi components*

#### => Apache NiFi Processors :

- ~ *Introduction to Apache NiFi Processors*
- ~ *GenerateFlowFile*
- ~ *LogAttribute*
- ~ *Functionality of NiFi Processors*

#### => Getting started with Spark :

- ~ *What is Spark and what it is purpose?*
- ~ *Why Spark is faster than Hadoop?*
- ~ *What is in-memory computation?*
- ~ *Features of Spark*
- ~ *Explain unified architecture of Spark*
- ~ *Components of the Spark unified architecture*
- ~ *Downloading and installing Spark standalone*
- ~ *Scala and Python overview, launching and using Sparks Scala and Python shell*
- ~ *Spark execution context*
- ~ *Driver*
- ~ *Executor*
- ~ *Master*

~ Worker

## => The Resilient Distributed Datasets (RDD) :

- ~ Overview of RDD's
- ~ Features of RDD
- ~ RDD operations
- ~ RDD and pair RDDs and RDD performance
- ~ Flat maps and filters
- ~ Data loading in RDD
- ~ RDD deep dive
- ~ Partitions
- ~ Dependencies
- ~ Transformation in RDD
- ~ Action in RDD
- ~ Map
- ~ Filter
- ~ Filter map
- ~ Group by
- ~ Group by key
- ~ Reduce by key
- ~ Map partitions
- ~ Union
- ~ Join
- ~ Distinct
- ~ Coalesce
- ~ Key by
- ~ Partition by
- ~ Zip
- ~ Collect
- ~ Reduce by key
- ~ Aggregate
- ~ RDD Lineage
- ~ DAG for RDD
- ~ Limitations of Spark RDD
- ~ RDD persistence
- ~ Shared variables and broadcast variables
- ~ Accumulators

## => Spark SQL, DataFrames and Datasets :

- ~ Introducing Spark SQL
- ~ Introducing datasets and DataFrame
- ~ Data sources
- ~ Distributed SQL engine
- ~ Creating DataFrame
- ~ DataFrame operations
- ~ DataFrame from csv
- ~ DataFrame from db tables
- ~ DataFrame from hive NoSQL tabel
- ~ DataFrame from json
- ~ DataFrame from RDD
- ~ Different operations on DataFrame
- ~ Filter
- ~ Join
- ~ Group
- ~ Aggregation
- ~ Having
- ~ Where
- ~ User define function(UDF)
- ~ Grouping aggregation
- ~ Multiple grouping
- ~ More aggregation
- ~ Hash aggregation
- ~ Spark SQL vs RDD
- ~ Executing SQL commands and SQL-style functions on a DataFrame
- ~ Using DataFrames instead of RDD's
- ~ Different operations with dataframes with DataFrames
- ~ Word Count with DataFrames
- ~ DataFrames vs RDDs
- ~ Operations on DFs
- ~ Parquet files with Spark Sql Read, Write, Partitioning, Merging schema
- ~ ORC files
- ~ JSON files

## => Spark streaming :

- ~ Basic concepts of Spark Streaming
- ~ Linking
- ~ Initializing Streaming Context
- ~ Discretized Streams (DStreams)
- ~ Input DStreams and Receivers
- ~ Transformations on DStreams
- ~ Output operations on DStreams
- ~ DataFrame and SQL operations
- ~ MLlib operations
- ~ Caching / Persistence
- ~ Checkpointing
- ~ Accumulators, Broadcast Variables, and Checkpoints
- ~ Deploying applications
- ~ Performance tuning
- ~ Writing Producer in Python



- ~ Writing Consumer in Python
- ~ Kafka Integration with Spark Streaming
- ~ Fault-tolerance semantics
- ~ Spark Cassandra

#### => Spark Structure streaming :

- ~ Handling Event-time and Late Data
- ~ API using Datasets and DataFrames
- ~ Creating streaming DataFrames and streaming Datasets
- ~ Input Sources
- ~ Schema inference and partition of streaming DataFrames/Datasets
- ~ Operations on streaming DataFrames/Datasets
- ~ Basic Operations - Selection, Projection, Aggregation
- ~ Window Operations on Event Time
- ~ Handling Late Data and Watermarking
- ~ Types of time windows

#### => Launching on a clusters :

- ~ Spark standalone
- ~ Running Spark on Mesos
- ~ Running Spark on YARN
- ~ The Spark Standalone Web UI

#### => PySpark Installtion :

- ~ Installtion using PyPi
- ~ Pyspark setup in local
- ~ Pyspark setup with anaconda
- ~ Pyspark setup with pycharm

#### => PySpark DataFrame :

- ~ DataFrame creation
- ~ Viewing data
- ~ Accesing data
- ~ Applying a function
- ~ Grouping data
- ~ Object creation
- ~ Missing data
- ~ Grouping
- ~ Plotting

#### => Spark Mlib :

- ~ Overview of Mlib
- ~ What is Machine Learning?
- ~ Supervised learning
- ~ Unsupervised learning
- ~ Basic statistics
- ~ Classification algorithms
- ~ Regression algorithms
- ~ Clustering algorithms
- ~ Collaborative filtering
- ~ Frequent pattern mining
- ~ Featurization
- ~ Pipelines
- ~ Persistence
- ~ Spark ml for ml
- ~ Collect traning data
- ~ Different proccessing technique
- ~ Supervised learning
- ~ Linear regression
- ~ Logistic regression

#### => GraphX :

- ~ Overview
- ~ Graph operations
- ~ Graph builders

#### => Spark configuration, monitoring and tuning :

- ~ Understand components of spark cluster
- ~ configure spark to modify the spark properties, environmental variables, or logging properties
- ~ Monitor Spark using the web UIs, metrics, and external instrumentation

#### => Connecting to Data sources :

- ~ Connecting to local file system
- ~ Understanding storage plugins and workspaces
- ~ Connecting to MySQL
- ~ Connecting to Mongo
- ~ Connecting to Kafka
- ~ Connecting to Hive
- ~ Connecting to HBase
- ~ Querying across data sources

#### => Introduction to Sqoop :

- ~ Sqoop introduction
- ~ How Sqoop works?
- ~ Why we use Sqoop?
- ~ Features of Sqoop

#### => Sqoop Tools :

- ~ Sqoop architecture and working
- ~ Using command aliases
- ~ Controlling the Hadoop installation
- ~ Using generic and specific arguments

~ *Using options files to pass arguments*

=> Sqoop import :

- ~ *Purpose of Sqoop import*
- ~ *Connecting to a database server*
- ~ *Selecting the data to import*

=> Sqoop export :

- ~ *Purpose of Sqoop export*
- ~ *Inserts vs Updates*
- ~ *Exports and Transactions*

=> Sqoop - Job :

- ~ *Create Job*
- ~ *Verify Job*
- ~ *Inspect Job*
- ~ *Execute Job*

=> Setup of Airflow :

- ~ *Components of Airflow*
- ~ *Installing Airflow on mac*
- ~ *Installing Airflow on linux*
- ~ *Installing Airflow on windows*
- ~ *Run Airflow locally*
- ~ *Introduction to the Airflow UI*

=> Core concepts of Airflow :

- ~ *What is DAG?*
- ~ *DAG skeleton*
- ~ *Default arguments*
- ~ *Instantiate a DAG*

=> Loading data to Data Warehouse :

- ~ *Set up*
- ~ *Connections*
- ~ *Load data from storage*
- ~ *Run SQL query*

=> Docker Image for Apache Airflow :

- ~ *Introduction to Docker*
- ~ *Why custom image?*
- ~ *How to build your own image?*
- ~ *Extending vs. customizing the image*

=> Monitoring Airflow :

- ~ *Airflow monitoring with StatsD*
- ~ *Airflow monitoring with Prometheus*
- ~ *Airflow monitoring with Graphana*
- ~ *Error tracking with Sentry*

=> Introduction to Zookeeper :

- ~ *Introduction of Apache Zookeeper*
- ~ *Why we need Zookeeper?*
- ~ *What is Distributed system?*

=> Internal structure :

- ~ *Zookeeper Background*
- ~ *Architecture Diagram*
- ~ *Important Components*

=> Data models and Znodes :

- ~ *Data model and Znode structure*
- ~ *What is Apache Zookeeper Znodes?*
- ~ *Sessions and watches*

=> Installation of Zookeeper :

- ~ *Installation of Apache zookeeper*
- ~ *Configuration of Apache zookeeper*
- ~ *Starting Apache zookeeper server*
- ~ *CLI operations*

=> Role of Zookeeper in kafka :

- ~ *Kafka brokers*
- ~ *Kafka consumers*
- ~ *How does Kafka talk to Zookeeper?*
- ~ *Zookeeper production deployment*

=> Monitoring in Zookeeper - Kafka :

- ~ *Operating system*
- ~ *JMX monitoring*

=> Introduction to Ambari :

- ~ *What is Apache Ambari?*
- ~ *Overview of Apache Ambari*
- ~ *History of Apache Ambari*
- ~ *Goals of Apache Ambari*

=> Core applications of Ambari :

- ~ *Server*
- ~ *Agent*
- ~ *Web UI*
- ~ *Database*

=> Ambari usage :

- ~ Provisioning of Hadoop cluster
- ~ Monitoring of Hadoop cluster
- ~ Management of Hadoop cluster

=> How is Ambari is different from Zookeeper? :

- ~ Basic task
- ~ Nature
- ~ Status maintenance

=> Introduction to Cloud Databricks :

- ~ Introduction about cloud
- ~ Why cloud is important
- ~ Introduction to Databricks

=> Data ingestion - CSV files :

- ~ Data ingestion overview
- ~ What is circuits file
- ~ Requirements
- ~ DataFrame reader
- ~ Select columns
- ~ DataFrame writer

=> Data ingestion - JSON files :

- ~ What is JSON File?
- ~ Write data

=> Introduction to Confluent :

- ~ Overview of Confluent
- ~ Features of Confluent

=> Getting started with Confluent :

- ~ Free trail for Confluent cloud
- ~ Quick start for Apache Kafka using confluent cloud
- ~ Confluent cloud console basics

=> Kafka Clusters :

- ~ Features and limits by cluster type
- ~ Create a cluster with a console
- ~ Expand a dedicated cluster with console
- ~ Shrink a dedicated cluster with console
- ~ Cluster management API overview
- ~ Migrate topics on confluent cloud clusters

=> Manage topics in cloud console :

- ~ Overview
- ~ Create, edit and delete topics
- ~ Use the message browser

=> Introduction to AWS :

- ~ What is AWS?
- ~ AWS solutions for BigData?
- ~ What is Data ingestion?

=> Cloud computing on AWS :

- ~ What is cloud computing?
- ~ Cloud services by AWS
- ~ Cloud Computing Tools on AWS
- ~ Cloding Computing Tools Pricing
- ~ Introduction to AWS S3
- ~ Creating your First S3 bucket
- ~ Uploading an object to your Bucket
- ~ Download an object
- ~ Copy your object to a Folder
- ~ Delete your object and Bucket

=> AWS Storage :

- ~ Introduction to AWS storage
- ~ What is Simple storage Service (S3)?
- ~ How S3 works?
- ~ Use cases of S3
- ~ Storage Hierarchy in S3
- ~ Buckets in S3
- ~ S3 pricing
- ~ Creating and S3 bucket
- ~ Uploading objects to the S3 bucket
- ~ What is Amazon S3 Glacier?
- ~ Glacier Vaults
- ~ Glacier Archives
- ~ Accessing Amazon S3 Glacier

=> AWS Databases :

- ~ Enabling object versioning in the S3 bucket
- ~ Databases on AWS
- ~ Introduction to Amazon Relational Database Service(RDS)
- ~ Features of RDS
- ~ Engine types Configuration
- ~ RDS Pricing
- ~ Creating a SQL Server DB Instance
- ~ Introduction to Amazon Aurora
- ~ Benefits of Amazon Aurora
- ~ Create an Aurora DB cluster
- ~ Introduction to Amazon Dynamo DB

- ~ Components of DynamoDB
- ~ Creating a DynamoDB table.
- ~ Connecting to the DB Instance From Your Machine
- ~ DynamoDB Items and Indexes
- ~ Dynamo Backup and Restore

#### => Collection :

- ~ Collection
- ~ Collection Section Introduction
- ~ Kinesis Data Streams Overview
- ~ Hot shard
- ~ Kinesis Producers
- ~ Kinesis Consumers
- ~ Kinesis Enhanced Fan Out
- ~ Kinesis Scaling
- ~ Kinesis - Handling Duplicate Records
- ~ Kinesis Security
- ~ Kinesis Data Firehose
- ~ CloudWatch Subscription Filters with Kinesis
- ~ SQS Overview
- ~ SQS Hands On
- ~ Kinesis Data Streams vs SQS
- ~ IoT Overview
- ~ IoT Components Deep Dive
- ~ Database Migration Service (DMS)
- ~ Direct Connect
- ~ AWS Snow Family Overview
- ~ AWS Snow Family Hands On
- ~ MSK: Managed Streaming for Apache Kafka
- ~ Kinesis vs MSK

#### => Storage :

- ~ S3 Overview
- ~ S3 Hands On
- ~ S3 Security: Bucket Policy
- ~ S3 Security: Bucket Policy Hands On
- ~ S3 Website Overview
- ~ S3 Website Hands On
- ~ S3 Versioning Overview
- ~ S3 Versioning Hands On
- ~ S3 Server Access Logging
- ~ S3 Server Access Logging Hands On
- ~ S3 Replication Overview
- ~ S3 Replication Hands On
- ~ S3 Storage Classes Overview
- ~ S3 Storage Classes Hands On
- ~ S3 Glacier Vault Lock & S3 Object Lock
- ~ S3 Encryption
- ~ Shared Responsibility Model for S3
- ~ DynamoDB Overview
- ~ DynamoDB RCU & WCU
- ~ DynamoDB Partitions
- ~ DynamoDB APIs
- ~ DynamoDB Indexes: LSI & GSI
- ~ DynamoDB DAX
- ~ DynamoDB Streams
- ~ DynamoDB TTL
- ~ DynamoDB Security
- ~ DynamoDB: Storing Large Objects

#### => Processing :

- ~ Section Introduction: Processing
- ~ Lambda Overview
- ~ Lambda Hands On
- ~ [Exercise] AWS Lambda
- ~ Why Cloud & Big Data on Cloud
- ~ What is Virtual Machine
- ~ On-Premise vs Cloud Setup
- ~ Major Vendors of Hadoop Distribution
- ~ Hdfs vs S3
- ~ Important Instances in AWS
- ~ Spark Basics
- ~ Why spark is difficult
- ~ Overview of EMR
- ~ What is EMR
- ~ Tez vs mapreduce
- ~ Launching an emr cluster
- ~ connecting to your cluster
- ~ Create a tunnel for web ui
- ~ Use Hue to interact with EMR
- ~ Transient vs Long Running Cluster Running
- ~ Copy File From S3 to Local Zeppelin Notebook
- ~ How to Create a VM
- ~ S3 & EBS
- ~ Public ip Vs Private Ip
- ~ Aws Command Line Interface
- ~ AWS Glue
- ~ Introduction to Amazon Redshift
- ~ Redshift Master Slave Architecture

- ~ redshift demo
- ~ redshift specturm
- ~ Redshift Distribution Styles
- ~ Redshift Fault Tolerance
- ~ Redshift Sort Keys

#### => Analysis :

- ~ Section Introduction: Analysis
- ~ Intro to Kinesis Analytics
- ~ Kinesis Analytics Costs; RANDOM\_CUT\_FOREST
- ~ Intro to Opensearch (formerly Elasticsearch)
- ~ Amazon Opensearch Service
- ~ Opensearch Features
- ~ What is Athena
- ~ When do we require Athena What problem Athena Solve How Athena Works
- ~ Athena Pricing
- ~ Athena Practical Demonstration

#### => Visualization :

- ~ The course overview
- ~ big data analytics and aws
- ~ How Quicksight is different than other BI Tools
- ~ BI solution based on quicksight
- ~ how to get started with quicksight
- ~ Performance Your first analysis
- ~ AWS Big data ecosystem
- ~ importing files to quicksight
- ~ importing databases to quicksight
- ~ importing data from saas services to quicksight
- ~ edit existing data sources in quicksight
- ~ Joining datasets
- ~ using functions
- ~ applying filters
- ~ understanding spice layer
- ~ Creating a Quicksight Analysis
- ~ Explore various charting options
- ~ Exploring various Map options
- ~ Exploring various table and other visual options
- ~ Mini project Overview
- ~ Mini Project Architecture
- ~ Data ingestion for mini project
- ~ Reports and dashboards

#### => Introducing Google Cloud Platform :

- ~ Google platform fundamentals overview.
- ~ Google cloud platform Big Data products.

#### => Compute and Storage Fundamentals :

- ~ CPUs on demand (compute engine).
- ~ A global filesystem (cloud storage).
- ~ CloudShell.
- ~ Set up an Ingest-Transform-Publish data processing pipeline.

#### => Data Analytics on the Cloud :

- ~ Stepping-stones to the cloud.
- ~ Cloud SQL: your SQL database on the cloud.
- ~ Importing data into CloudSQL and running queries.
- ~ Spark on Dataproc.
- ~ Machine Learning recommendations with Spark on Dataproc.

#### => Scaling Data Analysis :

- ~ Fast random access.
- ~ Datalab
- ~ BigQuery.

#### => Introduction to cloud :

- ~ Introduction to Cloud Computing
- ~ Cloud models
- ~ Different cloud providers

#### => Regions and Availability Zones :

- ~ Understanding regions and availability zones in Azure
- ~ Creating microsoft Azure account

#### => Resource Hierarchy :

- ~ Understanding resource hierchy
- ~ Demo on resource hierchy
- ~ Resource groups,subscription and managment groups
- ~ Active directory

#### => Introduction to azure cloud computing :

- ~ Azure services overview
- ~ Managed and unmanaged service
- ~ Demo create Azure SQL Database service

#### => Introduction to data engineer profile :

- ~ Introduction to data engineer Technologies
- ~ Data engineer role and responsibility
- ~ Introduction to data engineer technologies

#### => Azure sql database :

- ~ Module Introduction

- ~ Introduction
- ~ Why choosing SQL Server in Azure
- ~ Azure IaaS vs PaaS database offerings
- ~ SQL server PaaS deployment options
- ~ Introduction to Azure SQL server in virtual machine
- ~ SQL Server in Azure virtual machine
- ~ SQL server in Azure virtual machine
- ~ Introduction Azure single database

=> Introduction to SQL :

- ~ Why SQL?
- ~ Application of SQL
- ~ Characteristics of SQL
- ~ Installation guide
- ~ Connection & set up
- ~ Create database
- ~ RENAME database
- ~ Drop database
- ~ SELECT database

=> Data type of SQL :

- ~ Binary datatypes
- ~ Approximate numeric datatype
- ~ Exact numeric datatype
- ~ Character string datatype
- ~ Date and time datatype

=> Introduction to SQL syntax :

- ~ SQL SELECT statement
- ~ SQL WHERE clause
- ~ SQL DISTINCT clause
- ~ SQL AND/OR clause
- ~ SQL IN clause
- ~ SQL LIKE clause
- ~ SQL BETWEEN clause
- ~ SQL ORDER BY clause
- ~ SQL GROUP BY clause
- ~ SQL COUNT clause
- ~ SQL HAVING clause
- ~ SQL CREATE TABLE statement
- ~ SQL DROP TABLE statement
- ~ SQL CREATE INDEX statement
- ~ SQL DROP INDEX statement
- ~ SQL DESC statement
- ~ SQL TRUNCATE TABLE statement
- ~ SQL ALTER TABLE statement
- ~ SQL ALTER TABLE statement(rename)
- ~ SQL INSERT INTO statement
- ~ SQL UPDATE statement
- ~ SQL DELETE statement
- ~ SQL CREATE DATABASE statement
- ~ SQL DROP DATABASE statement
- ~ SQL USE statement
- ~ SQL COMMIT statement
- ~ SQL ROLLBACK statement

=> Operators in SQL :

- ~ Arithmetic operators
- ~ Comparison operators
- ~ Logical operators
- ~ Operators used to negate conditions

=> SQL Query :

- ~ CREATE table
- ~ CREATE table with PRIMARY KEY
- ~ CREATE table with FOREIGN KEY
- ~ DELETE table
- ~ TRUNCATE table
- ~ TEMP table
- ~ RENAME table
- ~ DROP table
- ~ COPY table
- ~ ALTER table
- ~ INSERT query
- ~ UPDATE query
- ~ DELETE query

=> SELECT Query :

- ~ SELECT statement
- ~ SELECT UNIQUE
- ~ SELECT DISTINCT
- ~ SELECT COUNT
- ~ SELECT TOP
- ~ SELECT LAST
- ~ SELECT RANDOM
- ~ SELECT IN
- ~ SELECT RANDOM
- ~ SELECT MULTIPLE
- ~ SELECT DATE
- ~ SELECT SUM

- ~ *SELECT NULL*
- ~ *SELECT group by*

=> SQL Clause :

- ~ *WHERE clause*
- ~ *AND clause*
- ~ *OR clause*
- ~ *WITH clause*
- ~ *AS clause*
- ~ *HAVING clause*
- ~ *Like clause*
- ~ *IS NULL clause*
- ~ *UNION clause*
- ~ *UNION All clause*
- ~ *Top clause*

=> SQL Order By :

- ~ *ORDER BY clause*
- ~ *ORDER BY ASC*
- ~ *ORDER BY DESC*
- ~ *ORDER BY*

=> SQL Constraints :

- ~ *NOT NULL constraint*
- ~ *DEFAULT constraint*
- ~ *UNIQUE constraint*
- ~ *PRIMARY key*
- ~ *FOREIGN key*
- ~ *CHECK constraint*
- ~ *INDEX*
- ~ *Introduction to views*

=> Functions(Aggregate) :

- ~ *Conditional aggregation*
- ~ *List concatenation*
- ~ *SUM*
- ~ *AVG()*
- ~ *Count*
- ~ *Min*
- ~ *Max*

=> SQL Joins :

- ~ *INNER JOIN*
- ~ *LEFT JOIN*
- ~ *RIGHT JOIN*
- ~ *FULL JOIN*
- ~ *SELF JOIN*
- ~ *CARTESIAN JOIN*

=> Views in SQL :

- ~ *Creating view*
- ~ *Creating view from single table*
- ~ *Creating view from multiple tables*
- ~ *Delete view*

=> Window Functions :

- ~ *Setting up a flag if other rows have a common property*
- ~ *Finding "Out-of-Sequence" records using the LAG() function*
- ~ *Getting a running total*
- ~ *Adding the total rows selected to every row*
- ~ *Getting the N most recent rows over multiple grouping*

=> Subqueries :

- ~ *Subquery in FROM clause*
- ~ *Subquery in SELECT clause*
- ~ *Subquery in WHERE clause*
- ~ *Correlated subqueries*
- ~ *Filter query results using query on different table*
- ~ *Subqueries in FROM clause*
- ~ *Subqueries in WHERE clause*

=> Stored Procedures :

- ~ *Create and call a stored procedure*
- ~ *In and out parameters*
- ~ *If, Elseif and Else*
- ~ *Case*
- ~ *While*

=> Triggers :

- ~ *CREATE TRIGGER*
- ~ *Use trigger to manage a "Recycle Bin" for deleted items*

=> AWS Lambda :

- ~ *What is AWS Lambda and Why it is needed?*
- ~ *Features & Limitations of Lambda*
- ~ *Hello world program using Lambda*
- ~ *Auto trigger Lambda Function based on S3 file upload notification*
- ~ *Access other services from Lambda*

=> AWS Secret Manager :

- ~ *Create and Maintain secrets*
- ~ *Accessing credentials from Secret Manager using Boto3*

=> AWS Glue :

- ~ Setting up cluster in Glue
- ~ Properties of Glue
- ~ Creating Catalogs in Glue
- ~ Read partitioned Data
- ~ Bulk and Incremental data processing from S3 in Glue
- ~ Data Processing in Glue
- ~ Glue jobs and Triggers

=> AWS SQS :

- ~ What is SQS?
- ~ Different types of SQS?
- ~ At-Least once and Exactly once delivery via SQS
- ~ Ingesting data to SQS
- ~ Inflight messages
- ~ Consume data from SQS
- ~ Dead Letter Queue

=> AWS Kinesis :

- ~ Ingesting real time data in Kafka Streams
- ~ Consume real time data from Kafka Streams

=> AWS Cloudwatch :

- ~ Cron based triggers
- ~ Event pattern based triggers
- ~ Monitoring & Alerting using Cloudwatch

=> AWS QuickSight :

- ~ Creating business dashboards using Quick sight

=> AWS EC2 :

- ~ Launch a Basic EC2 Instance
- ~ Different Types of instances - Reserved, On-Demand, Spot, Dedicated
- ~ Different configurations of EC2 machines
- ~ Attaching detaching of EBS Volume in EC2
- ~ Practising few commands on EC2

=> AWS IAM :

- ~ The Mechanics behind IAM
- ~ Managing IAM Users
- ~ IAM Administration (Guide) (Listing, Deleting Users & Accounts)
- ~ Managing Permissions for IAM Users
- ~ Changing IAM User Permissions
- ~ Creating and Administering IAM Groups
- ~ Creating and Administering IAM Group Policies
- ~ Assigning Preset and Custom Group Policies

=> AWS S3 :

- ~ Buckets
- ~ Objects
- ~ Upload, Delete Files
- ~ Data Encryptions
- ~ Pricing & Data Limitation on S3
- ~ S3 Versioning
- ~ Version ID
- ~ Bucket policy
- ~ Notifications from S3
- ~ Work with S3 using AWS CLI
- ~ AWS Lambda
- ~ What is AWS Lambda and Why it is needed?
- ~ Features & Limitations of Lambda
- ~ Hello world program using Lambda
- ~ Auto trigger Lambda Function based on S3 file upload notification
- ~ Access other services from Lambda
- ~ AWS Secret Manager
- ~ Create and Maintain secrets
- ~ Accessing credentials from Secret Manager using Boto3

=> AWS EMR :

- ~ Setting up EMR Cluster
- ~ Install Spark, Hive, Hadoop
- ~ Resource types in EMR cluster
- ~ Data Processing on EMR Cluster
- ~ AWS Glue
- ~ Setting up cluster in Glue
- ~ Properties of Glue
- ~ Creating Catalogs in Glue
- ~ Read partitioned Data
- ~ Bulk and Incremental data processing from S3 in Glue
- ~ Data Processing in Glue
- ~ Glue jobs and Triggers

=> AWS SNS :

- ~ What is SNS?
- ~ How SNS works?
- ~ Creating SNS Topics and subscribing
- ~ Different types of subscribers
- ~ Sending notifications via SNS
- ~ AWS SQS
- ~ What is SQS?
- ~ Different types of SQS?



- ~ At-Least once and Exactly once delivery via SQS
- ~ Ingesting data to SQS
- ~ Inflight messages
- ~ Consume data from SQS
- ~ Dead Letter Queue

=> AWS DMS :

- ~ What is DMS?
- ~ Capturing CDC event in DMS where Database as a source
- ~ Capture CDC events and sending it to downstream systems
- ~ AWS Kinesis
- ~ Creating Kinesis Streams
- ~ Ingesting real time data in Kafka Streams
- ~ Consume real time data from Kafka Streams

=> AWS RDS :

- ~ MySQL Database using AWS RDS
- ~ Scalability & Limitations of AWS RDS
- ~ Creating tables and loading data in AWS RDS
- ~ Querying data from RDS
- ~ AWS Athena
- ~ What is serverless database services
- ~ Athena vs RDS
- ~ Table metadata in Athena for the data residing in S3
- ~ Creating table for S3 data
- ~ Querying S3 data using Athena

=> AWS Redshift :

- ~ What is Data warehousing services?
- ~ Architecture of Redshift
- ~ Resources types in Redshift Cluster
- ~ Creating tables in Redshift
- ~ Internal & External tables
- ~ Partitioning, Sort Keys, Column compression
- ~ Querying data in Redshift
- ~ Views & Materialized views in Redshift

=> AWS Dynamo :

- ~ Architecture of DynamoDB
- ~ Creating tables and Ingesting data into DynamoDB table
- ~ Querying data from DynamoDB
- ~ AWS Cloudwatch
- ~ Cron based triggers
- ~ Event pattern based triggers
- ~ Monitoring & Alerting using Cloudwatch
- ~ AWS QuickSight
- ~ Creating business dashboards using Quick sight

### Project details :-

=> End to End Project :

- ~ Big data end to end project with deployment

# Crack the Tech Interview

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/Crack-the-Tech-Interview>

## Course Description :-

Algorithmic programming techniques are a must-have skill. Learn Algorithms through programming and puzzle solving to advance your Software Engineering or Data Science career. Then, implement each algorithmic problem in this program to ace coding interviews.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Non Tech Round Preparation
- => Array interview problems and solutions
- => String interview problems and solutions
- => Recursion interview problems and solutions
- => Linked list interview problems and solutions
- => Math interview problems and solutions
- => Stack and Queue interview problems and solutions
- => Sorting interview problems and solutions
- => Trees interview problems and solutions
- => Graphs interview problems and solutions
- => Dynamic programming interview problems and solutions

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Preparing for the interview :

- ~ FAQ before taking this course*
- ~ How to take this course*

=> Non Tech Round Preparation :

- ~ Are you ready for interviews*
- ~ Your resume needs more work*
- ~ 8 point resume check list*
- ~ Handle experience section*
- ~ FAANG interview process*
- ~ How to find jobs*
- ~ 3 pillars of answers*
- ~ Tell me about yourself*
- ~ Why our company*
- ~ Recent project problem*
- ~ Tell me your weakness*

=> Array interview problems and solutions :

- ~ Binary search - How to explain in interview*
- ~ Binary search - recursion explanation*
- ~ Binary search - iterative explanation*
- ~ Rotation of array - expected explanation*
- ~ Pivot problem code*

- ~ Search in rotated array - Theory
- ~ Search in rotated array - Code
- ~ Find by comparison
- ~ Find by comparison - crafting code
- ~ Target Triplet
- ~ Target Triplet Code expected solution
- ~ Move to 1 side problems
- ~ Move to 1 side code
- ~ Sell at max profit problem
- ~ Sell at max profit code

=> String interview problems and solutions :

- ~ Word in a sentence problem
- ~ Word in a sentence problem Code
- ~ Inplace duplicates
- ~ Inplace duplicates code
- ~ Longest Substring
- ~ Longest Substring Code
- ~ Palindrome makes and breaks interviews
- ~ Palindrome makes and breaks interviews CODE

=> Recursion interview problems and solutions :

- ~ PreReq for recursion
- ~ Classic fibonacci problem but with diary
- ~ Classic fibonacci problem but with diary code
- ~ Popular subset problem
- ~ Popular subset problem CODE
- ~ Decimal to binary for Round 1
- ~ Decimal to binary for Round 1 Code
- ~ NearBy Duplicates
- ~ NearBy Duplicates Code
- ~ Pascal nth row
- ~ Pascal nth row Code

=> Linked list interview problems and solutions :

- ~ Approach for linked list and head
- ~ Approach for linked list and head Code
- ~ Insert in doubly linked list
- ~ Insert in doubly linked list Code
- ~ Tail insertion in doubly linked list
- ~ Tail insertion in doubly linked list Code
- ~ Deleting a val in doubly linked list
- ~ Deleting a val in doubly linked list Code
- ~ Reverse a doubly linkedlist with traveller
- ~ Reverse a doubly linkedlist with traveller Code
- ~ Floyds loop detection
- ~ Floyds loop detection Code
- ~ Merge 2 linked list
- ~ Merge 2 linked list code

=> Math interview problems and solutions :

- ~ Not counted in
- ~ Permutation explanation on White board
- ~ Permutation explanation code
- ~ kth Permutation theory explained
- ~ kth Permutation code
- ~ Bit manipulation
- ~ Bit manipulation Code

=> Stack and Queue interview problems and solutions :

- ~ Stack using queue
- ~ Stack using queue Code
- ~ Stack using queue - Approach 2
- ~ Stack using queue - Approach 2 Code
- ~ Queue using stack
- ~ Queue using stack Code
- ~ Queue using stack - approach 2
- ~ Queue using stack - approach 2 Code
- ~ Stock Spanning
- ~ Stock Spanning Code
- ~ Valid brackets
- ~ Valid brackets Code

=> Sorting interview problems and solutions :

- ~ Bubble Sort
- ~ Bubble Sort Code
- ~ Selection Sort
- ~ Selection Sort Code
- ~ Insertion sort
- ~ Insertion sort Code
- ~ Merge Sort
- ~ Merge Sort Code
- ~ Quick Sort
- ~ Quick Sort Code
- ~ Tea Coffee and Milk problem

=> Trees interview problems and solutions :

- ~ A quick word before problems
- ~ Same tree problem
- ~ Same tree problem Code
- ~ Killer pays road tax problem

- ~ In order iterator
- ~ In order iterator Code
- ~ Flip or Inverse a Binary tree
- ~ Flip or Inverse a Binary tree Code
- ~ Level order of tree
- ~ Level order of tree Code
- ~ Boundary of a tree
- ~ Boundary of a tree Code

=> Graphs interview problems and solutions :

- ~ Basics of graph theory
- ~ Clone a graph or copy
- ~ Clone a graph or copy Code
- ~ DFS and Cycle detection with University course problem
- ~ DFS and Cycle detection with University course problem CODE
- ~ Breadth first search for graphs
- ~ Breadth first search for graphs CODE
- ~ Island problem
- ~ Island problem CODE

=> Dynamic programming interview problems and solutions :

- ~ Foundataion of dynamic programming
- ~ 0Knapsack - Coke 2C pepsi 2C redbull
- ~ 0Knapsack - Coke 2C pepsi 2C redbull CODE
- ~ Largest sum of subset
- ~ Largest sum of subset Code
- ~ Largest sum - Difficult
- ~ Largest sum - Difficult CODE
- ~ Coin change problem
- ~ Coin change problem CODE
- ~ Min path to reach target
- ~ Min path to reach target CODE

# Pro Backend Developer

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Pro-Backend-Developer>

## Course Description :-

This course is titled pro for a reason. In this practical hands-on course, you will learn how to build complex backend applications that can be used for any web or mobile application. Your REST API will be in production with docs, social logins, images, authentications, mail and, much more. This is a true pro backend course.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => MongoDB
- => Heroku Cloud
- => Swagger
- => Authentication
- => File, image and form handling
- => MORGAN and razorpay
- => Configs and imports
- => Controllers and routes

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started :

- ~ Goal of this course and instructions
- ~ Tools for backend developer
- ~ MongoDB MAC install
- ~ MongoDB WIN install
- ~ MongoDB in cloud - Atlas
- ~ Mongo GUI - compass

=> Take it up to Heroku - Production :

- ~ Things you need to deploy on Heroku
- ~ Plan your application
- ~ Types of web request
- ~ Framework - Express, Koa, Hapi
- ~ Starting with package JSON file
- ~ Your first express app
- ~ Request Response and Status code
- ~ All social routes
- ~ Handle the date situation
- ~ Parameters and bugs in route
- ~ Pushing app to HEROKU
- ~ Debug social app in production

=> Swagger Docs :

- ~ What is swagger and api docs
- ~ Nodemon ext and YAML docs

- ~ Authentication token for swagger docs
- ~ Docs for HTTP methods swagger
- ~ A new documentation centric project
- ~ Setup information - swagger
- ~ Authentication and Authorization - swagger
- ~ String based GET request - swagger
- ~ handling objects - swagger
- ~ handling array in Swagger docs
- ~ Sending data in URL - swagger
- ~ managing request body in swagger
- ~ handle url query in swagger
- ~ handling images in swagger
- ~ handling header tokens in swagger

=> Authentication :

- ~ What we have done till section 3 - backend
- ~ Hiding secrets in backend
- ~ Picking up a database for backend
- ~ Why we need mongoose - ODM
- ~ Pro db modeling tools
- ~ Creating model for auth system
- ~ Creating basic structure for auth system
- ~ Creating user schema and dotenv
- ~ Registering a user in auth system
- ~ Database connection in auth system
- ~ What is a middleware
- ~ Handling password situation
- ~ What is JWT and creating token
- ~ Register route in auth app
- ~ Login flow for auth app
- ~ Web vs Mobile
- ~ Writing custom middleware
- ~ Setting up secure cookies

=> File, image and form handling :

- ~ Why people face issue in image upload
- ~ Cloudinary and EJS
- ~ How GET works and postman issues
- ~ Using template engines
- ~ Biggest confusion in front end forms
- ~ Handling images in forms
- ~ Handling images in forms part 2
- ~ upload image to cloudinary or other providers
- ~ Handling multiple files and uploading them

=> Theory and Razorpay :

- ~ File structure for production app
- ~ Getting a logger - MORGAN
- ~ Error handler and Promises
- ~ Sending emails using nodemailer
- ~ Why mongoose docs are important
- ~ Razorpay project
- ~ Razorpay front end integration

=> Big Ecommerce app starts :

- ~ Project requirement
- ~ User modeling and file structure
- ~ Product model discussion
- ~ Order Model discussion
- ~ How forgot password feature work
- ~ Functions in user model and hooks

=> Basic Config and imports :

- ~ Getting files and folders ready
- ~ Preparing basic express app
- ~ Routes and controllers in dummy
- ~ Injecting docs and middleware
- ~ Custom error handlers
- ~ The big Promise

=> User model and signup :

- ~ Creating a user model and validator
- ~ password encryption and mongoose prototypes
- ~ Validating the password
- ~ creating JWT tokens
- ~ forgot password and crypto hashing
- ~ User routes and postman
- ~ Signup a user and cookies
- ~ Database connection
- ~ Testing the user signup with postman
- ~ Handling image upload
- ~ Testing photo upload and user signup
- ~ yes, we know about postman files

=> User controllers and routes :

- ~ Login route and controller
- ~ logout controller and route
- ~ Send email from node
- ~ Forgot password controller
- ~ Reset password controller and routes
- ~ Middleware - injecting information

- ~ User dashboard controller and routes
- ~ Update the password for a user
- ~ Updating the user profile
- ~ User, admin, manager and more roles
- ~ Manager only routes
- ~ Admin get a single user
- ~ Admin can update any user
- ~ Admin can delete a user now

#### => Working on Product Model :

- ~ Product middleware setup for routes
- ~ Product Model and refs
- ~ A long talk on URL replace and mongo operators
- ~ Creating a product
- ~ Where clause in search
- ~ Where clause Pager
- ~ Aggregation filter in Where Clause
- ~ Get all products with WHERE and pager
- ~ Debugging and testing of product add and get

#### => More routes in Products :

- ~ Single product route
- ~ Update the product with photos
- ~ Delete a product and minor bug
- ~ Testing and debugging
- ~ Add a review
- ~ Delete a review and requested routes
- ~ Configure routes for reviews

#### => Razorpay and Stripe :

- ~ Stripe Docs
- ~ Stripe controllers
- ~ Razorpay payments and order
- ~ Setup payment routes

#### => Processing Orders :

- ~ Order model in action
- ~ Creating an order and BSON
- ~ Testing create order and routes
- ~ Populate fields in order
- ~ Order of routes is important
- ~ Updating the stock
- ~ Delete order and push to git
- ~ Pushing code to production server

#### => OAuth and Social Logins :

- ~ Social login foundation and demo app
- ~ Consent screen and API keys
- ~ Why passport.js
- ~ Package installation
- ~ Home routes and EJS
- ~ Preparing routes for login
- ~ Showing consent screen of google
- ~ Getting information and email from google
- ~ Moving google data to database
- ~ Serialize and deserialize user
- ~ Protect the Home

# Data Science Masters

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Data-Science-Masters>

## Course Description :-

This is a data science masters course where you will learn all the stack required to work in data science, data analytics and big data industry including ML ops and cloud infrastructure .

## Course Features :-

- => Full stack Data Science masters certification
- => 56 + hands-on industry real-time projects.
- => 500 hours of recorded classes.
- => Lifetime Dashboard access
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => ML ops
- => Cloud
- => Data structure and algorithm
- => Architecture
- => Domain wise project
- => Databases
- => Negotiations skills
- => Mock interview
- => Interview preparation
- => Resume building after every module

## Requirements :-

- => Dedication
- => Computer with i3 and above configuration

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.



=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course Introduction :

- ~ course overview and dashboard description
- ~ Introduction of data science and its application in day to day life
- ~ Programming language overview
- ~ Installation (tools: sublime, vscode, pycharm, anaconda, atom, jupyter notebook, kite)
- ~ Virtual environment
- ~ Why python

=> Python Basic :

- ~ Introduction of python and comparison with other programming language Preview
- ~ Installation of anaconda distribution and other python ide
- ~ Python objects, number & Booleans, strings
- ~ Container objects, mutability of objects
- ~ Operators - arithmetic, bitwise, comparison and assignment operators, operators precedence and associativity
- ~ Conditions (if else, if-elif-else), loops (while, for)
- ~ Break and continue statement and range function

=> String Objects :

- ~ basic data structure in python
- ~ String object basics
- ~ String inbuilt methods
- ~ Splitting and joining strings
- ~ String format functions

=> List Object Basics :

- ~ List methods
- ~ List as stack and queues
- ~ List comprehensions

=> Tuples, Sets, Dictionaries & its Function :

- ~ Dictionary object methods
- ~ Dictionary comprehensions
- ~ Dictionary view objects
- ~ Functions basics, parameter passing, iterators
- ~ Generator functions
- ~ Lambda functions
- ~ Map, reduce, filter functions

=> Memory Management :

- ~ Multithreading
- ~ Multiprocessing

=> OOPs Concepts :

- ~ oops basic concepts.
- ~ Creating classes
- ~ Pillars of oops
- ~ Inheritance
- ~ Polymorphism
- ~ Encapsulation
- ~ Abstraction
- ~ Decorator
- ~ Class methods and static methods
- ~ Special (magic/dunder) methods
- ~ Property decorators - getters, setters, and deletes

=> Files :

- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other file methods
- ~ Logging, debugger
- ~ Modules and import statements

=> Exception Handling and Difference between Exception and Error :

- ~ Exceptions handling with try-except
- ~ Custom exception handling
- ~ List of general use exception
- ~ Best practice exception handling

=> GUI Framework :

- ~ What is desktop and standalone application
- ~ Use of desktop app
- ~ Examples of desktop app
- ~ Tinker
- ~ Kivy

=> Database :

- ~ SQLite
- ~ MySQL
- ~ Mongo dB
- ~ NoSQL - Cassandra

=> Web API :

- ~ What is web API
- ~ Difference b/w API and web API
- ~ Rest and soap architecture

~ *Restful services*

## => Flask :

~ *Flask introduction*  
~ *Flask application*  
~ *Open link flask*  
~ *App routing flask*  
~ *Url building flask*  
~ *Http methods flask*  
~ *Templates flask*  
~ *Flask project: food app*  
~ *Postman*  
~ *Swagger*

## => Django :

~ *Django introduction*  
~ *Django project: weather app*  
~ *Django project: memes generator*  
~ *Django project: blog app*  
~ *Django project in cloud*

## => Stream Lit :

~ *Stream lit introduction*  
~ *Stream lit project structure*  
~ *Stream lit project in cloud*

## => Pandas Basic :

~ *Python pandas - series*  
~ *Python pandas data frame*  
~ *Python pandas panel*  
~ *Python pandas - basic functionality*  
~ *Reading data from different file system*

## => Pandas Advance :

~ *Python pandas re indexing python*  
~ *Pandas iteration*  
~ *Python pandas sorting.*  
~ *Working with text data options & customization*  
~ *Indexing & selecting*  
~ *Data statistical functions*  
~ *Python pandas - window functions*  
~ *Python pandas - date functionality*  
~ *Python pandas time delta*  
~ *Python pandas - categorical data*  
~ *Python pandas visualization*  
~ *Python pandas - iotools*

## => Dask :

~ *Dask Array*  
~ *Dask Bag*  
~ *Dask DataFrame*  
~ *Dask Delayed*  
~ *Dask Futures*  
~ *Dask API*  
~ *Dask SCHEDULING*  
~ *Dask Understanding Performance*  
~ *Dask Visualize task graphs*  
~ *Dask Diagnostics (local)*  
~ *Dask Diagnostics (distributed)*  
~ *Dask Debugging*  
~ *Dask Ordering*

## => Python Numpy :

~ *Numpy - ND array object.*  
~ *Numpy - data types.*  
~ *Numpy - array attributes.*  
~ *Numpy - array creation routines.*  
~ *Numpy - array from existing.*  
~ *Data array from numerical ranges.*  
~ *Numpy - indexing & slicing.*  
~ *Numpy advanced indexing.*  
~ *Numpy broadcasting.*  
~ *Numpy - iterating over array.*  
~ *Numpy - array manipulation.*  
~ *Numpy - binary operators.*  
~ *Numpy - string functions.*  
~ *Numpy - mathematical functions.*  
~ *Numpy - arithmetic operations.*  
~ *Numpy - statistical functions.*  
~ *Sort, search & counting functions.*  
~ *Numpy - byte swapping.*  
~ *Numpy - copies & views.*  
~ *Numpy - matrix library.*  
~ *Numpy - linear algebra*

## => Visualization :

~ *Matplotlib*  
~ *Seaborn*  
~ *Cufflinks*  
~ *Plotly*  
~ *Bokeh*

## => Statistics Basic :

- ~ *Introduction to basic statistics terms*
- ~ *Types of statistics*
- ~ *Types of data*
- ~ *Levels of measurement*
- ~ *Measures of central tendency*
- ~ *Measures of dispersion*
- ~ *Random variables*
- ~ *Set*
- ~ *Skewness*
- ~ *Covariance and correlation*

## => Probability Distribution Function :

- ~ *Probability density/distribution function*
- ~ *Types of the probability distribution*
- ~ *Binomial distribution*
- ~ *Poisson distribution*
- ~ *Normal distribution (Gaussian distribution)*
- ~ *Probability density function and mass function*
- ~ *Cumulative density function*
- ~ *Examples of normal distribution*
- ~ *Bernoulli distribution*
- ~ *Uniform distribution*
- ~ *Z stats*
- ~ *Central limit theorem*
- ~ *Estimation*

## => Statistics Advance :

- ~ *a Hypothesis*
- ~ *Hypothesis testings mechanism*
- ~ *P-value*
- ~ *T-stats*
- ~ *Student t distribution*
- ~ *T-stats vs. Z-stats: overview*
- ~ *When to use a t-tests vs. Z-tests*
- ~ *Type 1 & type 2 error*
- ~ *Bayes statistics (Bayes theorem)*
- ~ *Confidence interval(ci)*
- ~ *Confidence intervals and the margin of error*
- ~ *Interpreting confidence levels and confidence intervals*
- ~ *Chi-square test*
- ~ *Chi-square distribution using python*
- ~ *Chi-square for goodness of fit test*
- ~ *When to use which statistical distribution?*
- ~ *Analysis of variance (anova)*
- ~ *Assumptions to use anova*
- ~ *Anova three type*
- ~ *Partitioning of variance in the anova*
- ~ *Calculating using python*
- ~ *F-distribution*
- ~ *F-test (variance ratio test)*
- ~ *Determining the values of f*
- ~ *F distribution using python*

## => Linear Algebra :

- ~ *linear algebra*
- ~ *Vector*
- ~ *Scaler*
- ~ *Matrix*
- ~ *Matrix operations and manipulations*
- ~ *Dot product of two vectors*
- ~ *Transpose of a matrix*
- ~ *Linear independence of vectors*
- ~ *Rank of a matrix*
- ~ *Identity matrix or operator*
- ~ *Determinant of a matrix*
- ~ *Inverse of a matrix*
- ~ *Norm of a vector*
- ~ *Eigenvalues and eigenvectors*
- ~ *Calculus*

## => Solving Stats Problem with Python

## => Stats Problem Implementation with Spicy

## => Introduction to Machine Learning :

- ~ *AI vs ml vs dl vs ds*
- ~ *Supervised, unsupervised, semi-supervised, reinforcement learning*
- ~ *Train, test, validation split*
- ~ *Performance*
- ~ *Overfitting, under fitting*
- ~ *Bias vs variance*

## => Feature Engineering :

- ~ *Handling missing data*
- ~ *Handling imbalanced data*
- ~ *Up-sampling*
- ~ *Down-sampling*
- ~ *Smote*
- ~ *Data interpolation*

- ~ Handling outliers
- ~ Filter method
- ~ Wrapper method
- ~ Embedded methods
- ~ Feature scaling
- ~ Standardization
- ~ Mean normalization
- ~ Min-max scaling
- ~ Unit vector
- ~ Feature extraction
- ~ Pca (principle component analysis)
- ~ Data encoding
- ~ Nominal encoding
- ~ One hot encoding
- ~ One hot encoding with multiple categories
- ~ Mean encoding
- ~ Ordinal encoding
- ~ Label encoding
- ~ Target guided ordinal encoding
- ~ Covariance
- ~ Correlation check
- ~ Pearson correlation coefficient
- ~ Spearmans rank correlation
- ~ Vif

=> Feature Selection :

- ~ Feature selection
- ~ Recursive feature elimination
- ~ Backward elimination
- ~ Forward elimination

=> Exploratory Data Analysis :

- ~ Feature engineering and selection.
- ~ Analyzing bike sharing trends.
- ~ Analyzing movie reviews sentiment.
- ~ Customer segmentation and effective cross selling.
- ~ Analyzing wine types and quality.
- ~ Analyzing music trends and recommendations.
- ~ Forecasting stock and commodity prices

=> Regression :

- ~ Linear regression
- ~ Gradient descent
- ~ Multiple linear regression
- ~ Polynomial regression
- ~ R square and adjusted r square
- ~ Rmse , mse, mae comparison
- ~ Regularized linear models
- ~ Ridge regression
- ~ Lasso regression
- ~ Elastic net
- ~ Complete end-to-end project with deployment on cloud and ui

=> Logistics Regression :

- ~ Logistics regression in-depth intuition
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Hyper parameter tuning
- ~ Grid search cv
- ~ Randomize search cv
- ~ Data leakage
- ~ Confusion matrix
- ~ Precision, recall, f1 score , roc, auc
- ~ Best metric selection
- ~ Multiclass classification in lr
- ~ Complete end-to-end project with deployment in multi cloud platform

=> Decision Tree :

- ~ Decision tree classifier
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Confusion matrix
- ~ Precision, recall, f1 score , roc, auc
- ~ Best metric selection
- ~ Decision tree repressor
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Performance metrics
- ~ Complete end-to-end project with deployment in multi cloud platform

=> Support Vector Machines :

- ~ Linear svm classification
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Soft margin classification
- ~ Nonlinear svm classification
- ~ Polynomial kernel
- ~ Gaussian, rbf kernel
- ~ Data leakage
- ~ Confusion matrix

- ~ *precision, recall, f1 score, roc, auc*
- ~ *Best metric selection*
- ~ *Svm regression*
- ~ *In-depth mathematical intuition*
- ~ *In-depth geometrical intuition*
- ~ *Complete end-to-end project with deployment*

#### => Naive Bayes :

- ~ *Bayes theorem*
- ~ *Multinomial naive Bayes*
- ~ *Gaussian naive Bayes*
- ~ *Various type of Bayes theorem and its intuition*
- ~ *Confusion matrix*
- ~ *precision, recall, f1 score, roc, auc*
- ~ *Best metric selection*
- ~ *Complete end-to-end project with deployment*

#### => Ensemble Technique and its Types :

- ~ *Definition of ensemble techniques*
- ~ *Bagging technique*
- ~ *Bootstrap aggregation*
- ~ *Random forest (bagging technique)*
- ~ *Random forest regressor*
- ~ *Random forest classifier*
- ~ *Complete end-to-end project with deployment*

#### => Boosting :

- ~ *Boosting technique*
- ~ *Ada boost*
- ~ *Gradient boost*
- ~ *Xgboost*
- ~ *Complete end-to-end project with deployment*

#### => Stacking :

- ~ *Stacking technique*
- ~ *Complete end-to-end project with deployment*

#### => KNN :

- ~ *Knn classifier*
- ~ *Knn regressor*
- ~ *Variants of knn*
- ~ *Brute force knn*
- ~ *K-dimension tree*
- ~ *Ball tree*
- ~ *Complete end-to-end project with deployment*

#### => Dimensionality Reduction :

- ~ *The curse of dimensionality*
- ~ *Dimensionality reduction technique*
- ~ *Pca (principle component analysis)*
- ~ *Mathematics behind pca*
- ~ *Scree plots*
- ~ *Eigen-decomposition approach*

#### => Clustering :

- ~ *Clustering and their types*
- ~ *K-means clustering*
- ~ *K-means++*
- ~ *Batch k-means*
- ~ *Hierarchical clustering*
- ~ *Dbscan*
- ~ *Evaluation of clustering*
- ~ *Homogeneity, completeness and v-measure*
- ~ *Silhouette coefficient*
- ~ *Davies-bouldin index*
- ~ *Contingency matrix*
- ~ *Pair confusion matrix*
- ~ *Extrinsic measure*
- ~ *Intrinsic measure*
- ~ *Complete end-to-end project with deployment*

#### => Anomaly Detection :

- ~ *Anomaly detection types*
- ~ *Anomaly detection applications*
- ~ *Isolation forest anomaly detection algorithm*
- ~ *Isolation forest anomaly detection algorithm*
- ~ *Support vector machine anomaly detection algorithm*
- ~ *Dbscan algorithm for anomaly detection*
- ~ *Complete end-to-end project with deployment*

#### => Time-Series :

- ~ *What is a time series?*
- ~ *Old techniques*
- ~ *Arima*
- ~ *Acf and pacf*
- ~ *Time-dependent seasonal components.*
- ~ *Autoregressive (ar),*
- ~ *Moving average (ma) and mixed arma- modeler.*
- ~ *The random walk model.*
- ~ *Box-jenkins methodology.*
- ~ *Forecasts with arima and var models.*

- ~ *Dynamic models with time-shifted explanatory variables.*
- ~ *The koyck transformation.*
- ~ *Partial adjustment and adaptive expectation models.*
- ~ *Granger's causality tests.*
- ~ *Stationarity, unit roots and integration*
- ~ *Time series model performance*
- ~ *Various approach to solve time series problem*
- ~ *Complete end-to-end project with deployment*
- ~ *Prediction of nifty stock price and deployment*

=> NLP Basic :

- ~ *Tokenization*
- ~ *Pos tags and chunking*
- ~ *Stop words*
- ~ *Stemming and lemmatization*
- ~ *Named entity recognition (ner)*
- ~ *Word vectorization (word embedding)*
- ~ *Tfidf*
- ~ *Complete end-to-end project with deployment*

=> Machine Learning Pipeline :

- ~ *Aws segmaker*
- ~ *Aure ml studio*
- ~ *ML flow*
- ~ *Kube flow*

=> Model Retraining Approach

=> Auto ML :

- ~ *H2o*
- ~ *Pycaret*
- ~ *Auto sklearn*
- ~ *Auto time series*
- ~ *Auto viml*
- ~ *Auto gluon*
- ~ *Auto viz*
- ~ *Tpot*
- ~ *Auto neuro*

=> Neural Network A Simple perception :

- ~ *Detail mathematical explanation*
- ~ *Neural network overview and its use case.*
- ~ *Various neural network architect overview.*
- ~ *Use case of neural network in nlp and computer vision.*
- ~ *Activation function -all name*
- ~ *Multilayer network.*
- ~ *Loss functions. - all 10*
- ~ *The learning mechanism.*
- ~ *Optimizers. - all 10*
- ~ *Forward and backward propagation.*
- ~ *Weight initialization technique*
- ~ *Vanishing gradient problem*
- ~ *Exploding gradient problem*
- ~ *Visualization of nn*

=> Hardware Setup - GPU :

- ~ *Gpu introduction.*
- ~ *Various type of gpu configuration.*
- ~ *Gpu provider and its pricing.*
- ~ *Paper space gpu setup.*
- ~ *Running model in gpu*

=> Tensor Flow Installation Environment Setup For Deep Learning :

- ~ *Colab pro setup*
- ~ *Tensor flow installation 2.0 .*
- ~ *Tensor flow installation 1.6 with virtual environment.*
- ~ *Tensor flow 2.0 function.*
- ~ *Tensor flow 2.0 neural network creation.*
- ~ *Tensor flow 1.6 functions.*
- ~ *Tensor flow 1.6 neural network and its functions.*
- ~ *Keras introduction.*
- ~ *Keras in-depth with neural network creation.*
- ~ *Mini project in tensorflow.*
- ~ *Tensorspace*
- ~ *Tensorboard integration*
- ~ *Tensorflow playground*
- ~ *Netron*

=> Pytorch :

- ~ *pytorch installation.*
- ~ *Pytorch functional overview.*
- ~ *Pytorch neural network creation.*

=> Mxnet :

- ~ *Mxnet installation*
- ~ *Mxnet in depth function overview*
- ~ *Mxnet model creation and training*

=> Keras Tuner :

- ~ *Keras tuner installation and overview*
- ~ *Finding best parameter from keras tuner*
- ~ *Keras tuner application across various neural network*

## => CNN Overview :

- ~ *Cnn definition*
- ~ *Various cnn based architecture*
- ~ *Explanation end to end cnn network*
- ~ *Cnn explainer*
- ~ *Training cnn*
- ~ *Deployment in azure cloud*
- ~ *Performance tuning of cnn network*

## => Advance Computer Vision - Part 1 :

- ~ *Various cnn architecture with research paper and mathematics*
- ~ *Lenet-5 variants with research paper and practical*
- ~ *Alexnet variants with research paper and practical*
- ~ *Googlenet variants with research paper and practical*
- ~ *Transfer learning*
- ~ *Vggnet variants with research paper and practical*
- ~ *Resnet variants with research paper and practical*
- ~ *Inception net variants with research paper and practical*
- ~ *Darknet variants with research paper and practical*

## => Advance Computer Vision - Part 2 :

- ~ *Object detection in-depth*
- ~ *Transfer learning*
- ~ *Rcnn with research paper and practical*
- ~ *Fast rcnn with research paper and practical*
- ~ *Faster r cnn with research paper and practical*
- ~ *Ssd with research paper and practical*
- ~ *Ssd lite with research paper and practical*

## => Training of Custom Object Detection :

- ~ *Tfod introduction*
- ~ *Environment setup with tfod*
- ~ *Gpu vs tpu vs cpu*
- ~ *Various gpu comparison*

## => Advance Computer Vision - Part 3 :

- ~ *Yolo v1 with research paper and practical*
- ~ *Yolo v2 with research paper and practical*
- ~ *Yolo v3 with research paper and practical*
- ~ *Yolo v4 with research paper and practical*
- ~ *Yolo v5 with research paper and practical*
- ~ *Retina net*
- ~ *Face net*
- ~ *Detectron2 with practical and live testing*

## => Object Segmentation :

- ~ *Semantic segmentation*
- ~ *Panoptic segmentation*
- ~ *Masked rcnn*
- ~ *Practical with detectron*
- ~ *Practical with tfod*

## => Object Tracking :

- ~ *Detail of object tracking*
- ~ *Kalman filtering*
- ~ *Sort*
- ~ *Deep sort*
- ~ *Object tracking live project with live camera testing*

## => OCR :

- ~ *Introduction to ocr*
- ~ *Various framework and api for ocr*
- ~ *Practical implementation of ocr*
- ~ *Live project deployment for bill parsing*

## => Image Captioning :

- ~ *Image captioning overview*
- ~ *Image captioning project with deployment*

## => Tensorflow JS :

- ~ *Tensorflow js overview*
- ~ *Tfjs implementation*

## => Model Conversion :

- ~ *Tfjs*
- ~ *Tflite*
- ~ *Tfrt*
- ~ *Torch to tf model*
- ~ *Mxnet to tf conversion*

## => Advance NLP with Deep Learning :

- ~ *Overview computational linguistic.*
- ~ *History of nlp.*
- ~ *Why nlp*
- ~ *Use of nlp*

## => Text Processing Importing Text :

- ~ *Web scrapping.*
- ~ *Text processing*
- ~ *Understanding regex.*
- ~ *Text normalization*
- ~ *Word count.*

- ~ Frequency distribution.
- ~ Text annotation.
- ~ Use of annotator.
- ~ String tokenization
- ~ Annotator creation.
- ~ Sentence processing.
- ~ Lemmatization in text processing
- ~ Pos
- ~ Named entity recognition
- ~ Dependency parsing in text.
- ~ Sentimental analysis

=> Spacy :

- ~ Spacy overview.
- ~ Spacy function
- ~ Spacy function implementation in text processing.
- ~ Pos tagging, challenges and accuracy.
- ~ Entities and named entry recognition
- ~ Interpolation, language models
- ~ Nltk
- ~ Text blob
- ~ Stanford nlp

=> RNN :

- ~ Recurrent neural networks.
- ~ Long short term memory (lstm)
- ~ Bi lstm.
- ~ Stacked lstm
- ~ Gru implementation.
- ~ Building a story writer using character level rnn.

=> Word Embedding :

- ~ Word embedding
- ~ Co-occurrence vectors
- ~ Word2vec
- ~ Doc2vec

=> Attention Based Model :

- ~ Seq 2 seq.
- ~ Encoders and decoders.
- ~ Attention mechanism.
- ~ Attention neural networks
- ~ Self-attention

=> Transfer Learning in NLP :

- ~ Introduction to transformers.
- ~ Bert model.
- ~ Elmo model.
- ~ Gpt1 model
- ~ Gpt2 model.
- ~ Albert model.
- ~ Distilbert model

=> Deployment of Model and Performance Tuning :

- ~ Deep learning model deployment strategies.
- ~ Deep learning project architecture
- ~ Deep learning model deployment phase.
- ~ Deep learning model retraining phase.
- ~ Deep learning model deployment in aws.
- ~ Deep learning model deployment in azure.
- ~ Deep learning model deployment in gcloud.

=> API for Speech and Vision :

- ~ AWS
- ~ Azure
- ~ GCP

=> Big Data Introduction :

- ~ What is big data?
- ~ Big data application
- ~ Big data pipeline

=> Hadoop :

- ~ Hadoop introduction
- ~ Hadoop setup and installation

=> Spark :

- ~ Spark
- ~ Spark overview.
- ~ Spark installation.
- ~ Spark rdd.
- ~ Spark data frame.
- ~ Spark architecture.
- ~ Spark ml lib
- ~ Spark NLP
- ~ Spark linear regression
- ~ Spark logistic regression
- ~ Spark decision tree
- ~ Spark naive bayes
- ~ Spark xg boost.
- ~ Spark time series
- ~ Spark deployment in local server



- ~ Spark job automation with
- ~ Scheduler

=> Kafka :

- ~ Kafka introduction
- ~ Kafka installation
- ~ Spark streaming
- ~ Spark with Kafka

=> ML Ops :

- ~ Jenkins
- ~ Kubernetes
- ~ Elasticsearch
- ~ Kibana
- ~ Git

=> SQL :

- ~ Introduction
- ~ ER Daigram
- ~ Schema Design
- ~ Normalization
- ~ SQL SELECT Statement
- ~ SQL SELECT Using common functions
- ~ SQL JOIN Overview
- ~ INNER JOIN
- ~ LEFT JOIN
- ~ RIGHT JOIN
- ~ FULL JOIN
- ~ SQL Best Practice
- ~ INNER JOIN - Advanced
- ~ INNER JOIN & LEFT JOIN Combo
- ~ SELF JOIN
- ~ Joins & Aggregation - Subqueries
- ~ Sorting
- ~ Independent Subqueries
- ~ Correlated Subqueries
- ~ Analytic Function
- ~ Set Operations
- ~ SQL Views
- ~ Create a view
- ~ Create a view using DDL
- ~ SQL Insert - Advanced Technique
- ~ INSERT to create a table
- ~ INSERT new data to an existing table-1
- ~ INSERT new data to an existing table-2
- ~ INSERT new data to an existing table-3
- ~ INSERT new data to an existing table-4
- ~ SQL Update - Advanced Technique and TCL
- ~ SQL DELETE and TCL
- ~ SQL Constraints
- ~ SQL Aggregations
- ~ SQL Programmability
- ~ SQL Query Performance
- ~ SQL Xtras

=> Advance Excel :

- ~ Microsoft Excel Fundamentals
- ~ Entering and Editing Text and Formulas
- ~ Working with Basic Excel Functions
- ~ Modifying an Excel Worksheet
- ~ Formatting Data in an Excel Worksheet
- ~ Inserting Images and Shapes into an Excel Worksheet
- ~ Creating Basic Charts in Excel
- ~ Printing an Excel Worksheet
- ~ Working with Excel Templates
- ~ Working with an Excel List
- ~ Excel List Functions
- ~ Excel Data Validation
- ~ Importing and Exporting Data
- ~ Excel PivotTables
- ~ Working with Excel's PowerPivot Tools
- ~ Working with Large Sets of Excel Data
- ~ Conditional Functions
- ~ Lookup Functions
- ~ Text Based Functions
- ~ Auditing an Excel Worksheet
- ~ Protecting Excel Worksheets and Workbooks
- ~ Mastering Excel "What If?"Tools
- ~ Automating Repetitive Tasks in Excel with Macros
- ~ Macro Recorder Tool
- ~ Excel VBA Concepts
- ~ Advance VBA
- ~ Preparing and Cleaning Up Data with VBA
- ~ VBA to Automate Excel Formulas
- ~ Preparing Weekly Report
- ~ Working with Excel VBA User Forms
- ~ Importing Data from Text Files

=> Tableau :

- ~ Talking about Business Intelligence

- ~ Tools and Methodologies used in BI
- ~ Why Visualization is getting more popular
- ~ Why Tableau?
- ~ Gartner Magic Quadrant of Market Leaders
- ~ Future business impact of BI
- ~ Tableau Products
- ~ Tableau Architecture
- ~ BI Project Execution
- ~ Tableau Installation in local system
- ~ Introduction to Tableau Prep
- ~ Tableau Prep Builder User Interface
- ~ Data Preparation techniques using Tableau Prep Builder tool
- ~ How to connect Tableau with different data source
- ~ Visual Segments
- ~ Visual Analytics in depth
- ~ Filters, Parameters & Sets
- ~ Tableau Calculations using functions
- ~ Tableau Joins
- ~ Working with multiple data source (Data Blending)
- ~ Building Predictive Models
- ~ Dynamic Dashboards and Stories
- ~ Sharing your Reports
- ~ Tableau Server
- ~ User Security
- ~ Scheduling
- ~ PDF File
- ~ JSON File
- ~ Spatial File
- ~ Statistical File
- ~ Microsoft SQL Server
- ~ Salesforce
- ~ AWS
- ~ Azure
- ~ Google Analytics
- ~ R
- ~ Python
- ~ Hadoop
- ~ OneDrive
- ~ Microsoft Access
- ~ SAP HANA
- ~ SharePoint
- ~ Snowflake
- ~ Subject
- ~ Planning
- ~ Pen & Paper approach
- ~ Tools
- ~ Color theme
- ~ Shapes
- ~ Fonts
- ~ Image Selection
- ~ text position
- ~ visual placing
- ~ Story layout & design
- ~ Dashboard planning

=> Power BI :

- ~ Power BI introduction and overview
- ~ Key Benefits of Power BI
- ~ Power BI Architecture
- ~ Power BI Process
- ~ Components of Power BI
- ~ Power BI - Building Blocks
- ~ Power BI vs other BI tools
- ~ Power Installation
- ~ Overview of Power BI Desktop
- ~ Data Sources in Power BI Desktop
- ~ Connecting to a data Sources
- ~ Query Editor in Power BI
- ~ Views in Power BI
- ~ Field Pane
- ~ Visual Pane
- ~ Custom Visual Option
- ~ Filters
- ~ Introduction to using Excel data in Power BI
- ~ Exploring live connections to data with Power BI
- ~ Connecting directly to SQL Azure, HD Spark, SQL Server Analysis Services/ My SQL
- ~ Import Power View and Power Pivot to Power BI
- ~ Power BI Publisher for Excel
- ~ Content packs
- ~ Introducing Power BI Mobile
- ~ Power Query Introduction
- ~ Query Editor Interface
- ~ Clean and Transform your data with Query Editor
- ~ Data Type
- ~ Column Transformations vs Adding Columns
- ~ Text Transformations
- ~ Cleaning irregularly formatted data -Transpose
- ~ Date and Time Calculations
- ~ Advance editor: Use Case

- ~ Query Level Parameters
- ~ Combining Data Merging and Appending
- ~ Data Modelling
- ~ Calculated Columns
- ~ Measures/New Quick Measures
- ~ Calculated Tables
- ~ Optimizing Data Models
- ~ Row Context vs Set Context
- ~ Cross Filter Direction
- ~ Manage Data Relationship
- ~ Why is DAX important?
- ~ Advanced calculations using Calculate functions
- ~ DAX queries
- ~ DAX Parameter Naming
- ~ Time Intelligence Functions
- ~ Types of visualization in a Power BI report
- ~ Custom visualization to a Power BI report
- ~ Matrixes and tables
- ~ Getting started with color formatting and axis properties
- ~ Change how a chart is sorted in a Power BI report
- ~ Move, resize, and pop out a visualization in a Power BI report
- ~ Drill down in a visualization in Power BI

=> GPT-3

=> GAN

=> Reinforcement Learning

## Project details :-

=> Python Project :

- ~ Weeding script
- ~ Image resizing
- ~ Jupyter notebook merging, reading etc.
- ~ Sending emails
- ~ Weather app
- ~ Memes generator
- ~ Food log app
- ~ Web scrapping
- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis.
- ~ Integration with web portal.
- ~ Integration with rest api, web portal and mongo db. on azure
- ~ Deployment on web portal on azure.
- ~ Text mining
- ~ Social media data churn
- ~ Mass copy, paste

=> Chatbot Projects :

- ~ Chatbot using Microsoft Luis
- ~ Chatbot using google dialog flow
- ~ Chatbot using amazon lex
- ~ Chatbot using rasa nlu
- ~ Deployment of Chabot with web , telegram , WhatsApp, skype

=> Major Projects :

- ~ Healthcare analytics prediction of medicines based on Fitbit band.
- ~ Revenue forecasting for startups.
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ anomaly detection in inventory packaged material.
- ~ Fault detection in wafers based on sensor data.
- ~ Demand forecasting for fmcg product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with zomato dataset.
- ~ Fault detection in wafers based on sensor data.
- ~ Cement strength reg.
- ~ Credit card fraud.
- ~ Forest cover classification.
- ~ Fraud detection.
- ~ Income prediction.
- ~ Mushroom classifier.
- ~ phishing classifier
- ~ Thyroid detection.
- ~ Visibility climate

=> Computer Vision Project :

- ~ Traffic surveillance system.
- ~ Object identification.
- ~ Object tracking.
- ~ Object classification.
- ~ Tensorflow object detection.
- ~ Image to text processing.
- ~ Speech to speech analysis.
- ~ Vision based attendance system

=> Mini NLP Project :

- ~ Machine translation.
- ~ Abstractive text summarization.
- ~ Keyword spotting.
- ~ Language modelling.

~ Document summarization

=> NLP Transfer Learning Project :

- ~ Deployment and integration with UI machine translation.
- ~ Question answering (like chat bot)
- ~ Sentiment analysis imdb.
- ~ Text search (with synonyms).
- ~ Text classifications.
- ~ Spelling corrector.
- ~ Entity (person, place or brand) recognition.
- ~ Text summarization.
- ~ Text similarity (paraphrase).
- ~ Topic detection.
- ~ Language identification.
- ~ Document ranking.
- ~ Fake news detection
- ~ Plagiarism checker
- ~ Text summarization extractive
- ~ Text summarization abstractive.

=> NLP End to End Project with Architecture and Deployment :

- ~ Movie review using bert
- ~ NER using Bert
- ~ Pos bert
- ~ Text generation gpt 2
- ~ Text summarization xlnet
- ~ Abstract bert
- ~ Machine translation
- ~ Nlp text summarization custom
- ~ Keras/tensorflow
- ~ Language identification
- ~ Text classification using fast bert
- ~ Neuralcore
- ~ Detecting fake text using gltr with bert and gpt2
- ~ Fake news detector using gpt2
- ~ Python plagiarism checker type a message
- ~ Question answering

=> NLP Project End to End with Deployment in Various Cloud and UI Integration :

- ~ Topic modeling.
- ~ Word sense disambiguation
- ~ Text to speech
- ~ Keyword spotting
- ~ Document ranking
- ~ Text search (with synonyms)
- ~ Language modeling
- ~ Spam detector
- ~ Image captioning

=> SQL Project :

- ~ Ecommerce Analysis - Tableau Integration
- ~ Sales Data Analysis - Tableau Integration

=> Tableau Project :

- ~ Human Resource - Tableau
- ~ Supply Chain - Tableau
- ~ Sale Return - Tableau
- ~ E-Commerce Customer Analysis
- ~ Project Management Dashbaord
- ~ Sales Dashboard

=> Power BI Project :

- ~ Cost Insights - Power BI
- ~ Management Insights- Power BI
- ~ Retail Insights- Power BI

# GIT

---

Topic Name : DEVOPS

Sub-topic Name : GIT

Course link : <https://ineuron.ai/course/GIT>

## Course Description :-

The Git course will teach you how to utilise the Git version management system in a hands-on manner. Git is a collaborative file management system for large and small projects. As a result, the team can develop its product regularly.

## Course Features :-

- => Source code
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Git Introduction
- => Git Commands
- => Git Branching
- => Merging
- => Tagging
- => Rebasing

## Requirements :-

- => Prior knowledge of Linux
- => A System with good internet connection
- => Your dedication

## Instructors :-

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Git Introduction :

- ~ Git Introduction Preview
- ~ What is Version Control?
- ~ Types of Version Control
- ~ What is Git?
- ~ Why Git?
- ~ Git Installation in Windows
- ~ Git Installation in Linux
- ~ Git Setup
- ~ Git Terminologies

=> GIT Repository :

- ~ Repositories in GIT Preview
- ~ Creating Repository
- ~ Checking Repository History
- ~ Doing Commits
- ~ Git Diff
- ~ Git Restore
- ~ Git Ignore

=> Git Commands :

- ~ Tagging Preview
- ~ Branching
- ~ Branching Practicals
- ~ Merging
- ~ Merge Conflicts

=> Git Branching :

- ~ Remote Repository
- ~ Cloning Repository
- ~ Working with Remote Repository
- ~ Pushing to Remote Failed in Github
- ~ Personal Access Token Setup in Windows
- ~ Personal Access Token Setup in Linux
- ~ Pull Request
- ~ GIT Fetch & Pull

~ *Fork*

=> **Rebasing :**

~ *Rebasing*

~ *Interactive Rebasing*

~ *Git Rewrite History*

~ *Git Rewrite History continued*

~ *Cherry Picking*

=> **Outro :**

~ *Modify Recent Commits*

~ *Git Revert*

~ *Git Checkout*

~ *Git Reset*

~ *Git Stash*

~ *Git Reflog*

~ *Course Outro*

# Complete Backend development with Nodejs

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : NODE JS

Course link : <https://ineuron.ai/course/Complete-Backend-development-with-Nodejs>

## Course Description :-

Javascript is being used for much more than it originally intended. All backend work may now be done through javascript. In this course, we will learn how to use current javascript to develop comprehensive backend code. To begin, we'll use VSCode to set up some tools. Then we'll learn how to build our own web server without using any third-party modules. After that, we'll learn express. We will also learn how to send web-based and JSON-based responses. We'll go into body parser, middleware, and templating in more detail later.

## Course Features :-

- => Course material
- => Course resources
- => On-demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Javascript
- => ES6
- => Web servers
- => Express Js
- => Body parsers
- => Middlewares
- => View Engines
- => Multer
- => Passport JS
- => Big Stack
- => MongoDB integration with Node JS

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction to NodeJS course :

- ~ Tools to be downloaded
- ~ Setting up VSCode and reading docs
- ~ Our very first project - web server Preview
- ~ How to get exercise files

=> A web server - Manual work and reading docs :

- ~ Reading Docs for next project
- ~ Configuration for html js and css files Preview
- ~ Finding the file on the server
- ~ Handling server error response
- ~ Finishing up node server project

=> Learn Express :

- ~ What is ExpressJs and templating Preview
- ~ Installing express and detail about package file

- ~ *Creating routes using express*
- ~ *Get Post and Delete requests - Postman*
- ~ *Routing in express*
- ~ *Server response and status code*

=> Body parser, middleware and view engines :

- ~ *What is a middleware*
- ~ *Applying bodyparser with express*
- ~ *Serving static files and form data*
- ~ *Using template engine*

=> Multer - Upload a user profile photo :

- ~ *Overview of multer and documentation*
- ~ *Setting up multer*
- ~ *configuring multer for uploads*
- ~ *Change profile pic using multer*

=> PassportJS - Facebook Authentication :

- ~ *Authentication in nodeJs*
- ~ *Setting up facebook app*
- ~ *Installing dependencies*
- ~ *Create all views*
- ~ *Configuring middleware*
- ~ *Configuring our routes*
- ~ *http and https problem in facebook auth*
- ~ *A demo on Heroku - Not a heroku tutorial Preview*

=> Moving to Big Stack Project :

- ~ *Why we are using MongoDB*
- ~ *Setting up an Amazon instance using Mlab*
- ~ *Take time and read these npm docs*
- ~ *Design scalable folder structure*
- ~ *Creating home route and setup*

=> Move to MongoDB :

- ~ *Connect your project with mongoDB*
- ~ *Creating auth API and a challenge*
- ~ *Solution of challenge*
- ~ *Creating a person schema*
- ~ *Our first query in MongoDB*
- ~ *Creating new object from Mongo model*
- ~ *Generating salt and hash to save password*
- ~ *Using postman for testing*

=> Bigstack Major Project - login routes and tokens :

- ~ *Setting up login route*
- ~ *Validation of password in login route Preview*
- ~ *Creating a Strategy using Passport*
- ~ *Creating tokens with information*
- ~ *Fixing errors and profile route*

=> Bigstack Major Project Working on User Profile :

- ~ *Creating model for UserProfile*
- ~ *Creating route for profile*
- ~ *Collecting user profile values*
- ~ *Update the profile values and save them*
- ~ *Debugging routes part 1*
- ~ *Debugging application - part 2*

=> Bigstack Major Project Unique Username and other routes :

- ~ *Unique username and url based access*
- ~ *Getting all users from database*
- ~ *Deleting a user from database*
- ~ *Workrole - Pushing array in database*
- ~ *Testing array based routes*
- ~ *Writing and testing delete route in array*

=> Bigstack Major Project - Questions and Upvotes :

- ~ *Question Model - Challenge*
- ~ *Creating question model*
- ~ *Creating post question routes and debugging*
- ~ *How to take help from stackoverflow and get route*
- ~ *Posting answers for questions*
- ~ *Upvotes routes and some assignments*



# Azure Data Engineering

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA ON CLOUD

Course link : <https://ineuron.ai/course/Azure-Data-Engineering>

## Course Description :-

Explore how the world of data has evolved and how the advent of cloud technologies is providing new opportunities for business to explore. You will learn the various data platform technologies that are available, and how a Data Engineer can take advantage of this technology to an organization benefit.

## Course Features :-

- => Self paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Learn about the responsibilities of a data engineer
- => Find out how they relate to the jobs of other data and AI professionals.
- => Explore common data engineering practices and a high-level architecting process for a data-engineering project.

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Introduction to cloud :
  - ~ Introduction to cloud computing
  - ~ cloud models
  - ~ Different cloud providers Preview
- => Regions and Availability Zones :
  - ~ Understanding Regions and Availability Zones in Azure
  - ~ creating Microsoft azure account
- => Resource Hierarchy :
  - ~ Understanding Resource Hierarchy
  - ~ Demo on Resource Hierarchy Preview
  - ~ Resource groups, subscription and management groups
- => Azure Active Directory :
  - ~ Active Directory part 1
  - ~ Active Directory part 2
- => Introduction to azure cloud computing :
  - ~ Azure services overview
  - ~ managed and unmanaged service
  - ~ demo create azure sql database service
- => Introduction to data engineer profile :
  - ~ Introduction
  - ~ data engineer role and responsibility
  - ~ introduction to data engineer technologies
- => Azure sql database :
  - ~ Module Introduction
  - ~ Introduction
  - ~ Why choosing sql server in azure
  - ~ Azure laas vs Paas database offerings
  - ~ SQL server paas deployment options
  - ~ Introduction to Azure sql server in virtual machine
  - ~ sql server in azure virtual machine
  - ~ demo part 1 sql server in azure virtual machine
  - ~ demo part 2 sql server in azure virtual machine
  - ~ introduction azure single database
  - ~ Demo Azure single database
  - ~ purchasing models and service tier
  - ~ azure database vs azure datawarehouse

- ~ *introduction elastic data pool*
- ~ *azure elastic database*
- ~ *Demo part 1 Azure Elastic database Preview*
- ~ *Demo part 2 Azure Elastic database*
- ~ *introduction managed instance database*
- ~ *azure managed instance database*
- ~ *difference between on premises and managed instance*
- ~ *service tiers for managed instance*
- ~ *management operations*
- ~ *demo managed instance*

# Tibco Spotfire

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Tibco-Spotfire>

## Course Description :-

Upskill your analytics skills through the power of Tibco Platform. Tibco spotfire is an analytical platform enabling us to explore and visualize new discoveries in data through immersive dashboards and advanced analytics. You will be able to enhance your skill by exploring this Tibco Analytical platform.

## Course Features :-

- => Learning of different tibco spotfire Services
- => Understanding Data integration
- => BI & Analytics
- => Creating charts, reports, graphs, dashboards for analytical purpose
- => Completion Certificate
- => Understanding spotfire Features

## What you will learn :-

- => Tibco spotfire Features
- => Analytics
- => spotfire mods
- => Dashboard
- => creating report
- => creating Dashboard

## Requirements :-

- => No prior knowledge in Analytics
- => System with Internet Connection
- => Interest to learn
- => Basic knowledge of BI
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Tibco Fundamentals :
  - ~ Overview Preview
  - ~ Tibco feature Preview
  - ~ Tibco benefits features
  - ~ Tibco dashboard overview
  - ~ Tibco spotfire mods
  - ~ Tibco spotfire analysis
  - ~ Tibco spotfire handson

# Write quicker HTML5 and CSS 3; productivity hacks with emmet4

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : HTML

Course link : <https://ineuron.ai/course/Write-quicker-HTML5-and-CSS-3;-productivity-hacks-with-emmet4>

## Course Description :-

This course will help you to grab the fundamentals of emmet when using HTML5 and CSS3.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Getting started with emmet
- => Emmet and speedy html
- => Parent child and grouping
- => emmet in css
- => An old style blog
- => Why DOM is important
- => Inline vs Block and bring in images
- => Lists and interlinking pages
- => Getting a video on service page
- => 3 Plans in a table
- => GET and POST forms
- => Types of input forms
- => Secret to learn CSS

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> We do not write without emmet :

- ~ Getting started with emmet
- ~ Emmet and speedy html
- ~ Parent child and grouping
- ~ emmet in css

=> HTML and DOM :

- ~ An old style blog
- ~ Why DOM is important
- ~ Inline vs Block and bring in images
- ~ Lists and interlinking pages
- ~ Getting a video on service page
- ~ 3 Plans in a table
- ~ GET and POST forms
- ~ Types of input forms

=> Starting with CSS :

- ~ Secret to learn CSS
- ~ Explore and bring in fonts
- ~ Bring in colors and styles
- ~ Transition and box shadow DOCS

- ~ *Margin and padding*
- ~ *Button gets all and assignment*

# PowerBI Foundation

---

Topic Name : DATA ANALYTICS

Sub-topic Name : POWER BI

Course link : <https://ineuron.ai/course/PowerBI-Foundation>

## Course Description :-

Microsoft Power BI is a data and analytics reporting tool that helps organizations bring together disparate data sets into reporting dashboards. Power BI is a suite of business analytics tools to analyze data and share insights. Power BI dashboards provide a 360-degree view for business users with their most important metrics in one place, updated in real time, and available on all of their devices.

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Assignment
- => Quiz

## What you will learn :-

- => PowerBI tool
- => Complete MS Excel and its Usages
- => Practical implementation
- => MS Excel utility

## Requirements :-

- => Computer with Internet Connectivity
- => Basic programming understanding

## Instructors :-

- => Amit Bose :
- ~

## Curriculum details :-

- => PowerBI Types of Connections and Connectors :
  - ~ Introduction Preview
- => Power BI Introduction
- => Power BI DAX1 Day3
- => Working with different DAX functions- Table relations in Power BI
- => Power BI Report&Visuals
- => Power Query in Detail
- => DAX Architecture
- => Power Query | Data Modelling | DAX in Detail Power BI Part 2
- => Power Query | Data Modelling | DAX in Detail Power BI

# Complete Flutter Course - iOS Android Apps

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : FLUTTER

Course link : <https://ineuron.ai/course/Complete-Flutter-Course---iOS-Android-Apps>

## Course Description :-

Learn how to use Flutter, Google's latest mobile framework, to develop quick and beautiful mobile apps. With no prior expertise, you will rapidly learn how to construct any application with Flutter in this course. upon successful completion of this course, you will be able to create interactive and responsive applications using the flutter development kit.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Stateless widgets
- => Stateful widgets
- => Background changers
- => Randomization
- => Camera and gestures
- => API handling
- => SQLite
- => Firebase
- => BLOC in flutter

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to Flutter :
  - ~ Introduction to flutter
- => Installation of Flutter :
  - ~ Flutter installation on MAC
  - ~ Flutter doctor - Do not panic
  - ~ Flutter installation for Windows
  - ~ Installing plugins for VSCode
- => First Flutter project :
  - ~ Creating first project in flutter
  - ~ Run your flutter project
  - ~ Actual hello world
- => Stateless and Stateful - 2 projects :
  - ~ Stateless and Stateful Widgets
  - ~ How to read flutter documentation
  - ~ Scaffold widget in flutter
  - ~ A stateless app in flutter
  - ~ Multi child layout in flutter
  - ~ Raised buttons in flutter
  - ~ Converting into stateless widget
  - ~ Making a stateful app

- ~ *Designing Visual part*
- ~ *Finishing stateful number app*

=> Background Changer and randomization :

- ~ *What we will create in this section*
- ~ *Stateless widget work*
- ~ *Random value generation in flutter*
- ~ *Button properties in bgchanger*

=> Dice Roller App project :

- ~ *Getting assets for dice roller flutter*
- ~ *create main dart in dice app*
- ~ *logic part of dice roller*
- ~ *Design of dice roller and assignment*

=> Tic Tac Toe App in flutter :

- ~ *Getting started with TicTacToe in flutter*
- ~ *taking main dart file for TicTacToe*
- ~ *Initialize state for Tic Tac Toe*
- ~ *playgame and reset game tictactoe*
- ~ *Winning logic for Tictactoe in flutter*
- ~ *Understand gridview in flutter*
- ~ *Final design of TicTacToe and assignment*

=> Scratch and win App in flutter :

- ~ *Scratch and win assets in flutter*
- ~ *scratch and win todos*
- ~ *rest game and lucky number in flutter*
- ~ *showall and gameplay*
- ~ *Finishing sratch and win in flutter*

=> Spanish Audio number app :

- ~ *Third pary libraries in flutter*
- ~ *Audio helper in flutter*
- ~ *defining audio methods in spanish app*
- ~ *Flinishing spanish number app in flutter*

=> Camera and Gesture :

- ~ *Reading gesture docs*
- ~ *Preparing project with imagepicker*
- ~ *methods for camera and gallery and ios fix*
- ~ *Finishup camra app in flutter*

=> Navigation and keys in flutter - 2 apps :

- ~ *Reading assignment for drawer and keys*
- ~ *Creating catogory page*
- ~ *Routing basics in flutter*
- ~ *drawer links and navigation*
- ~ *drawer app assignment*
- ~ *A signup app*
- ~ *bring in logo assets*
- ~ *Design your first input field*
- ~ *Global keys and validator*
- ~ *Collect key values in next screen*
- ~ *HomePage and assignment*

=> API handling in flutter - 2 Apps :

- ~ *introduction to API in flutter*
- ~ *passing key in stateful*
- ~ *Making a web request with Future*
- ~ *storing web response*
- ~ *getting data on screen and debug*
- ~ *Understand the API response*
- ~ *Fetching data with web*
- ~ *picking up data from JSON*
- ~ *Run the app and assignment*

=> Sqlite - A publishable App :

- ~ *Before we start this project*
- ~ *Reading the docs for database*
- ~ *adding dependencies for database*
- ~ *custom notes class part 1*
- ~ *custom notes class part 2*
- ~ *Start with database helper file*
- ~ *create table in sqlite*
- ~ *insert update and delete query*
- ~ *Get value count from database*
- ~ *creating semi list screen*
- ~ *rewriting stateful widget of detail class*
- ~ *saving notes and helper method*
- ~ *delete and UI part of details screen*
- ~ *Adding methods in listview*
- ~ *Finally done with this app*

=> Firebase and flutter - Authentication :

- ~ *Firebase for flutter*
- ~ *exploring firebase*
- ~ *Configure iOS and Android app for firebase*
- ~ *Define router in main*
- ~ *add lister to check state of login*
- ~ *Signin with email and password*
- ~ *Android X bug and signin UI*



- ~ Home page logic methods
- ~ Home page UI and link for signup
- ~ logic part of singup page
- ~ Final one on authentication

#### => Firebase Database and Storage :

- ~ getting started with database and storage
- ~ stackoverflow and file structure
- ~ creating model for contact
- ~ upgrading to AndroidX and homePage
- ~ uploading image and data to firebase
- ~ UI for add screen
- ~ UI for home page and bug fix
- ~ Get values from snapshot
- ~ Phone and sms intent launch
- ~ Delete contact from firebase
- ~ Edit screen - passing id
- ~ upload new photo in editcontact
- ~ final touch to database app - contact

#### => UI Challenge - WhatsApp :

- ~ Create whatsapp project and exercise files
- ~ reusable widgets
- ~ creating whatsapp title bar
- ~ creating tab bar
- ~ creating chat UI page
- ~ Design calls ui
- ~ Status screen ui

#### => BLOC in flutter :

- ~ What is BLOC in flutter
- ~ Creating a flutter block app - structure
- ~ Creating BLOC pattern code in flutter
- ~ Creating UI for BLOC project and calls

# GCP Interview

---

Topic Name : CLOUD

Sub-topic Name : GCP INTERVIEW PREPARATION

Course link : <https://ineuron.ai/course/GCP-Interview>

## Course Description :-

This course is designed to prepare you for certifications and provide you with a comprehensive path to help you get started in your career as a newbie, or even an experienced individual, with a deeper grasp of Google Cloud architecture and services.

## Course Features :-

- => Roadmap
- => Interview questions
- => Resume preparation
- => Completion Certificate
- => Downloadable resources

## What you will learn :-

- => How to design solutions for GCP platform

- => Understand the important concepts of GCP
- => Interview questions
- => Sample resumes

## Requirements :-

- => Prior understanding of GCP

- => GCP account
- => A System with internet connection
- => Your dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Basic cloud computing :

- ~ What are the key features of cloud computing ? Preview
- ~ What are the different cloud deployment models?
- ~ What is the following is not true about IaaS ?
- ~ Which of the following Statements regarding a private cloud delivery model are inaccurate ?
- ~ An organization has decided to host its website on Microsoft Azure using WordPress. The CFO would like to know what the best delivery model is for all customers. The CFO wants to be assured the website is publicly accessible. What would you recommend?
- ~ Your organization, a health care practice, is required by law to maintain patient records for seven years. Recently, the organization invested in an electronic health records (EHR) system. The business has been in practice for 18 years and still maintains 5,000+ previous patient files from the past. By law, all these records must be digitized. What type of cloud solution deployment model should the EHR company suggest the health practice implement?
- ~ The most fundamental unit of cloud computing is?

=> GCP :

- ~ Which of the following is not database services offered by Google GCP? Preview
- ~ Which of the following services are serverless?
- ~ Which of the below gcloud command is used to create a custom role?
- ~ Which Google Cloud Platform service can be used for serverless file processing and running website backend?
- ~ Which of the following Cloud shell command will open a blank editor window?
- ~ In Google Cloud Platform, your network and all its resources are considered as?
- ~ Cloud Vision is a \_\_\_\_
- ~ How are the Google Compute Engine and Google App Engine related?
- ~ What is Google App Engine?
- ~ You have a definition for an instance template that contains a web application. You have been asked to deploy the application to scale based on the HTTP traffic it receives. What should you do?

~ You are creating a Kubernetes Engine cluster to deploy multiple pods inside the cluster. All container logs must be stored in BigQuery for later analysis. You want to follow Google-recommended practices. Which two approaches can you take?

=> gcloud :

~ Which of the following gcloud command is used to set scopes?  
~ You have a project using BigQuery. You want to list all BigQuery jobs for that project. You want to set this project as the default for the bq command-line tool. What should you do?  
~ Your project has all its Compute Engine resources in the europe-west1 region. You want to set europe-west1 as the default region for gcloud commands. What should you do?

=> Billing :

~ What is the difference between Billing Alerts and Budget Quotas in GCP ?  
~ Which one is not GCP Billing and Budgeting services ?  
~ Your company has reserved a monthly budget for your project. You want to be informed automatically of your project spend so that you can take action when you approach the limit. What should you do?  
~ You developed a new application for App Engine and are ready to deploy it to production. It would be best to estimate the costs of running your application on the Google Cloud Platform as accurately as possible. What should you do?  
~ It would help to estimate the annual cost of running a Bigquery query that is scheduled to run nightly. What should you do?  
~ Scopes are access controls that are applied to \_\_\_\_\_?  
~ When admin assigns read-only permission to a custom role? What level of privileges are given?  
~ Which one is not the method for the authentication of Google Compute Engine API?  
~ What is the command to authenticate through Docker Container Registry ?  
~ You are a project owner and need your co-worker to deploy a new version of your application to App Engine. You want to follow Googles recommended practices. Which IAM roles should you grant your co-worker?  
~ You want to find out who in your organization has Owner access to a project called "my-project".What should you do?  
~ You want to create a new role for your colleagues that will apply to all current and future projects created in your organization. The role should have the permissions of the BigQuery Job User and Cloud Bigtable User roles. You want to follow Googles recommended practices. How should you create the new role?  
~ You work in a small company where everyone should be able to view all resources of a specific project. You want to grant them access following Google's recommended practices. What should you do?

=> IAM :

~ You created an update for your application on App Engine. You want to deploy the update without impacting your users. You want to be able to roll back as quickly as possible if it fails. What should you do?

=> Compute Engine :

~ In Google cloud platform, pricing of a VM can be vary based on \_\_\_\_\_  
~ In Google cloud platform, you cannot increase RAM of deployed VM instance.  
~ In GCP, to change the machine type of an existing VM instance, the instance must be  
~ You have an application server running on Compute Engine in the europe-west1-d zone. You need to ensure high availability and replicate the server to the europe-west2-c zone using the fewest steps possible. What should you do?  
~ If an instance is deleted by mistake, is it possible to retrieve it back? If yes then how?

=> Storage :

~ What are the libraries and tools for cloud storage on GCP ?  
~ Your company processes high volumes of IoT data that are time-stamped. The total data volume can be several petabytes. The data needs to be written and changed at a high speed. You want to use the most performant storage option for your data. Which product should you use?  
~ Your application has a large international audience and runs stateless virtual machines within a managed instance group across multiple locations. One feature of the application lets users upload files and share them with other users. Files must be available for 30 days; after that, they are removed from the system entirely. Which storage solution should you choose?  
~ Your company has a mission-critical application that serves users globally. You need to select a transactional, relational data storage system for this application. Which two products should you choose?

=> App Engine :

~ You need to create a new Kubernetes Cluster on Google Cloud Platform that can autoscale the number of worker nodes. What should you do?  
~ You have a Kubernetes cluster with 1 node-pool. The cluster receives a lot of traffic and needs to grow. You decide to add a node. What should you do?  
~ You have created a Kubernetes deployment, called Deployment-A, with 3 replicas on your cluster. Another deployment, called Deployment-B, needs access to Deployment-A. You cannot expose Deployment-A outside of the cluster. What should you do?

=> Kubernetes :

~ You need to create a new Kubernetes Cluster on Google Cloud Platform that can autoscale the number of worker nodes. What should you do?  
~ You have a Kubernetes cluster with 1 node-pool. The cluster receives a lot of traffic and needs to grow. You decide to add a node. What should you do?  
~ You have created a Kubernetes deployment, called Deployment-A, with 3 replicas on your cluster. Another deployment, called Deployment-B, needs access to Deployment-A. You cannot expose Deployment-A outside of the cluster. What should you do?

=> ML :

~ You work for a textile manufacturer and have been asked to build a model to detect and classify fabric defects. You trained a machine learning model with high recall based on high resolution images taken at the end of the production line. You want quality control inspectors to gain trust in your model. Which technique should you use to understand the rationale of your classifier?  
~ You need to write a generic test to verify whether Dense Neural Network (DNN) models automatically released by your team have a sufficient number of parameters to learn the task for which they were built. What should you do?  
~ Your team is using a TensorFlow Inception-v3 CNN model pretrained on ImageNet for an image classification prediction challenge on 10,000 images. You will use AI Platform to perform the model training. What TensorFlow distribution strategy and AI Platform training job configuration should you use to train the model and optimize for wall-clock time?  
~ You work on a team where the process for deploying a model into production starts with data scientists training different versions of models in a Kubeflow pipeline. The workflow then stores the new model artifact into the corresponding Cloud Storage bucket. You need to build the next steps of the pipeline after the submitted model is ready to be tested and deployed in production on AI Platform. How should you configure the architecture before deploying the model to production?  
~ You work for a maintenance company and have built and trained a deep learning model that identifies defects based on thermal images of underground electric cables. Your dataset contains 10,000 images, 100 of which contain visible defects. How should you evaluate the performance of the model on a test dataset?  
~ You are an ML engineer at a media company. You want to use machine learning to analyze video content, identify objects, and alert users if there is inappropriate content. Which Google Cloud products should you use to build this project?  
~ You work for a large retailer. You want to use ML to forecast future sales leveraging 10 years of historical sales data. The historical data is stored in Cloud Storage in Avro format. You want to rapidly  
~ You need to build an object detection model for a small startup company to identify if and where the companys logo appears in an image. You were given a large repository of images, some with logos and some without. These images are not yet labelled. You need to label these pictures, and then train and deploy the model. What should you do?  
~ You work for a large financial institution that is planning to use Dialogflow to create a chatbot for the companys mobile app. You have reviewed old chat

logs and tagged each conversation for intent based on each customers stated intention for contacting customer service. About 70% of customer inquiries are simple requests that are solved within 10 intents. The remaining 30% of inquiries require much longer and more complicated requests. Which intents should you automate first?

~ You work for a gaming company that develops and manages a popular massively multiplayer online (MMO) game. The games environment is open-ended, and a large number of positions and moves can be taken by a player. Your team has developed an ML model with TensorFlow that predicts the next move of each player. Edge deployment is not possible, but low-latency serving is required. How should you configure the deployment?

~ You should feed your machine learning model your \_\_\_\_ not \_\_\_\_\_. It will learn those for itself.

~ You are building an ML model to detect anomalies in real-time sensor data. You will use Pub/Sub to handle incoming requests. You want to store the results for analytics and visualization. How should you configure the pipeline?

~ Your company manages a video sharing website where users can watch and upload videos. You need to create an ML model to predict which newly uploaded videos will be the most popular so that those videos can be prioritized on your companys website. Which result should you use to determine whether the model is successful?

~ You are working on a Neural Network-based project. The dataset provided to you has columns with different ranges. While preparing the data for model training, you discover that gradient optimization is having difficulty moving weights to a good solution. What should you do?

~ Your data science team needs to rapidly experiment with various features, model architectures, and hyperparameters. They need to track the accuracy metrics for various experiments and use an API to query the metrics over time. What should they use to track and report their experiments while minimizing manual effort?

~ You work for a bank and are building a random forest model for fraud detection. You have a dataset that includes transactions, of which 1% are identified as fraudulent. Which data transformation strategy would likely improve the performance of your classifier?

~ Your team is working on an NLP research project to predict political affiliation of authors based on articles they have written. You have a large training dataset that is structured like this: You followed the standard 80%-10%-10% data distribution across the training, testing, and evaluation subsets. How should you distribute the training examples across the train-test-eval subsets while maintaining the 80-10-10 proportion?

~ Your team has been tasked with creating an ML solution in Google Cloud to classify support requests for one of your platforms. You analyzed the requirements and decided to use TensorFlow to build the classifier so that you have full control of the models code, serving, and deployment. You will use Kubeflow pipelines for the ML platform. To save time, you want to build on existing resources and use managed services instead of building a completely new model. How should you build the classifier?

~ You recently joined a machine learning team that will soon release a new project. As a lead on the project, you are asked to determine the production readiness of the ML components. The team has already tested features and data, model development, and infrastructure. Which additional readiness check should you recommend to the team?

~ You work for a credit card company and have been asked to create a custom fraud detection model based on historical data using AutoML Tables. You need to prioritize detection of fraudulent transactions while minimizing false positives. Which optimization objective should you use when training the model?

~ You work for an online travel agency that also sells advertising placements on its website to other companies. You have been asked to predict the most relevant web banner that a user should see next. Security is important to your company. The model latency requirements are 300ms@p99, the inventory is thousands of web banners, and your exploratory analysis has shown that navigation context is a good predictor. You want to implement the simplest solution. How should you configure the prediction pipeline?

~ Your team is building a convolutional neural network (CNN)-based architecture from scratch. The preliminary experiments running on your on-premises CPU-only infrastructure were encouraging, but have slow convergence. You have been asked to speed up model training to reduce time-to-market. You want to experiment with virtual machines (VMs) on Google Cloud to leverage more powerful hardware. Your code does not include any manual device placement and has not been wrapped in Estimator model-level abstraction. Which environment should you train your model on?

# Computer Vision Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION PROJECT

Course link : <https://ineuron.ai/course/Computer-Vision-Projects>

## Course Description :-

Computer vision is the study of how computers extract useful information from photos or videos and put it to use in various applications. Examples include reverse engineering, security checks, image editing and processing, computer animation, autonomous navigation, and robots. This course will help you learn and implement various unique vision-based projects with real-world applications and use Computer Vision to tackle numerous real-world issues.

## Course Features :-

- => Real-time project building
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Working with desktop & web applications
- => Computer vision projects like object detection, object tracking & much more.
- => Full project pipelines from scratch to deployment.
- => Project presentation skills

## Requirements :-

- => Prior knowledge in Python
- => Prior knowledge in Computer vision models
- => Quiz questions
- => A system with a stable Internet Connection
- => Your dedication

## Instructors :-

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Introduction to Course :

- ~ Introduction to the course Preview
- ~ Course Curriculum
- ~ Installing Software and Applications
- ~ Working with Anaconda environments
- ~ Pycharm introduction
- ~ Pycharm with conda
- ~ Pycharm with venv
- ~ Pycharm with pipenv

=> Covering Python Basics :

- ~ Building a calculator part Preview
- ~ Working with command line arguments
- ~ Building the Flask Application
- ~ Testing our app in Postman
- ~ Learn to debug with Pycharm
- ~ Adding an UI to our Web App

=> Intro to Object detection :

- ~ What is Object Detection? Preview
- ~ What are Bounding boxes?
- ~ Applications of Object detection
- ~ Metrics used in Object detection

=> Practicals Object Detection using Tensorflow 1.x :

- ~ Introduction to TFOD1.x
- ~ Using Google colab with Google drive
- ~ Installation of Libraries in colab
- ~ TFOD 1.x setup in colab
- ~ Visiting the Model Zoo
- ~ Inferencing in Colab

- ~ *Inferencing in Local*
- ~ *Important Configuration Files*
- ~ *Webcam Testing*

=> Practicals Training a Custom Cards Detector using Tensorflow1.x :

- ~ *Custom model training in TFOD 1.x*
- ~ *Our custom dataset*
- ~ *Doing Annotations or labeling data*
- ~ *Selection of pretrained model from Model zoo*
- ~ *Files setup for training*
- ~ *Let's start Training in Colab*
- ~ *Export Frozen inference graph*
- ~ *Inferencing with our trained model in Colab*
- ~ *Training in Local*
- ~ *Inferencing with our trained model in Local*

=> Practicals Creating an Cards Detector Web App with TFOD1 :

- ~ *Creating a Pycharm project & Environment Setup*
- ~ *WebApp workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*
- ~ *Debugging our Application*

=> Practicals Object Detection using Tensorflow 2.x :

- ~ *Introduction to TFOD2.x*
- ~ *Using the default colab notebook*
- ~ *Google colab & Drive setup*
- ~ *Visiting TFOD2.x Model garden*
- ~ *Inference using Pretrained model*

=> Inferencing in Local with a pretrained model :

- ~ *Custom model training in TFOD 2.x*
- ~ *Our custom dataset*
- ~ *File setup for training*
- ~ *Let's start training*
- ~ *Stop training or resume training*
- ~ *Evaluating the trained model*
- ~ *Convert CKPT to saved model*
- ~ *Inferencing using the custom trained model in colab*
- ~ *Inferencing using the custom trained model in local PC*

=> Practicals Training a Custom Chess Piece Detector using Tensorflow2 :

- ~ *Creating a pycharm project & environment setup*
- ~ *Application workflow*
- ~ *Code understanding*
- ~ *Testing our app with postman*
- ~ *Debugging our application*

=> Practicals creating an chess piece detector web app with TFOD2 :

- ~ *Introduction to detectron2*
- ~ *Detectron2 colab setup*
- ~ *Visiting detectron2 model zoo*
- ~ *Detectron2 pretrained model inferencing*

=> Practicals object detection using detectron2 :

- ~ *Detectron2 custom training*
- ~ *Exploring the dataset*
- ~ *Registering dataset for training*
- ~ *Let's start training*
- ~ *Inferencing using the custom trained model in colab*
- ~ *Evaluating the model*

=> Practicals training a custom detector using detectron2 :

- ~ *Creating a pycharm project & environment setup*
- ~ *Application workflow*
- ~ *Code understanding*
- ~ *Testing our app with postman*
- ~ *Debugging our application*

=> Practicals creating an custom detector web app with detectron2 :

- ~ *Introduction to yolov5*
- ~ *Yolov5 colab setup*
- ~ *Inferencing using pre trained model*

=> Practicals object detection using yolov5 :

- ~ *Custom training with yolov5*
- ~ *Exploring the dataset*
- ~ *Doing annotations or labeling data*
- ~ *Setting up google colab & drive*
- ~ *Let's start training*
- ~ *Inferencing using the custom trained model in colab*

=> Practicals training a custom warehouse apparel detector using yolov5 :

- ~ *Creating a pycharm project & environment setup*
- ~ *Application workflow*
- ~ *Code understanding*
- ~ *Testing our app with postman*
- ~ *Debugging our application*

=> Practicals creating an warehouse apparel detector web app with YOLOV5 :

- ~ *Introduction to vehicle detection project*
- ~ *Requirement gathering*

- ~ Framework selection
- ~ Detailed project workflow
- ~ Data collection
- ~ Data preparation
- ~ Data augmentation
- ~ Data annotations
- ~ Model training
- ~ Creating a pycharm project & environment setup
- ~ Webapp workflow
- ~ Code understanding
- ~ Prediction with postman
- ~ Debugging our application

=> Traffic vehicle detection :

- ~ Object tracking project
- ~ Project installation
- ~ Project demo
- ~ Code understanding

=> Object tracking with detection :

- ~ Introduction to helmet detection project
- ~ Requirement gathering
- ~ Techstack selection
- ~ Detailed project workflow
- ~ Data collection
- ~ Data preparation
- ~ Data augmentation
- ~ Data annotations
- ~ Model training
- ~ Creating a pycharm project & environment setup
- ~ Webapp workflow
- ~ Code understanding
- ~ Prediction with postman
- ~ Debugging our application

=> Helmet detection :

- ~ Introduction to fashion apparel detection project
- ~ Requirement gathering
- ~ Techstack selection
- ~ Detailed project workflow
- ~ Data collection
- ~ Data preparation
- ~ Data augmentation
- ~ Data annotations
- ~ Model training
- ~ Creating a pycharm project & environment setup
- ~ Project demo
- ~ Webapp workflow
- ~ Code understanding
- ~ Prediction with postman
- ~ Debugging our application

=> Fashion apparel detection :

- ~ Introduction to project
- ~ Project installation
- ~ Project demo
- ~ Application Workflow
- ~ Code Understanding
- ~ Debugging our App
- ~ Different OCR's available

=> Image TO text OCR :

- ~ Introduction to Project
- ~ Requirement Gathering
- ~ Techstack Selection
- ~ Project Installation
- ~ Project Demo
- ~ Project Workflow
- ~ Core Components of the Application
- ~ Data Collection Module
- ~ Generate Face Embeddings
- ~ Training Face Recognition Module
- ~ Prediction Pipeline
- ~ Entrypoint of the Application
- ~ Application Workflow

=> Vision based attendance system :

- ~ Introduction to Shredder Systems
- ~ Requirement Gathering
- ~ Techstack Selection
- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation
- ~ Model Selection from Zoo
- ~ Model Training
- ~ Creating a Pycharm project & Environment Setup
- ~ Application Workflow
- ~ Project Demo
- ~ Code Understanding

- ~ *Debugging our Application*
- ~ *Project Workflow*

=> **Shredder System :**

- ~ *Introduction to ANPR Project*
- ~ *Requirement Gathering*
- ~ *Tech Stack Selection*
- ~ *Project Workflow*
- ~ *Data Collection*
- ~ *Data Augmentation*
- ~ *Data Preparation*
- ~ *Data Annotation*
- ~ *Model Selection From Zoo*
- ~ *Model Training*
- ~ *Creating a Pycharm project & Environment Setup*
- ~ *Application Workflow*
- ~ *Create Google OCR API Key*
- ~ *Project Demo*
- ~ *Code Understanding*
- ~ *Debugging our Application*

=> **Automatic Number plate Recognition with TFOD1.x**



# DevOps Foundation

---

Topic Name : DEVOPS

Sub-topic Name : DEVOPS MASTERS

Course link : <https://ineuron.ai/course/DevOps-Foundation>

## Course Description :-

DevOps is a set of practices that works to automate and integrate the processes between software development and IT teams, so they can build, test, and release software faster and more reliably. The term DevOps was formed by combining the words development and operations and signifies a cultural shift that bridges the gap between development and operation teams, which historically functioned in siloes

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Assignment
- => Quiz

## What you will learn :-

- => DevOps Concepts
- => Usages of DevOps
- => Practical Implementation
- => Logical ability

## Requirements :-

- => Computer with Internet Connectivity
- => Basic programming understanding

## Instructors :-

- => Ritesh Singh :
- ~

## Curriculum details :-

- => Introduction of Devops (Hindi) Day 1 :
  - ~ Introduction Preview
- => Devops Day 1 (Hindi)
- => DevOpsDay2 Why we use Linux in DevOp | Basics concept of CLI vs GUI (Hindi)
- => DevOps Day3 | Basic of Linux Command | PART 1 (Hindi)
- => Basics of Linux PART 2 (Hindi)
- => Linux Command | PART 3 (Hindi)
- => DevOps | Bash Shell Scripting | PART 3 (Hindi)
- => Introduction to Ansible | DevOps | (Hindi)
- => Introduction to Git/Github | Installation of Git | DevOps Day8 | (Hindi)
- => Git/Github | PART1 | DevOps Day9 | (Hindi)
- => Working with Git branches | PART 2 | DevOps Day10 | (Hindi)
- => Introduction to Docker | PART 1 | DevOps | (Hindi)
- => Installation of Docker | Basic command of Docker | PART 2 | DevOps Day12 | (Hindi)
- => Command of Docker | Setup Webserver in Docker Container | commit | PART 3 | DevOps Day13 | (Hindi)
- => Docker Network | PART 4 | DevOps Day14 | (Hindi)
- => Docker Volumes | PART 5 | DevOps Day15 | (Hindi)
- => Dockerfile | PART 6 | DevOps Day16 | (Hindi)
- => Docker Project Using Docker Compose Part 7 DevOps (Hindi)
- => Introduction To Jenkins | PART 1 | DevOps Day18 | (Hindi)
- => Configuring Jenkins | Introduction to Jobs in Jenkins | PART 2 | DevOps Day19 | (Hindi)
- => Build Pipeline in Jenkins| PART 3 | DevOps Day20 | (Hindi)
- => Project using Build Pipeline in Jenkins | PART 4 | DevOps Day21 | (Hindi)

# Class 9th Chemistry

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-9th-Chemistry>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. In this, the Chemistry section focuses on concepts like Matter around us, Atoms and Molecules, etc.

## Course Features :-

=> Self Paced Videos

=> Completion Certificate

## What you will learn :-

=> Matter in our surroundings

=> Is the matter around us pure?

=> Atoms and Molecules

=> Structure of the Atom

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Jayant Topnani :

~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

=> Ch1 Matter In Surroundings :

~ Matter in our Surroundings part1

~ Matter in our Surroundings part2 Preview

=> Ch2 Is Matter Around Us Pure :

~ Is Matter Around us Pure

=> Ch3 Atoms & Molecules :

~ Atoms & Molecules part1

~ Atoms and Molecules part2

=> Ch4 Structure Of Atom :

~ Structure of the Atom part1 Preview

~ Structure of the Atom part2

# Class 10 Chemistry

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10-Chemistry>

## Course Description :-

Find out how to create a balanced chemical equation and learn about chemical reactions with CBSE Class 10 Chemistry learning resources. On TopperLearning, our experts support you to understand chemistry with CBSE Class 10 Chemistry notes such as CBSE Class 10 Chemistry notes, MCQs and NCERT solutions as per the latest syllabus.

## Course Features :-

- => Self paced video session
- => Covered entire class 10th Chemistry syllabus
- => Solved questions chapter wise
- => Notes
- => Previous year solved questions

## What you will learn :-

- => Entire NCERT Class 10th Chemistry Syllabus
- => Chapter wise solution with detailed explanation

## Requirements :-

- => Computer with Internet Connectivity

## Curriculum details :-

### => CHEMICAL REACTION AND EQUATIONS :

- ~ TYPE OF CHEMICAL CHANGES in matter
- ~ CHEMICAL REACTION AND ACTIVITIES
- ~ CHEMICAL EQUATIONS
- ~ Types of chemical reactions
- ~ Effect of oxidation

### => ACID, BASES AND SALTS :

- ~ ACIDS (CLASSIFICATION, PREPARATION, PROPERTIES, USES)
- ~ BASES (CLASSIFICATION, PREPARATION, PROPERTIES, USES)
- ~ REACTIVITY OF ACIDS AND BASES
- ~ INDICATOR
- ~ STRENGTH OF ACIDS AND BASES
- ~ SALTS (CLASSIFICATION, NOMENCLATURE, PREPARATION, PROPERTIES, USES)
- ~ SOME IMPORTANT SALTS

### => METALS AND NON METALS :

- ~ METALS (OCCURRENCE, PHYSICAL AND CHEMICAL PROPERTIES, USES)
- ~ NON-METALS (OCCURRENCE, PHYSICAL AND CHEMICAL PROPERTIES, USES)
- ~ INTERACTIONS IN METALS AND NON METALS
- ~ IONIC BOND
- ~ COVALENT BOND
- ~ METALLURGY
- ~ ALLOYS
- ~ CORROSION

### => CARBON AND ITS COMPOUNDS :

- ~ OCCURRENCE, ALLOTROPES, TETRAVALENCY
- ~ VERSATILE NATURE OF CARBON
- ~ HYDROCARBONS
- ~ FUNCTIONAL GROUPS
- ~ HOMOLOGOUS SERIES
- ~ IUPAC Nomenclature
- ~ Type of chemical reaction in Organic compound
- ~ Ethanol and Ethanoic Acid
- ~ Soaps and Detergents

### => PERIODIC CLASSIFICATION OF ELEMENTS :

- ~ HISTORICAL DEVELOPMENT
- ~ MENDELEEV'S PERIODIC TABLE
- ~ MODERN PERIODIC TABLE
- ~ IMPORTANT PROPERTIES IN PERIODIC TABLE
- ~ DIAGONAL RELATIONSHIP

# Dart Language

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : DART

Course link : <https://ineuron.ai/course/Dart-Language>

## Course Description :-

Learn how to write Dart programmes from the ground up. This course is designed for those who have never programmed before. Dart is a strong and expressive language with a simple learning curve. This makes it an excellent first language. Dart provides a client-optimized language, rich and powerful frameworks, and flexible tools to help you create attractive, high-quality experiences across all screens.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Dart fundamentals
- => Data types in Dart
- => Arrays
- => Maps
- => Constants
- => Operators
- => Conditionals
- => Functions
- => Object Oriented Programming in Dart
- => Asynchronous subroutines

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction to Dart programming language :

~ Introduction to dart

=> Hello World on MAC in Dart :

~ Dart installation on MAC

=> Hello World on Windows in Dart :

~ Dart installation for WINDOWS

=> Basics of Dart :

~ Introduction to variables

~ datatypes in dart

~ array in dart

~ Maps in dart

~ Constants and operations

=> Conditionals in Dart :

~ Introduction to if and else in dart

~ Advance if else statement in dart

~ Switch and case in dart

=> Functions and loops in Dart :

- ~ *basics of functions*
- ~ *Creating a calculator in dart*
- ~ *advance functions in dart*
- ~ *While loop in dart*
- ~ *Do While loop in dart*
- ~ *Solution for assignment in dart*
- ~ *for and for in loop in dart*

=> Intermediate Dart :

- ~ *More on Arrays in dart*
- ~ *More on maps in dart*
- ~ *for each loop for map in dart*

=> OOPS in Dart :

- ~ *Classes in dart*
- ~ *Objects in dart*
- ~ *constructor in Dart*
- ~ *getters and setters in dart*
- ~ *inheritance in dart*
- ~ *Interface in dart*
- ~ *Using multiple files in classes*

=> Advance Dart :

- ~ *Generics in dart*
- ~ *Code cascading in dart*
- ~ *Custom exception handling*
- ~ *Dart libraries*
- ~ *future and async in dart*
- ~ *Web server in dart*
- ~ *Making an API request in dart*

# DLCV with NLP

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/DLCV-with-NLP>

## Course Description :-

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. It is a function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Learn Deep Learning, Transfer Learning and Neural Networks using the latest frameworks. Become a Deep Learning Guru!

## Course Features :-

- => Deep Learning
- => Natural Language processing
- => Computer Vision
- => Course Certificate
- => One to One Resume Discussion
- => Doubt Clearing session
- => Email Support
- => All 7 Days in a week Skype Support
- => Career Guidance

## What you will learn :-

- => Advance NLP with deep-learning overview.
- => TensorFlow Installation.
- => Pytorch.
- => Neural Network.
- => CNN overview
- => Advance Computer Vision Part 1.
- => Advance computer Vision Part 2.
- => ChatBot.
- => Text processing
- => Spacy.
- => NLP terminalogy.
- => RNN
- => Attention Based model.
- => Hardware Setup GPU.
- => Transfer Learning in NLP.
- => Mini NLP Project.
- => Deployment of Model and Performance tuning.
- => NLP Transfer learning project with deployment and integration with UI.
- => NLP end to end project with architecture and deployment.
- => NLP project end to end with deployment in various cloud and UI integration.
- => Computer Vision Project.

## Requirements :-

- => Dedication
- => Computer with i3 processor and internet

## Curriculum details :-

- => Introduction
- => Advance NLP with deep-learning overview :
  - ~ Computational Linguistic
  - ~ History of NLP
  - ~ Why NLP
  - ~ Use of NLP

#### => TensorFlow Installation :

- ~ *Tensorflow Installation 2.0*
- ~ *Tensorflow Installation 1.6 with virtual environment*
- ~ *TensorFlow 2.0 function*
- ~ *Tensorflow 2.0 neural network creation*
- ~ *Tensorflow 1.6 functions*
- ~ *Tensorflow 1.6 neural network and its functions*
- ~ *Keras Introduction*
- ~ *Keras in-depth with neural network creation*
- ~ *Mini project in Tensorflow*

#### => Pytorch :

- ~ *Pytorch installation*
- ~ *Pytorch functional overview*
- ~ *Pytorch neural network creation*

#### => Neural Network :

- ~ *A Simple Perception*
- ~ *Neural Network overview and its use case*
- ~ *Various Neural Network architect overview*
- ~ *Use case of Neural Network in NLP and computer vision*
- ~ *Multilayer Network*
- ~ *Loss Functions*
- ~ *The Learning Mechanism*
- ~ *Optimizers*
- ~ *Forward and Backward Propagation*
- ~ *Gradient Descent*

#### => CNN overview :

- ~ *CNN definition and various CNN based architecture*
- ~ *End to End CNN network training*
- ~ *Deployment in Azure*
- ~ *Cloud performance tuning of CNN network*

#### => Advance computer vision part 1 :

- ~ *GAN*
- ~ *Generative Model Using GAN*
- ~ *BERT*
- ~ *Semi-Supervised learning using GAN*
- ~ *Restricted Boltzmann Machine (RBM) and Autocoders*
- ~ *CNN Architectures*
- ~ *LeNet-5*
- ~ *AlexNet*
- ~ *GoogLeNet*
- ~ *VGGNet*
- ~ *ResNet*
- ~ *SSD*
- ~ *SSD lite*
- ~ *Faster R CNN*

#### => Advance computer Vision Part 2. :

- ~ *SCNN*
- ~ *Masked R-CNN*
- ~ *Xception*
- ~ *SENet*
- ~ *Facenet*
- ~ *Implementing a ResNet 34 CNN using Keras*
- ~ *Pretrained Models from Keras*
- ~ *Pretrained Models for Transfer Learning*

#### => ChatBot :

- ~ *Intents and Entities*
- ~ *Fulfillment and integration*
- ~ *Chatbot using Microsoft bot builder and LUIS, development to Telegram, Skype*
- ~ *Chatbot using Microsoft bot builder and LUIS, development to Telegram, Skype*
- ~ *Chatbot using Amazon Lex, deployment to Telegram, Skype*
- ~ *Chatbot using RASA NLU, deployment to Telegram , Skype*
- ~ *Semantic Segmentation*
- ~ *Classification and Localisation*
- ~ *TensorFlow Object Detection*
- ~ *You Only Look Once (YOLO)*

#### => Text processing :

- ~ *Importing Text*
- ~ *Web Scrapping*
- ~ *Text Processing*
- ~ *Understanding Regex*
- ~ *Text Normalisation*
- ~ *Word Count*
- ~ *Frequency Distribution*
- ~ *Text Annotation*
- ~ *Use of Anotator*
- ~ *String Tokenization*
- ~ *Annotator Creation*
- ~ *Sentence processing*
- ~ *Lemmatization in text processing*
- ~ *POS*
- ~ *Named Entity Recognition*
- ~ *Dependency Parsing in text*
- ~ *Sentimental Analysis*

=> Spacy :

- ~ Spacy Overview
- ~ Spacy function
- ~ Spacy function implementation in text processing
- ~ POS tagging, challenges and accuracy
- ~ Entities and named entity Recognition, interpolation, Language models

=> NLP Terminology :

- ~ Morphology and Diversity
- ~ Ambiguity and Paradigms
- ~ Structures and meanings
- ~ Lexical Knowledge, Network Metaphors and co-references
- ~ Lexical Ambiguity
- ~ Polysemy and homonymy
- ~ Coreference Resolution
- ~ Anaphora and cataphora resolution
- ~ Multi-sentential resolution
- ~ Humans and Ambiguity
- ~ Machines and ambiguity
- ~ Co-occurrence and distributional similarity
- ~ Similarity and relatedness
- ~ Knowledge graphs and repositories
- ~ Computational Linguistics
- ~ Word embeddings and co-occurrence vectors
- ~ Word Sim353 Dataset examples
- ~ Word2vec
- ~ Part of speech tagging

=> RNN :

- ~ Recurrent Neural Networks
- ~ Long Short Term Memory (LSTM)
- ~ Bi LSTM
- ~ GRU implementation
- ~ Building a Story writer using character level RNN

=> Attention Based model :

- ~ Seq 2 Seq
- ~ Encoders and Decoders
- ~ Attention Mechanism
- ~ Attention Neural Networks
- ~ Self Attention

=> Hardware Setup GPU :

- ~ GPU Introduction
- ~ Various type of GPU configuration
- ~ GPU provider and its pricing
- ~ Paperspace GPU setup
- ~ Running model in GPU

=> Transfer Learning in NLP :

- ~ Introduction to transformers
- ~ BERT Model
- ~ ELMo Model
- ~ GPT1 Model.
- ~ GPT2 Model
- ~ ALBERT Model
- ~ DistilBERT Model

## Project details :-

=> NLP project end to end with deployment in various cloud and UI integration :

- ~ Topic Modeling
- ~ Word sense disambiguation
- ~ Text to speech
- ~ Keyword Spotting
- ~ Document Ranking
- ~ Text Search (with Synonyms)
- ~ Language Modeling
- ~ Spam Detector
- ~ Image Captioning

=> Mini NLP project :

- ~ Machine Translation
- ~ Abstractive text summarization
- ~ Keyword spotting
- ~ Language modelling
- ~ Document summarization

=> Deployment of Model and Performance tuning :

- ~ Deep learning model deployment strategies
- ~ Deep learning project architecture
- ~ Deep learning model deployment phase
- ~ Deep learning model retraining phase
- ~ Deep learning model deployment in aws
- ~ Deep learning model deployment in azure
- ~ Deep learning model deployment in gcloud

=> NLP transfer learning Project :

- ~ Deployment and integration with ui machine translation
- ~ Question answering (like chat bot)
- ~ Sentiment analysis imdb



- ~ Text search (with synonyms)
- ~ Text classifications
- ~ Spelling corrector
- ~ Entity (person, place or brand) recognition
- ~ Text summarization
- ~ Text similarity (paraphrase)
- ~ Topic detection
- ~ Language identification
- ~ Document ranking
- ~ Fake news detection
- ~ Plagiarism checker
- ~ Text summarization extractive
- ~ Text summarization abstractive

=> NLP end to end project with architecture and deployment :

- ~ Movie review using bert
- ~ Ner using bert
- ~ Pos bert
- ~ Text generation gpt 2
- ~ Text summarization xlnet
- ~ Abstract bert
- ~ Machine Translation
- ~ Nlp text summarization custom
- ~ Keras/tensorflow
- ~ Language identification
- ~ Text classification using fast bert
- ~ Neuralcore
- ~ Detecting fake text using gltr with bert and gpt2
- ~ Fake news detector using gpt2
- ~ Python plagiarism checker type a message
- ~ Question answering

=> Computer Vision Project :

- ~ Traffic Surveillance System
- ~ Object identification
- ~ Object tracking
- ~ Object classification
- ~ Tensorflow object detection
- ~ Image to text processing
- ~ Speech to speech analysis
- ~ Vision based attendance system

# Full Stack Data Science Masters

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science-Masters>

## Course Description :-

This program teaches students how to extract insights from data using statistical and machine learning techniques, as well as data visualization and data operational skills. Students learn to work with popular data analysis tools such as Python, SQL, and machine learning frameworks, and work on hands-on projects to apply their knowledge. Overall, the course provides students with the skills to make informed decisions based on data, relevant to a wide range of industries. You will learn all the stack required to work in data science, including machine learning operations and cloud infrastructure, as well as real-time industry projects.

## Course Features :-

- => Full Stack Data Science Masters Certification
- => Job Guarantee Program
- => Self-Paced Learning
- => 150+ hours content recorded by Industry Veterans
- => 20+ hands-on industry real-time projects
- => 2 year Dashboard access
- => Doubt clearing live classes
- => Doubt clearing through mail and support team
- => Assignment in all the modules
- => Quiz in all modules
- => End-to-End Projects
- => Resume Building
- => Career Guidance
- => Interview Preparation
- => Regular Assessment
- => Job Fair & Internal Hiring
- => Mock Interview Anytime
- => Internship Portal Access
- => NeuroLabs Access

## What you will learn :-

- => Python
- => Flask
- => Numpy
- => Pandas
- => Visualization
- => Databases
- => EDA
- => Linear Algebra
- => Statistics
- => Machine Learning
- => Deep learning
- => Computer vision
- => Natural language processing

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Week 0 Course Introduction :

- ~ Welcome to the Course
- ~ Platform Overview

=> Week 1 Python Basic Building :

- ~ Python Keywords and identifiers
- ~ Comments, indentation and statements
- ~ Variables and data types in Python
- ~ Standard Input and Output
- ~ Operators
- ~ Control flow: if else elif
- ~ Control flow: while loop
- ~ Control flow: for loop
- ~ Control flow: break and continue

=> Week 2 Python Data Structures :

- ~ Strings
- ~ Lists, Lists comprehension
- ~ Tuples
- ~ Sets
- ~ Dictionary, Dictionary Comprehension

=> Week 3 Python Functions :

- ~ Python Built-in Functions.
- ~ Python User-defined Functions.
- ~ Python Recursion Functions.
- ~ Python Lambda Functions.

=> Week 4 Python Exception Handling, Logging And Debugging :

- ~ Exception Handling Using Try Catch Block
- ~ Custom Exception Handling
- ~ Logging With Python
- ~ Debugging With Python

=> Week 5 Python OOPS :

- ~ Python Objects And Classes
- ~ Python Constructors
- ~ Python Inheritance
- ~ Abstraction In Python
- ~ Polymorphism in Python
- ~ Encapsulation in Python

=> Week 6 Flask :

- ~ Flask Fundamentals
- ~ Building Rest API's

=> Week 7 Python Project With Deployment :

- ~ End To End Review Scraper Project With Deployment In Cloud
- ~ Weather App- Build A Web app that displays current weather conditions for a specific location using OpenWeatherMap API
- ~ Image web scraper- Build A Image Web Scraper which extracts images of Google

=> Milestone 1 :

- ~ Milestone 1 Test

=> Week 8 Python For Data Science- Numpy :

- ~ Numpy Basics to Advance
- ~ Key Operations using Numpy

=> Week 9 Python For Data Science- Pandas :

- ~ Pandas Basic To Advance- Dataframe And Series
- ~ Key Operations on DataFrames

=> Week 10 Python For Visualization :

- ~ Getting Started with Matplotlib
- ~ Getting Started with Seaborn

=> Milestone 2 :

- ~ Milestone 2 Test

=> Week 11 SQL-Basic to Intermediate :

- ~ Working with MySQL Using NeuroLabs
- ~ USE, DESCRIBE, SHOW TABLES
- ~ SELECT
- ~ INSERT
- ~ UPDATE & DELETE
- ~ CREATE TABLE
- ~ ALTER: ADD, MODIFY, DROP
- ~ DROP TABLE, TRUNCATE, DELETE
- ~ LIMIT, OFFSET
- ~ ORDER BY
- ~ DISTINCT

- ~ WHERE, Comparison operators, NULL
- ~ Logical Operators
- ~ Aggregate Functions: COUNT, MIN, MAX, AVG, SUM
- ~ GROUP BY
- ~ HAVING

=> Week 12 SQL- Intermediate To Advance :

- ~ Join and Natural Join
- ~ Inner, Left, Right and Outer joins
- ~ Sub Queries/Nested Queries/Inner Queries
- ~ SQL Primary And Foreign Key
- ~ SQL Function And Stored Procedures
- ~ SQL Window Function
- ~ CTE In SQL
- ~ Normalization In SQL

=> Week 13 SQL Interview Questions :

- ~ Discussing FAANG SQL Interview Questions
- ~ Discussing Other Top Product And Service Based Companies SQL Interview Questions

=> Week 14 Python With MongoDB :

- ~ MongoDB Tutorials With Various Operations- We will see how we can perform various database operations using MongoDB(No SQL)

=> Milestone 3 :

- ~ Milestone 3 Test

=> Week 15 Exploratory Data Analysis - 1 :

- ~ Analyzing Bike Sharing Trends.
- ~ Analyzing Movie Reviews Sentiment.
- ~ Customer Segmentation And Effective Cross Selling.

=> Week 16 Exploratory Data Analysis - 2 :

- ~ Analyzing Wine Types And Quality.
- ~ Analyzing Music Trends And Recommendations.
- ~ Forecasting Stock And Commodity Prices

=> Milestone 4 :

- ~ Milestone 4 Test

=> Week 17 Maths For Data Science(Linear Algebra 1) :

- ~ Linear Systems and Gaussian Elimination

In this module we will learn what a matrix is and what it represents. We will explore how a system of linear equations can be expressed via matrices.

- ~ Matrix- In this module, we will learn how to solve a linear system of equations with matrix algebra.

=> Week 18 Maths For Data Science(Linear Algebra 2) :

- ~ Projection And Least Square-In this module we will discuss projections and how they work. We will build on a foundation using 1D 2D projections and explore the concept in higher dimensions over time.

~ Determinant and Eigens-In this module we will learn how to compute the determinant of a matrix. Afterwards, Eigenvalues and Eigenvectors will be covered.

=> Week 19 Maths For Data Science(Probability) :

- ~ Important concepts in probability theory including random variables and independence

=> Week 20 Maths For Data Science(Calculus) :

- ~ Definition of a Derivative- What is a derivative? Calculate simple derivatives from the definition of a derivative.
- ~ Product and Chain Rule-Use the product and chain rules to calculate the derivatives of more complicated functions.
- ~ Using Derivatives to Graph Functions-Use where derivatives are positive and negative to help graph a function.
- ~ Finding Maximums and Minimums-Use derivatives to find the maximum and minimum values of functions.

=> Week 21 Statistics 1 :

~ Introduction & Descriptive Statistics- In this module, you will learn about the fundamentals of descriptive statistics, which include mean, median, mode, variance, and standard deviation. The module aims to demonstrate the importance of measures of central tendency and dispersion for various levels of measurement. You will gain an understanding of how these statistical tools are used to analyze and interpret data accurately. The module will cover the basics of mean, median, mode, variance, and standard deviation and provide examples of their practical applications. By the end of the module, you will be equipped with the knowledge to use these measures for data analysis effectively.

=> Week 22 Statistics 2 :

- ~ Introduction to Probability Distributions- In this module, we will cover various distributions and understand pdf, pmf and cdf

=> Week 23 Statistics 3 :

~ Hypothesis Testing-This module aims to equip you with the necessary knowledge to choose the appropriate test when analyzing data and determining their relationships. It will provide a detailed explanation of the assumptions underlying each test and teach you how to interpret the results of a hypothesis test accurately.

=> Milestone 5 :

- ~ Milestone 5 Test

=> Week 24 Feature Engineering :

- ~ Feature Selection
- ~ Handling missing values
- ~ Handling imbalanced data
- ~ Handling outliers
- ~ Encoding
- ~ Feature Scaling

=> Week 25 Machine Learning (Supervised - 1) :

- ~ AI Vs ML Vs DL Vs DS
- ~ Types Of ML Techniques
- ~ Supervised vs unsupervised and semi-supervised and reinforcement learning
- ~ Linear Regression
- ~ End To End Project With Deployment

=> Week 26 Machine Learning (Supervised - 2) :

- ~ Logistic Regression

- ~ Task- End To End Project With Deployment
- ~ Support Vector Machines
- ~ Naive Bayes
- ~ Task- End To End Project With Deployment

=> Week 27 Machine Learning (Supervised - 3) :

- ~ Decision Tree
- ~ Gradient Boosting
- ~ Xgboost
- ~ Task- End To End Project With Deployment

=> Week 28 Machine Learning (Unsupervised) :

- ~ PCA
- ~ Kmeans Clustering
- ~ Hierarchical Clustering
- ~ DbSCAN Clustering
- ~ Performance Metrics In Clustering

=> Week 29 Machine Learning (Time Series) :

- ~ Time Series Using fbprophet
- ~ Time Series Using AutoTs
- ~ Time Series Using Darts

=> Week 30 End To End ML Projects With Deployment :

- ~ Developing a Comprehensive Image Scraper with Python
- ~ Machine Learning-Based Fault Prediction for Industrial Sensors End To End Project
- ~ Developing an Advanced Review Scraper with Python and Data Visualization

=> Week 31 End To End ML Projects With Deployment :

- ~ ShipSage: Machine Learning for Smart Shipment Price Prediction
- ~ GreenVision: AI-driven Forest Cover Type Classification System
- ~ Customer Categorizer: Leveraging Machine Learning to Uncover Hidden Market Segments
- ~ PhishFinder: Machine Learning-Based Phishing Detection and Classification With Bento ML and MLFOW

=> Milestone 6 :

- ~ Milestone 6 Test

=> Week 32 Interview Preparation :

- ~ Resume Discussion And Resume Preparation
- ~ Python Interview Questions Discussion
- ~ Stats Interview Questions Discussion
- ~ Machine Learning Interview Questions Discussion
- ~ How To Explain End to Projects To Interviewer

=> Week 33 Deep Learning ANN :

- ~ Artificial Neural Network Working
- ~ Back Propagation In ANN
- ~ Chain Rule Of Derivatives
- ~ Vanishing Gradient Problem
- ~ Exploding Gradient Problem

=> Week 34 Deep Learning Fundamentals :

- ~ Different Activation functions
- ~ Different types of Loss Function
- ~ Different types Of Optimizers
- ~ Weight Initialization Techniques
- ~ Drop Out Layer
- ~ Batch Normalization

=> Week 35 Deep Learning Frameworks :

- ~ Working With Tensorflow Keras
- ~ Working With Pytorch

=> Week 36 Deep Learning (Computer Vision Fundamentals) :

- ~ CNN Fundamentals
- ~ Lenet-5 Variants With Research Paper And Practical
- ~ Alexnet Variants With Research Paper And Practical

=> Week 37 Deep Learning (Image Classification & Transfer Learning) :

- ~ GoogLeNet Variants With Research Paper And Practical
- ~ VggNet Variants With Research Paper And Practical
- ~ ResNet Variants With Research Paper And Practical

=> Week 38 Deep Learning (Computer Vision - Object Detection) :

- ~ Object Detection(In this module we will discuss various advanced algorithms which will help us perform object detection )

=> Week 39 Deep Learning (Computer Vision - Segmentation Tracking) :

- ~ Image Segmentation(In this module we will discuss various advanced algorithms which will help us perform image segmentation)
- ~ Object Tracking (In this module we will discuss various advanced algorithms which will help us perform object tracking)

=> Week 40 Deep Learning (NLP - 1) :

- ~ NLP With Machine Learning- In this module, we will discuss how we can apply different NLP techniques in text and work with ML algorithms
- ~ NLP With Recurrent Neural Network and Its variants

=> Week 41 Deep Learning (NLP - 2) :

- ~ NLP with Sequence Models- In this module, we will discuss about various Sequence Models in Deep Learning
- ~ NLP With Attention Models- In this module, we will discuss Transformers,BERT, and GPT models

=> Week 42 End To End Deep Learning Projects With Deployment :

- ~ Developing an Audio Classification System for Accurate Speech Recognition
- ~ Developing a Robust Helmet Detection System using Computer Vision

=> Week 43 End To End Deep Learning Projects With Deployment :

- ~ Developing an AI-Driven Text Summarization System with Deep Learning Techniques

~ *Developing an AI Model for Automated Lungs Disease Diagnosis Using Bento ML and MLFLOW*

=> Week 44 End To End Deep Learning Projects With Deployment :

~ *Developing a High-Quality Text-to-Speech System with Advanced NLP Techniques*

~ *AI-Enabled Object Detection for Improved Industrial Safety*

=> Milestone 7 :

~ *Milestone 7 Test*

=> Week 45 Big Data - Hadoop :

~ *Hadoop*

=> Week 46 Big Data - Spark :

~ *Spark*

=> Milestone 8 :

~ *Milestone 8 Test*

=> Week 47 Data Analytics - PowerBi :

~ *PowerBI*

=> Week 48 Data Analytics - Tableau :

~ *Tableau*

=> Milestone 9 :

~ *Milestone 9 Test*

=> Week 49 - 52 Interview Preparation :

~ *Resume Discussion And Resume Preparation*

~ *Computer Vision Interview Preparation*

~ *NLP Interview Preparation*

~ *Internship Tasks For Deep Learning*

~ *Mock Interview Sessions*

~ *Industry Expert Talks*

~ *How To Build Analytical Thinking*

~ *Discussing Different Project Architectures*

~ *Project Building End to END*

# Image Processing

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Image-Processing>

## Course Description :-

In this hands-on course, You will learn how to filter, change, and edit images. You will also learn how to use OpenCV to perform various image processing tasks with hands-on practical experience. Students who complete this course will be able to apply what they have learned in this course to a variety of fields, including machine learning and artificial intelligence, machine and robotic vision, space and medical image analysis, and many more.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with tech
- => Real-time project
- => Live class recording
- => Doubt clearing
- => Assignment in all the module
- => Quiz in every module
- => Career Counselling
- => Completion certificate

## What you will learn :-

- => Image processing
- => OpenCV
- => Scikit
- => Color space
- => Creating Basic Drawings
- => Advanced OpenCV
- => Projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

=> Course Introduction :

- ~ Welcome to image processing course
- ~ What you will learn from this course
- ~ Course pre-requisites
- ~ What is image processing?
- ~ Who is this course for?
- ~ What you will get from this course?
- ~ How to get access to course materials?
- ~ What career path you can follow after completion of this course?

=> Introduction to Image Processing :

- ~ What do you mean by image processing?
- ~ Why image processing is used?
- ~ What are images?
- ~ Fundamentals of images
- ~ What do you mean by pixel?
- ~ Image resolution
- ~ PPI and DPI
- ~ What is a bitmap image?

## => Compression :

- ~ *What is compression?*
- ~ *How compression is helpful?*
- ~ *Lossless compression*
- ~ *Lossy compression*
- ~ *Different format of images*

## => Color Spaces :

- ~ *What is a color spcaes*
- ~ *RBG color space explanation*
- ~ *XYZ color space*
- ~ *HSV/HSL*
- ~ *LAB color space*

## => Scikit image :

- ~ *Scikit image introduction*
- ~ *Uploading and Viewing an Image*
- ~ *Getting Image Resolution*
- ~ *Looking at Pixel Values*
- ~ *Converting Color Space*
- ~ *Saving an Image*

## => Creating Basic Drawings :

- ~ *Lines*
- ~ *Rectangles*
- ~ *Circles*
- ~ *Bezier Curve*
- ~ *Doing Gamma Correction*
- ~ *Rotating, Shifting, and Scaling Images*

## => Advanced OpenCV :

- ~ *Introduction to OpenCV*
- ~ *Blending two images*
- ~ *Changing brightness*
- ~ *Changing contrast*
- ~ *Adding Text to Images*
- ~ *Smoothing Images*
- ~ *Median Filter*
- ~ *Gaussian Filter*
- ~ *Bilateral Filter*
- ~ *Resizing images*
- ~ *Image Thresholding*
- ~ *Histogram Equalization*

## => Projects :

- ~ *Project 1: Creating a HDR with OpenCV*
- ~ *Project 2: Removing Green screen from an image with different background*

## => Summary :

- ~ *Course Outro*
- ~ *Future Scope of Image processing*



# Data Analysis Using Python

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Data-Analysis-Using-Python>

## Course Description :-

In this course, you will learn to extract insights from the data. This course is designed to help you deal with data analysis and data manipulation using the popular python library Pandas. You will learn powerful functions to present the facts from data in the most straightforward and accurate way.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => One to One Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to Data Analysis
- => Importance of Data
- => Integrated development environment
- => Python Packages
- => Python Libraries
- => File Formats
- => Pandas Library
- => Data Cleaning
- => Data Manipulation
- => Pandas Functions
- => Feature engineering

## Requirements :-

- => Interest to learn
- => Dedication
- => System with good internet connection

## Curriculum details :-

- => Introduction to the Course :
  - ~ Course Introduction
  - ~ Who is this course for?
  - ~ Course overview & course outcome
  - ~ What is Data? How data will be collected?
  - ~ Why Data is important?
  - ~ What is DataFrame?
  - ~ Why DataFrame is used?
  - ~ What is Analysis?
  - ~ Why Analysis is used?
  - ~ What is Data Analysis?
  - ~ Why Data Analysis is used?
- => Assignment 1 :
  - ~ Is it possible to create and innovate things without data?
- => System setup :
  - ~ What is IDE?
  - ~ Why IDE is used?
  - ~ Advantages of using an IDE?
  - ~ Google Colab

=> PyPI :

- ~ What is PyPI?
- ~ What is pip?
- ~ Installing your first library using pip

=> Packages :

- ~ What is a Package?
- ~ How to install Python Packages

=> Library :

- ~ What is a library?
- ~ What is the difference between package and library?

=> Open Source :

- ~ What do you mean by open-source community?
- ~ What is an open-source package or a library?
- ~ Why is it important to do open-source contribution

=> File Formats :

- ~ What is a file?
- ~ What are the types of files?
- ~ .csv
- ~ .txt
- ~ .json
- ~ .xlsx

=> Pandas :

- ~ What is pandas library?
- ~ Alternatives of pandas
- ~ Why pandas use ?

=> Cleaning Data Using Pandas :

- ~ What is Data cleaning?
- ~ Why it is necessary to perform Data cleaning?
- ~ Performing Data cleaning using pandas
- ~ Removing unnecessary columns in the DataFrame
- ~ Skipping unnecessary rows in a CSV file
- ~ Changing the index of a DataFrame.
- ~ Renaming columns to a more recognizable set of labels.

=> Pandas Functions :

- ~ What do you mean by functions?
- ~ What are Pandas functions?
- ~ Important Pandas functions
- ~ Practical: read\_csv()
- ~ Practical: head()
- ~ Practical: describe()
- ~ Practical: memory\_usage()
- ~ Practical: astype()
- ~ Practical: loc[:]
- ~ Practical: value\_counts()
- ~ Practical: groupby()

=> Assignment 2 :

- ~ Download any dataset of your choice and try to apply these pandas functions on your own.
- ~ Try to find out more functions in pandas and try to implement them.

=> Feature Engineering :

- ~ What do you mean by feature?
- ~ What is Feature Engineering?
- ~ What do you mean by Feature Transformations?
- ~ Practical: Transforming columns into the same scale
- ~ What do you mean by Feature Extraction?
- ~ Practical: Extracting year from your birthdate
- ~ What do you mean by Feature selection?
- ~ Practical: selecting the important features from the dataset

=> Project :

- ~ Take any dataset and perform data analysis using pandas

=> Course Summary :

- ~ Course outro
- ~ Future learning path
- ~ Order by clause

# C Programming

---

Topic Name : PROGRAMMING

Sub-topic Name : C

Course link : <https://ineuron.ai/course/C-Programming>

## Course Description :-

This course is designed mostly for novice programmers who may not have any prior programming language knowledge. From the most fundamental to the most sophisticated subjects, there is something for everyone. Step by step, from a simple to a sophisticated programme. This course should be taken if one want to pursue a career as a programmer. C programming is widely regarded as the cornerstone for all computer languages. If one is comfortable with C, they may go on to other languages such as PHP, C++, or Java.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => C programming basics
- => C data types
- => C data structures
- => Input/Output in C
- => Control Flow
- => Loops
- => Functions
- => Structs
- => Memory management
- => Macros

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Before we Begin :

- ~ Introduction to the course.mp4
- ~ C environment on Xcode for MAC users.mp4
- ~ Setting C environment on WINDOWS.mp4
- ~ Running C file in WINDOWS.mp4
- ~ Running C file from terminal in MAC.mp4
- ~ What just happened in the source code.mp4

=> Exercise we begin :

- ~ How to use exercise file on a MAC.mp4
- ~ How to use exercise file on a WINDOWS.mp4

=> Basics Theory while writing code :

- ~ Programming 101.mp4
- ~ Solving Problems for MAC.mp4
- ~ Solving Problems for WINDOWS.mp4
- ~ Explaining Data Types in C.mp4
- ~ Variables and common associated problems.mp4
- ~ Constants and their importance.mp4
- ~ Format Specifiers.mp4
- ~ Character Constants.mp4

=> Operations & Decision :

- ~ *Performing Arithmetic operations in C.mp4*
- ~ *Relations and Logics.mp4*
- ~ *Introduction to decisions aka if else.mp4*
- ~ *Switch and various cases.mp4*
- ~ *Read number and print a sum.mp4*
- ~ *Biggest of three.mp4*
- ~ *ODD or EVEN.mp4*

=> Loops and Functions :

- ~ *Looping basics with While loop.mp4*
- ~ *Do While loop in C.mp4*
- ~ *For Loop in C.mp4*
- ~ *Break keyword usage.mp4*
- ~ *Continue keyword in C.mp4*
- ~ *Type Casting in C.mp4*
- ~ *Custom Functions in C.mp4*
- ~ *Value call vs reference call.mp4*
- ~ *Scope of a variable.mp4*

=> Coding is Fun :

- ~ *Sum of three digits.mp4*
- ~ *Armstrong Number.mp4*
- ~ *Odds in Hundred.mp4*
- ~ *Fibonacci Series.mp4*
- ~ *The number Pyramid.mp4*
- ~ *The character pyramid.mp4*
- ~ *Reverse Number pyramid.mp4*

=> Advance C Programming :

- ~ *Introduction to Array datatype.mp4*
- ~ *Advance part in Array.mp4*
- ~ *Details about Strings.mp4*
- ~ *Pointers as simple as possible.mp4*
- ~ *Structures.mp4*
- ~ *Saving memory in Bit Fields.mp4*
- ~ *Custom Defined DataTypes.mp4*
- ~ *Macros in C.mp4*
- ~ *Error Handling.mp4*
- ~ *File Handling in C.mp4*
- ~ *Command Line Arguments.mp4*
- ~ *Recursion.mp4*

=> Fun to code applications :

- ~ *Upper to lower case converter.mp4*
- ~ *Case Converter.mp4*
- ~ *Highest in Array.mp4*
- ~ *Linear Search in Array.mp4*

# Prisma Backend Development

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Prisma-Backend-Development>

## Course Description :-

This is a crash course on Prisma. Not an in-depth series on Prisma. This will get you up and running with Prisma in no time.

## Course Features :-

- => Course material
- => Course resources
- => On-demand recorded videos
- => Practical exercises
- => Course completion certificate

## What you will learn :-

- => Prisma
- => Prisma with MongoDB
- => Prisma Backend Development

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Prisma development :

- ~ Learn backend development with Prisma and MongoDB
- ~ Learn backend development with Prisma and MongoDB | part 2
- ~ Learn backend development with Prisma and MongoDB | part 3

# Machine Learning Interview Preparation

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/Machine-Learning-Interview-Preparation>

## Course Description :-

This course is designed mostly for Machine Learning Interview Preparation.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => ML Theoretical Test
- => ML Practical Test
- => ML Aptitude Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

=> Machine Learning Interview Preparation :

- ~ ML Interview Test 1
- ~ ML Interview Test 2
- ~ ML Interview Test 3
- ~ ML Interview Test 4
- ~ ML Interview Test 5
- ~ ML Interview Test 6
- ~ ML Interview Test 7
- ~ ML Interview Test 8
- ~ ML Interview Test 9
- ~ ML Interview Test 10
- ~ ML Interview Test 11
- ~ ML Interview Test 12
- ~ ML Interview Test 13
- ~ ML Interview Test 14
- ~ ML Interview Test 15
- ~ ML Interview Test 16
- ~ ML Interview Test 17
- ~ ML Interview Test 18
- ~ ML Interview Test 19
- ~ ML Interview Test 20

# Full Stack Data Science Bootcamp

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science-Bootcamp>

## Course Description :-

Full-stack data science course is a live mentor-led job guaranteed certification program with a full-time one-year internship provided by iNeuron, in this course you will learn the entire stack required to work in the data science, data analytics, and big data domain, including machine learning, deep learning, computer vision NLP and Big data including MLOps and cloud infrastructure, along with real-time industry projects and product development with the iNeuron product development team, which will enable you to contribute on various levels.

## Course Features :-

- => Full stack Data Science masters certification
- => Job guarantee otherwise refund
- => One year of internship Anytime
- => 1:1 Personalized Mentorship
- => Revision Classes
- => Online Instructor-led learning: Live teaching by instructors
- => 56 + hands-on industry real-time projects.
- => 500 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Doubt clearing through mail and skype support team
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building Anytime
- => Career guidance Anytime
- => Interview Preparation Anytime
- => Regular assessment
- => Job Fair and Internal Hiring
- => Mock Interview Anytime

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => ML ops
- => Cloud
- => Architecture
- => Domain wise project
- => Databases
- => Negotiations skills
- => Mock interview
- => Interview preparation

=> Resume building after every module

=> Industry grade projects

### Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to learn

### Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

### Curriculum details :-

=> Python Basics :

~ Python Introduction, Installation and Setup Preview  
~ Python Basics & Conditionals Preview  
~ Conditionals & Loops  
~ Working with Loops  
~ Working with Strings & Lists  
~ List manipulation  
~ Tuple, Set & Dictionary  
~ Working with Functions  
~ Functions, Generators & File Handling  
~ Logging and debugging  
~ Modules and Exception

=> OOPS :

~ OOPs, Classes & Objects Preview  
~ OOPS, Abstraction & Inheritance Preview  
~ Inheritance, Polymorphism & Intro to Databases

=> Databases :

~ Working with SQL & Python Preview  
~ SQL Continued, MongoDB installation & Working with MongoDB  
~ Working with Cassandra & Python

=> Pandas :

~ Introduction to Pandas Preview  
~ Pandas Basics  
~ Pandas Data Manipulation  
~ Working with Pandas

=> Numpy :

~ Introduction to Numpy Preview

=> Matplotlib :

~ Working with Pandas & Matplotlib

=> Plotly :

~ Working with Plotly

=> Seaborn :

~ Working with Seaborn

=> EDA :

~ EDA

=> Web Frameworks :

~ Rest API, Flask & Working with Postman Preview  
~ Working with Flask & Debugging Calculator Application

=> Python Projects with Deployment :

~ Project Discussion Review Scraper with Deployment on Heroku, AWS and Azure Preview  
~ Project Discussion Advance Review Scraper

=> Stats :

~ Different types of Statistics Preview  
~ Population vs Sample Preview  
~ Mean, Median and Mode  
~ Variance, Standard Deviation  
~ Sample Variance why  $n-1$   
~ Standard Deviation  
~ Variables  
~ Random Variables  
~ Percentiles & quartiles  
~ 5 number summary



- ~ Histograms
- ~ Gaussian - Normal distribution
- ~ Standard Normal distribution
- ~ Application Of Zscore
- ~ Basics Of Probability
- ~ Addition Rule In Probability
- ~ Multiplication rule in probability
- ~ Permutation
- ~ Combination
- ~ Log Normal Distribution
- ~ Central Limit theorem
- ~ Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median And Mode
- ~ Covariance
- ~ Pearson And Spearman Rank Correlation
- ~ What is P Value
- ~ What is Confidence Intervals
- ~ How To Perform Hypothesis Testing - Confidence IntervalZ Test Statistics Derive Conclusion
- ~ Hypothesis testing part 2
- ~ Hypothesis testing part 3
- ~ Finalizing statistics

#### => Machine Learning :

- ~ Introduction to Machine learning Preview
- ~ Linear Regression Preview
- ~ Linear Regression live coding demonstration part-1
- ~ Linear Regression live coding demonstration part-2
- ~ Project Admission Prediction, Lasso, Ridge & Elastic Net
- ~ Project deployment in Heroku, Azure & AWS
- ~ Logistic Regression
- ~ Logistic Regression implementation
- ~ Decision Tree
- ~ Decision Tree Part 2 , Ensemble Tech, Random Forest & Boosting
- ~ KNN and SVM
- ~ Decision Tree Practical Implementation
- ~ Decision Tree Live Coding & Grid Search
- ~ Grid Search, Bagging Classifier & Random Forest
- ~ KNN, SVC, SVR & Stacking
- ~ Clustering
- ~ Clustering and PCA
- ~ PCA practical, DBSCAN and Naive Bayes
- ~ XG Boost, NLTK & TF-IDF

#### => ML Projects :

- ~ Detailed Project Report explanation Preview
- ~ Project :- Wafer Fault Detection Part 1 Preview
- ~ Project :- Wafer Fault Detection Part 2
- ~ Deployment in Heroku using docker and circleci

#### => ML Project 1 :- Fault detection in wafers based on sensor data :

- ~ Introduction Preview
- ~ The problem statement and Data Description
- ~ The Application Flow
- ~ Ingestion and Validation Part1
- ~ Validation Part2
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 2 :- Cement Strength Prediction :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code Intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 3 :- Credit Card Defaulters :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Deployment

#### => ML Project 4 :- Forest Cover :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation

- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 5 :- Income Prediction :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 6 :- Insurance Fraud Detection :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code Intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment
- ~ The Problem Statement and Data Description

#### => ML Project 7 :- Mushroom Classification :

- ~ Introduction Preview
- ~ The Application Flow
- ~ Code Intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Predictions
- ~ Deployment

#### => ML Project 8 :- Phishing Classifier :

- ~ Introduction Preview
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 9 :- Thyroid Detection :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operation
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => ML Project 10 :- Visibility Climate :

- ~ Introduction Preview
- ~ The Problem Statement and Data Description
- ~ The Application Flow
- ~ Code intro and Logging
- ~ Validations and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Clustering
- ~ Model Selection and Tuning
- ~ Prediction
- ~ Deployment

#### => Time Series :

- ~ Arima, Sarima, Auto Arima Preview
- ~ Time series using RNN LSTM, Prediction of NIFTY stock price
- ~ Time series using RNN LSTM, Prediction of NIFTY stock price

#### => DL ANN - Introduction :

- ~ Introduction to Deep Learning Preview

- ~ Importance of Deep learning Preview
- ~ Why you should study Deep Learning? (Motivation)
- ~ ANN vs BNN
- ~ The first Artificial Neuron

#### => DL ANN - Perceptron :

- ~ Overview of Perceptron Preview
- ~ More about Perceptron
- ~ Perceptron implementation using python - 1
- ~ Perceptron implementation using python - 2
- ~ Perceptron implementation using python - 3
- ~ Perceptron implementation using python - 4
- ~ Perceptron implementation using python - 5
- ~ Perceptron implementation using python - 6
- ~ Perceptron implementation using python - 7
- ~ Python scripting & modular coding for Perceptron
- ~ Python logging basics and docstrings

#### => DL ANN -1 :

- ~ Multilayer Perceptron Preview
- ~ Forward propagation Preview
- ~ Why we need Activation function?
- ~ ANN implementation using tf.keras - 1
- ~ ANN implementation using tf.keras - 2
- ~ ANN implementation using tf.keras - 3
- ~ ANN implementation using tf.keras - 4
- ~ ANN with Callbacks | Tensorboard | Early Stopping | Model Checkpointing

#### => DL ANN - 2 :

- ~ Vector Preview
- ~ Differentiation Preview
- ~ Partial differentiation
- ~ Maxima and minima concept
- ~ Gradient descent basics
- ~ In-depth understanding of Gradient descent with mathematical proof

#### => DL ANN - 3 :

- ~ Chain rule Preview
- ~ Back propagation

#### => DL ANN - 4 :

- ~ General problems in training Neural Networks Preview
- ~ Vanishing and Exploding gradients
- ~ Activation Function Basics
- ~ Weight initialization
- ~ Activation Functions - 1
- ~ Activation functions - 2
- ~ Activation functions - 3
- ~ Transfer learning
- ~ Batch normalization -1
- ~ Batch normalization -2
- ~ Batch normalization -3

#### => DL ANN - 5 :

- ~ Introduction to fast optimizers Preview
- ~ Momentum optimization
- ~ NAG
- ~ Loss functions
- ~ Regularization
- ~ Dropout

#### => Computer Vision - Introduction :

- ~ Introduction to Course
- ~ Course Overview
- ~ Installing Anaconda, Pycharm & Postman Preview
- ~ Working with Conda Envs Preview
- ~ Pycharm Introduction
- ~ Pycharm with Conda
- ~ Pycharm with venv
- ~ Pycharm with Pipenv

#### => Computer Vision - CNN Foundations :

- ~ Why CNN? Building an Intuition for CNN Preview
- ~ CNN, Kernels, Channels, Feature Maps, Stride, Padding
- ~ Receptive Fields, Image Output Dimensionality Calculations, MNIST Dataset Explorations with CNN
- ~ MNIST CNN Intuition, Tensorspace.js, CNN Explained, CIFAR 10 Dataset Explorations with CNN
- ~ Dropout & Custom Image Classification Dog Cat Dataset
- ~ Deployment in Heroku, AWS, Azure

#### => Computer Vision - CNN Architectures :

- ~ LeNet-5 Preview
- ~ LeNet-5 Practical
- ~ AlexNet
- ~ AlexNet Practical
- ~ VGGNet
- ~ VGG16 Practical
- ~ Inception
- ~ Inception Practical
- ~ ResNet
- ~ Resnet Practical

#### => Computer Vision - Data Augmentation :

- ~ *What is Data Augmentation? Preview*
- ~ *Benefits of Data Augmentation*
- ~ *Exploring Papers like RICAP, Random Erasing, Cutout*
- ~ *Exploring Augmentor*
- ~ *Exploring Roboflow*

#### => Computer Vision - Object Detection Basics :

- ~ *What is Object Detection? Preview*
- ~ *Competitions for Object Detection Preview*
- ~ *Bounding Boxes*
- ~ *Bounding Box Regression*
- ~ *Intersection over Union (IoU)*
- ~ *Precision & Recall*
- ~ *What is Average Precision?*

#### => Computer Vision - Object Detection Architectures :

- ~ *Object Detection Family Preview*
- ~ *RCNN Preview*
- ~ *RCNN Network Architecture*
- ~ *Cons of RCNN*
- ~ *FAST RCNN*
- ~ *FAST RCNN Network Architecture*
- ~ *Cons of FAST RCNN*
- ~ *FASTER RCNN*
- ~ *FASTER RCNN Network Architecture*
- ~ *YOLO*
- ~ *YOLO Architecture*
- ~ *YOLO Limitations*

#### => Computer Vision - Practicals Object Detection using Tensorflow 1.x :

- ~ *Introduction to TFOD1.x Preview*
- ~ *Using Google Colab with Google Drive Preview*
- ~ *Installation of Libraries in Colab*
- ~ *TFOD1.x Setup in Colab*
- ~ *Visiting the Model Zoo*
- ~ *Inferencing in Colab*
- ~ *Inferencing in Local*
- ~ *Important Configurations Files*
- ~ *Webcam Testing*

#### => Computer Vision - Practicals Training a Custom Cards Detector using Tensorflow1.x :

- ~ *Custom Model Training in TFOD1.x Preview*
- ~ *Our Custom Dataset*
- ~ *Doing Annotations or labeling data*
- ~ *Selection of Pretrained Model from Model Zoo*
- ~ *Files Setup for Training*
- ~ *Let's start Training in Colab*
- ~ *Export Frozen Inference Graph*
- ~ *Inferencing with our trained model in Colab*
- ~ *Training in Local*
- ~ *Inferencing with our trained model in Local*

#### => Computer Vision - Practicals Creating an Cards Detector Web App with TFOD1 :

- ~ *Code Understanding Preview*
- ~ *WebApp Workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*
- ~ *Debugging our Application*

#### => Computer Vision - Practicals Object Detection using Tensorflow 2.x :

- ~ *Introduction to TFOD2.x Preview*
- ~ *Using the Default Colab Notebook Preview*
- ~ *Google Colab & Drive Setup*
- ~ *Visiting TFOD2.x Model Garden*
- ~ *Inference using Pretrained Model*
- ~ *Inferencing in Local with a pretrained model*

#### => Computer Vision - Practicals Training a Custom Chess Piece Detector using Tensorflow2 :

- ~ *Custom Model training in TFOD2.x Preview*
- ~ *Our Custom Dataset TF2 Preview*
- ~ *File Setup for Training*
- ~ *Let's start Training*
- ~ *Let's start Training*
- ~ *Stop Training or resume Training*
- ~ *Evaluating the trained model*
- ~ *Convert CKPT to Saved Model*
- ~ *Inferencing using the Custom Trained Model in Colab*
- ~ *Inferencing using the Custom Trained Model in Local PC*

#### => Computer Vision - Practicals Creating an Chess Piece Detector Web App with TFOD2 :

- ~ *Creating a Pycharm project & Environment Setup TF2 Preview*
- ~ *Application Workflow*
- ~ *Code understanding*
- ~ *Testing our App with Postman*
- ~ *Debugging our Application*

#### => Computer Vision - Practicals Object Detection using Detectron2 :

- ~ *Introduction to Detectron2 Preview*
- ~ *Detectron2 Colab Setup*

#### => Computer Vision - Practicals Training a Custom Detector using Detectron2 :

- ~ *Detectron2 Custom Training Preview*

- ~ *Exploring the Dataset*
- ~ *Registering Dataset for Training*
- ~ *Let's start Training*
- ~ *Inferencing using the Custom Trained Model in Colab*
- ~ *Evaluating the Model*

=> Computer Vision - Practicals Creating an Custom Detector Web App with Detectron2 :

- ~ *Creating a Pycharm project & Environment Setup Detectron2 Preview*
- ~ *Application Workflow*
- ~ *Code understanding*
- ~ *Testing our App with Postman*
- ~ *Debugging our Application*

=> Computer Vision - Practicals Object Detection using YoloV5 :

- ~ *Introduction to YoloV5 Preview*
- ~ *YoloV5 Colab Setup*
- ~ *Inferencing using Pre Trained Model*

=> Computer Vision - Practicals Training a Custom Warehouse Apparel Detector using YoloV5 :

- ~ *Custom Training with YoloV5 Preview*
- ~ *Exploring the Dataset*
- ~ *Doing Annotations or labeling data*
- ~ *Setting up Google Colab & Drive*
- ~ *Let's start Training*
- ~ *Inferencing using the Custom Trained Model in Colab*

=> Computer Vision - Practicals Creating an Warehouse Apparel Detector Web App with YOLOV5 :

- ~ *Creating a Pycharm project & Environment Setup Yolo Preview*
- ~ *Application Workflow*
- ~ *Code understanding*
- ~ *Testing our App with Postman*
- ~ *Debugging our Application*

=> Computer Vision - Image Segmentation :

- ~ *Segmentation Introduction Preview*
- ~ *From Bounding Box to Polygon Masks Preview*
- ~ *What is Image Segmentation?*
- ~ *Types of Segmentation*
- ~ *MASKRCNN*
- ~ *MASK RCNN Architecture*

=> Computer Vision - MASK RCNN Practicals with TFOD :

- ~ *Segmentation with TFOD1.x Preview*
- ~ *Local Setup MASKRCNN*
- ~ *Exploring the Dataset*
- ~ *Data Annotation*
- ~ *Model Selection*
- ~ *Files Setup for Training*
- ~ *Model Training*
- ~ *Export Frozen Inference Graph*
- ~ *Model Prediction*

=> Computer Vision - MASKRCNN practical with Detectron2 :

- ~ *Introduction to Detectron2 Preview*
- ~ *Data Preparation*
- ~ *Setup for Training*
- ~ *Let's start Training*
- ~ *Inferencing using the Custom Trained Model in Colab*
- ~ *Evaluating the Model*

=> Computer Vision - Face Recognition Project :

- ~ *Introduction to Project Preview*
- ~ *Requirement Gathering*
- ~ *Techstack Selection*
- ~ *Project Installation*
- ~ *Project Demo Preview*
- ~ *Project Workflow*
- ~ *Core Components of the Application*
- ~ *Data Collection Module*
- ~ *Generate Face Embeddings*
- ~ *Training Face Recognition Module*
- ~ *Prediction Pipeline*
- ~ *Entry point of the Application*
- ~ *Application Workflow*
- ~ *Debugging our Application*

=> Computer Vision - Object Tracking Project :

- ~ *Object Tracking project*
- ~ *Project Installation Tracking*
- ~ *Project Demo Preview*
- ~ *Code Understanding*

=> Computer Vision - GANS :

- ~ *Introduction to GANS Preview*
- ~ *GAN Architecture*
- ~ *GAN PRACTICALS Implementation*

=> Computer Vision Project - Fashion Apparel Detection :

- ~ *Introduction to Fashion Apparel Detection project Preview*
- ~ *Requirement Gathering*
- ~ *Techstack Selection*
- ~ *Detailed Project Workflow*

- ~ Data Collection
- ~ Data Preparation
- ~ Data Augmentation
- ~ Data Annotations

=> Computer Vision Project - Image TO Text OCR :

- ~ Introduction to Project Preview
- ~ Project Installation OCR
- ~ Project Demo

=> Computer Vision Project - Shredder System :

- ~ Introduction to Shredder Systems Preview
- ~ Requirement Gathering
- ~ Techstack Selection
- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation
- ~ Model Selection from Zoo
- ~ Model Training

=> Computer Vision Project - Automatic Number plate Recognition with TFOD1.x :

- ~ Introduction to ANPR Project Preview
- ~ Requirement Gathering
- ~ Tech Stack Selection
- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation

=> NLP Overview :

- ~ NLP Overview Preview
- ~ NLP very basic

=> NLP Word Embeddings :

- ~ TFIDF Preview
- ~ Word Embeddings Part-1
- ~ Word Embeddings Part-2

=> NLP RNN :

- ~ RNN basic Preview
- ~ RNN Implementation

=> NLP Project:- Text to Speech :

- ~ Introduction Preview
- ~ Project Setup Text to Speech
- ~ Project Demo

=> NLP Project:- Speech To Text :

- ~ Introduction Preview
- ~ Project Setup Speech To Text
- ~ Project Demo

=> NLP Project:- Spell Corrector :

- ~ Introduction Preview
- ~ Project Setup Spell Corrector
- ~ Project Demo

=> BigData - Introduction to Distributed Systems - Hadoop and MapReduce :

- ~ Big Data Engineering Introduction

=> BigData - Hive :

- ~ Apache hive Preview

=> BigData - NoSQL and Hbase :

- ~ Big Data HBase
- ~ Hbase hands On

=> BigData - Spark :

- ~ Spark - Introduction Preview
- ~ Big Data Engineering using PySpark- RDDs
- ~ Spark hands on - RDD
- ~ Big Data Engineering using PySpark- Shared Vars , Coalesce Repartition
- ~ Spark hands on - Dataframe

=> BigData - Spark ML :

- ~ Big Data Engineering using PySpark- MLLib
- ~ Spark hands On - Spark ML Lib

=> BigData - Spark Streaming :

- ~ Big Data Engineering using PySpark- Streaming Part 1 Preview
- ~ Big Data Engineering using PySpark- Streaming Part 2
- ~ Spark hands On - Spark Streaming

=> BigData - Kafka :

- ~ Big Data Kafka Preview
- ~ Big Data Kafka Hands on

=> Basic Charts in Power BI :

- ~ 2.0 Basic Charts in Power BI Desktop Preview
- ~ 2.1 Column Chart in Power BI Preview
- ~ 2.2 Stacked Column Chart in Power BI
- ~ 2.3 Pie Chart in Power BI

=> Working with Maps :

- ~ 3.1 Creating a Map in Power BI Preview
- ~ 3.2 Filled Map
- ~ 3.3 Map with Pie Chart
- ~ 3.4 Formatting in Map

=> Tables and Matrix in Power BI :

- ~ 4.0 Table and Matrix in Power BI Preview
- ~ 4.1 Creating a Table in Power BI
- ~ 4.2 Formatting a Table

=> Introduction to tableau :

- ~ Tableau Introduction Preview
- ~ Download and Install Tableau
- ~ Tableau Vs Excel

=> SQL :

- ~ Database Architecture Preview
- ~ Introduction to SQL Preview
- ~ Constraints
- ~ Joins
- ~ Import Export
- ~ Aggregate Functions
- ~ Order by, Having & Limit Clause
- ~ String Functions
- ~ Datetime functions
- ~ Nested Queries
- ~ Views

=> Excel :

- ~ Introduction to Excel Preview
- ~ Pre-defined functions Preview
- ~ Datetime Functions
- ~ String functions
- ~ Mathematical functions
- ~ Lookup

=> Chatbot - Google Dialog Flow :

- ~ What is Chatbot? Preview
- ~ Why Chatbot? Preview
- ~ Types of Chatbot
- ~ Use of Chatbot
- ~ Examples of chatbot
- ~ Dialogflow - Inline editor
- ~ Create Intent and Entities
- ~ Food order Intent

=> Interview Preparation - Interview Questions Discussion :

- ~ Interview Question Discussion Preview
- ~ Resume Discussion

=> Interview Preparation - Project Discussion :

- ~ Vision-Based Attendance System Preview

=> Interview Preparation - Interview Questions Discussions :

- ~ Interview Question Discussion - 1 Preview

=> Interview Preparation - General Discussion :

- ~ Discussion Session - 1 Preview
- ~ Discussion Session - 2

# Scratch Programming for kids

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Scratch-Programming-for-kids>

## Course Description :-

This course will provide learners with a strong knowledge of basic programming concepts without writing code. Scratch is a computer programming language that allows creating interactive stories, games, and animations and sharing them online straight forward and exciting. Students will receive hands-on practical experience in basic game creation after successfully completing the course.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to programming
- => Working with sprites
- => Scratch components
- => Scratch Motion block
- => Scratch Looks block
- => Scratch Sound block
- => Scratch Control block
- => Scratch Sensing block
- => Scratch Operators
- => Scratch Variables

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to Scratch Programming :
  - ~ Course Introduction
  - ~ Who is this course for?
  - ~ Course Overview
  - ~ Course Outcome
  - ~ Why start with Scratch programming?
  - ~ Sprites vs Images
  - ~ Block Categories
- => Scratch Components :
  - ~ Variables
  - ~ Datatypes
  - ~ Keywords
  - ~ Conditional Statements
  - ~ Control Flow
- => Scratch Motion Block :
  - ~ Move tag
  - ~ Turn tag
  - ~ Go to tag
  - ~ Glide tag
- => Assignment 1 :
  - ~ Set x and y to (10, 100) and glide 5 seconds to random position.



=> Scratch Looks Block :

- ~ Say tag
- ~ Think tag
- ~ Various costume tags
- ~ Backdrop tag

=> Assignment 2 :

- ~ Make a new costume for sprite and make it change using next costume block

=> Scratch Sound Block :

- ~ Play sound
- ~ Start sound
- ~ Stop sound
- ~ Change pitch

=> Assignment 3 :

- ~ Make sprite walk 10 steps in the right direction and make a pop sound at the end.

=> Scratch Events Block :

- ~ Backdrop events
- ~ Broadcast events
- ~ Loudness events

=> Assignment 4 :

- ~ Use 'when this sprite clicked block and say hello after 1 second

=> Scratch Control Block :

- ~ Wait control tag
- ~ Repeat control tag
- ~ Forever control tag

=> Assignment 5 :

- ~ Make a sprite rotate forever using control flow

=> Scratch Sensing Block :

- ~ Touching sensing tag
- ~ Touching colour sensing tag
- ~ Distance sensing tag
- ~ Set drag mode

=> Assignment 6 :

- ~ Use the mouse x and mouse y of sense block and let the sprite say the coordinates

=> Scratch Operators Block :

- ~ Arithmetic operators
- ~ Comparison operators
- ~ and, or, not operations

=> Assignment 7 :

- ~ Pick a random number from 1 to 100 and let sprite say true if its greater than 50 otherwise false

=> Scratch Variables Block :

- ~ Set variable
- ~ Change variable
- ~ Show variable
- ~ Hide variable

=> Assignment 8 :

- ~ Set my variable value using a random operator and check if the variable is greater than 50. Let sprite say the output.

=> Conclusion :

- ~ Scratch in a nutshell
- ~ Various applications work in Scratch

# Amazon Lex

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Amazon-Lex>

## Course Description :-

This course is designed for creating intelligent chatbot interfaces with AWS (Application Web Services). We'll make our hands dirty and take a challenge in which we have to develop a sample chatbot together; after that, we'll learn how to make our bot available on our Facebook Page, Slack, WhatsApp, Telegram.

## Course Features :-

- => Basics of Chatbot
- => Integration of your bot in Facebook, Telegram, slack
- => Challenges
- => Source code
- => Downloadable resources
- => Quizzes
- => Completion Certificate

## What you will learn :-

- => Complete architecture of Amazon lex
- => How to create one hybrid conversational interfaces (i.e. chatbots) that work in any of the platforms like in Facebook Messenger, Skype, Slack, Telegram, and many more
- => Create business ready chatbots

## Requirements :-

- => AWS Account
- => No prior experience with Chatbot
- => Slack, Facebook, Telegram accounts
- => A system with Internet Connection.
- => Your Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Course Introduction :
  - ~ Introduction Preview
  - ~ What is Chatbot?
  - ~ Why Chatbot?
  - ~ Get your AWS account ready
- => Amazon Lex Fundamentals :
  - ~ What is Lex Preview
  - ~ Lex supported languages
  - ~ Programming modal
  - ~ Intent/Slot
  - ~ Lex model building API
  - ~ Runtime API Operations
  - ~ Managing messages
  - ~ Confidence score
  - ~ Conversation history log
  - ~ Built-in intent
  - ~ Built-in slot
  - ~ Custom slot
  - ~ Sentiment analysis
  - ~ Lambda function schema
  - ~ Integration
- => Custom Building Chatbot :
  - ~ Chatbot Problem Statement
  - ~ Create Intents
  - ~ JokeIntent
  - ~ Lambda

- ~ *Facebook integration*
- ~ *Slack integration*
- ~ *Slack app*

=> Course Summary :

- ~ *Summary*

# Email Marketing

---

Topic Name : DIGITAL MARKETING

Sub-topic Name : EMAIL MARKETING

Course link : <https://ineuron.ai/course/Email-Marketing>

## Course Description :-

Email Marketing course is designed to provide an in depth knowledge on various aspects & concepts of Email Marketing. A step by step learning will help to focus on each & every parameter of Email Marketing. The Email marketing course will take you through the end-to-end process of Emailing, Email design & templates, Email Subscription management, Reporting, Analytics and more.

## Course Features :-

- => Completion Certificate
- => Quiz in every module
- => Real-time Project
- => Assignment in all modules

## What you will learn :-

- => Understanding the importance, Benefits & forms of Email Marketing
- => Gain understanding on Email Marketing Domain & its concepts
- => Setting up of an Email Campaign
- => Deployment of an Email Campaign
- => Automation Process
- => Analyse Email Campaign

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Ankur Khanna :

~ Highly-motivated, energetic and dynamic Digital Marketing Mentor and Assistant Professor having 7+ years of experience in Digital Marketing Industry. Strong practical knowledge of different digital marketing tools aimed at meeting the needs of diverse groups of learners.

## Curriculum details :-

=> Email Marketing :

- ~ Introduction to email marketing Preview
- ~ Who opt email marketing?
- ~ What we learn in email marketing? Preview
- ~ Understanding the definition of email marketing Preview
- ~ Understanding the process of email marketing
- ~ Process steps in email marketing
- ~ Important roles in email marketing
- ~ What actually email marketer do?
- ~ Understanding audience in email marketing
- ~ Understanding the audience targeting in email marketing
- ~ Why it is important to define and targeting audience in email marketing?
- ~ Benefits of targeting those willing to buy
- ~ How to identify target?
- ~ Email deliverability in email marketing
- ~ Why is it essential to track email deliverability?
- ~ How to track email deliverability?
- ~ Difference between email delivery and email deliverability
- ~ How to improve email deliverability? Preview
- ~ Factor affecting email deliverability
- ~ More factors affecting email deliverability
- ~ Understanding email deliverability scheme

# JAX

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/JAX>

## Course Description :-

JAX

## Course Features :-

- => Roadmap
- => Interview Questions and their approach discussions
- => Learn solving Scenario-based questions
- => Improve your skills and knowledge by solving different types of questions
- => Assignments
- => Quizzes
- => Challenges
- => Completion certificate

## What you will learn :-

- => Basic understanding of digital marketing tools
- => Search engine optimization techniques
- => How to work on Google ads
- => Social media marketing on various platforms
- => Email marketing using Mailchimp
- => Content creations like written, graphics and video

## Requirements :-

- => Understanding of basic marketing terminologies
- => A system with internet connection
- => Dedication

## Instructors :-

=> Boktiar Ahmed Bappy :

~ This is Bappy. I aim for simplicity in Data Science. Real Creativity won't make things more complex. Instead, I will simplify them, Interested in a Data Science Career and so develop myself accordingly. Data Scientist and lecturer with working experience in Machine Learning, Deep Learning, Microcontrollers and Electronics systems. Hands-on experience in classification, regression, clustering, computer vision, natural language processing and transfer learning models to solve challenging business problems. I have a huge interest in Robotics. I have innovated a lot of innovations, ideas, projects & robots and got a lot of achievements.

# Android P with Machine Learning Apps

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : ANDROID

Course link : <https://ineuron.ai/course/Android-P-with-Machine-Learning-Apps>

## Course Description :-

Learning Android Development with Machine Learning will look great on any Android developer's CV. Machine Learning is a kind of Artificial Intelligence (AI) that allows software to learn, explore, and predict outcomes without the need for human intervention. Machine learning has been employed in a variety of industries, and it is currently being actively used to the creation of mobile applications. Machine learning algorithms can analyse specific user activity patterns and respond to search queries with ideas and recommendations. This course will teach you how to use Android with Machine Learning .

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Android Studio fundamentals
- => Theme customization
- => Buttons and toasts
- => Fully customized Gradles
- => Android elements and components
- => SQLite database
- => JSON and APIs
- => Firebase
- => Machine Learning in Android
- => Various projects

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to Android P development :
  - ~ Pep talk - Do not skip
  - ~ Tools that we will need
  - ~ Android History
- => Windows installation and setup :
  - ~ Installation of Android studio in WINDOWS
  - ~ AVD configuration and Hello world for WINDOWS
- => MAC setup and installation :
  - ~ Installation of Android - MAC
  - ~ Setting up Android Virtual device and config
- => Tour theme and App icons :
  - ~ Creating a project - API levels
  - ~ Exploring files in dir structure
  - ~ A tour of Android studio and customization - part 1
  - ~ A tour of Android studio and customization - part 2

- ~ Theme customisation and app on real device
- ~ Problems in App icon - Customization

#### => Buttons and toasts :

- ~ Button Customization
- ~ Click events for buttons
- ~ Assignment Solution
- ~ Methods and buttons
- ~ Basics of Toast and assignment
- ~ Shorter toasts

#### => Fully Customized Toasts and Gradles :

- ~ Basics setup for custom layouts
- ~ Preparing custom layouts
- ~ Customized layout inflation
- ~ Designing Elements in Linear layout
- ~ Gradle documentation
- ~ Final customization with gradle

#### => Components Tour of Android elements :

- ~ Components tour
- ~ Exploring text fields
- ~ Buttons and widgets in android
- ~ Understand layouts in Android

#### => Dice Roller app :

- ~ Designing assets for dice game
- ~ UI for DiceRoller
- ~ Writing code for diceRoller
- ~ Your assignment for this section

#### => Fun Background app :

- ~ Fun Background Design
- ~ Code part - fun background app

#### => Animated Login App :

- ~ Design assets for project AnimatedLogin
- ~ Applying animations in layout
- ~ Button Customization for app
- ~ Everything about button Customization
- ~ 1 more thing about buttons

#### => Truth Dare Game :

- ~ Setting up UI for Truth dare game
- ~ Code for Game and assignment

#### => Components of Android App :

- ~ Country Selector App - UI
- ~ Country Selector App - Code
- ~ Quick Change App
- ~ Burger Rating app - UI
- ~ Burger Rating app - code and assignment
- ~ Seekbar implementation
- ~ Uploader App UI
- ~ Uploader App Code with thread
- ~ Date Time picker in Android

#### => Currency Converter app :

- ~ Design of currency Converter app
- ~ Design of currency Converter app part 2
- ~ Handling Empty input and Assignment

#### => 3 Apps - Drumpad, examTimer, Music Player :

- ~ Going to a new screen

#### => 4 Apps - Drumpad, examTimer, Music Player :

- ~ Passing multiple values from intent

#### => 5 Apps - Drumpad, examTimer, Music Player :

- ~ MediaPlayer Class

#### => 6 Apps - Drumpad, examTimer, Music Player :

- ~ Setting layout for DrumPad App

#### => 7 Apps - Drumpad, examTimer, Music Player :

- ~ DrumApp code and assignment

#### => 8 Apps - Drumpad, examTimer, Music Player :

- ~ Exam Timer App design

#### => 9 Apps - Drumpad, examTimer, Music Player :

- ~ MediaPlayer App UI

#### => 10 Apps - Drumpad, examTimer, Music Player :

- ~ Exam Timer App code and sound

#### => 11 Apps - Drumpad, examTimer, Music Player :

- ~ Finishing Music Player app and Rockers

#### => Recycler and Card Views :

- ~ Recycler and Card Views Introduction
- ~ Custom layouts and getters
- ~ ArrayList for views
- ~ 10 Step guide for custom adapters
- ~ Main config for Insta cards

- ~ Refractoring the data
- ~ Add and remove Cards

=> SQLiteDatabase App - Student Record :

- ~ Introduction to database - UI setup
- ~ Database Helper introduction
- ~ Insert and Update data using helper
- ~ CRUD helper in Sqlite
- ~ Helper for showing messages
- ~ Adding data in sqlite
- ~ Getting data and handling cursor
- ~ Getting all data at once
- ~ Update and deletion of data

=> Jason and API apps :

- ~ What is API and JSON
- ~ Converting regular objects in JSON
- ~ Json to regular objects and Serialized name
- ~ Objects inside an object
- ~ Array in an object
- ~ Volley and API Introduction
- ~ Fetching an API request
- ~ Singleton in Volley

=> Firebase - Amazing Online database :

- ~ Section Intro
- ~ What is Firebase?
- ~ Exploring Firebase for Android
- ~ Setting layout for login system
- ~ User Registration System
- ~ User login & logout
- ~ Firebase Database - Rock Paper Scissor Online Game
- ~ Understanding Firebase Database
- ~ Running game on multiuser
- ~ Setting user registration system to database UI
- ~ Setting user registration system to database - code
- ~ Getting complex user data from database
- ~ Firebase Image Uploader Part 1
- ~ Firebase Image Uploader Part 2

=> Machine learning - Face and Smile detection app :

- ~ Machine Learning KIT in Firebase
- ~ Connecting with MLKIT online
- ~ Custom assets and gradle
- ~ Firebase app initializer
- ~ Inflating result dialog box
- ~ Open a camera on a REAL device
- ~ Final code for Face and smile detection

=> Machine Learning - Text Detection app :

- ~ Text Recognization app
- ~ How to download exercise files
- ~ Adding Custom Assets
- ~ Firebase initializer
- ~ Result Activity
- ~ Firecamera in our app
- ~ Text Recognization and Debug

=> How to publish app on store :

- ~ How to publish app on store



# A Job Ready Bootcamp in C++, DSA and IOT Tech Neuron

---

Topic Name : PROGRAMMING

Sub-topic Name : C++

Course link : <https://ineuron.ai/course/A-Job-Ready-Bootcamp-in--C++,-DSA-and-IOT-Tech-Neuron>

## Course Description :-

C++ Job Ready course has been created specifically to familiarize you with the concepts & applications of C++ in industry. This Course is for the students who want to build strong concepts & theories with Project Building with data structures and IOT applications.

Through this course, you will be confident enough to crack any kind of interview related to C++.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Flow Control Statements
- => Arrays and Strings
- => Object Oriented Programming
- => Memory management
- => Exception handling
- => File Management in C++
- => Standard Template Library
- => Arduino Simulation Overview
- => Data Structures and its Implementation in C++
- => FAANG interview Preparation
- => Arduino Based Projects
- => Resume development

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction :
  - ~ Introduction to C++ Language
  - ~ Features of C++
  - ~ History of C++
  - ~ Version history of C++
  - ~ An introduction to programming for absolute beginner
  - ~ How to develop a software using C++?
  - ~ Setup Environment on Windows
  - ~ Setup Environment on Mac
  - ~ Setup Environment on Linux
  - ~ Exploring first program
  - ~ Tokens: Constant, Variables and Keywords
  - ~ Data Types, Variable Declaration
  - ~ Input/output statements: cout and cin
  - ~ unary operator
  - ~ Arithmetic Operators
  - ~ Bitwise Operators
  - ~ Relational Operators
  - ~ Logical Operators
  - ~ Assignment Operators
  - ~ Other Operators
- => Flow Control Statements :
  - ~ Decision Control: if, if else, conditional operator
  - ~ Decision Control: Nested if else, if else ladder

- ~ Iterative Control: while
- ~ Iterative Control: do while
- ~ Iterative Control: for Loop
- ~ Break and Continue
- ~ Nested Loops
- ~ Star Pattern Problems
- ~ Switch case control
- ~ Menu Driven Programming

#### => Functions and Recursion :

- ~ What is a function?
- ~ usages of function in modular coding
- ~ Type of function in C++
- ~ predefined function
- ~ user defined function
- ~ Tracing a code with multiple functions
- ~ merits and demerits of function
- ~ Ways to define a function
- ~ Recursion
- ~ Inline function
- ~ Default Arguments

#### => Making the hard jargon simple - Pointers :

- ~ Pointers Introduction
- ~ Address of operator (&)
- ~ Dereferencing operator (\*)
- ~ Base Address
- ~ Extended Concept of Pointers
- ~ Pointers Arithmetic
- ~ Application of pointers
- ~ Call by value
- ~ call by address
- ~ Reference Variable
- ~ call by reference
- ~ Difference between pointers and reference variable?
- ~ Types of pointers
- ~ Wild Pointer
- ~ dangling pointer
- ~ null pointer
- ~ void pointer
- ~ Function pointers

#### => Arrays and Strings :

- ~ Array basics
- ~ Declaring Arrays
- ~ Initializing Arrays
- ~ Accessing Array Elements
- ~ Two Dimensional
- ~ Multidimensional Arrays
- ~ pointers to an array
- ~ passing arrays to function
- ~ array of pointers
- ~ pointer to array

#### => Strings :

- ~ String basics
- ~ String and Functions
- ~ Handling Multiple strings
- ~ Handling Strings with pointers
- ~ Assignment
- ~ Concatenation
- ~ Substrings
- ~ Character access
- ~ String Utilities
- ~ String comparison
- ~ String I/O
- ~ String Searching
- ~ String reverse
- ~ string Transformation
- ~ String length

#### => Object Oriented Programming :

- ~ Introduction to OOP
- ~ Object oriented Vs procedural programming
- ~ key principals of OOPS
- ~ Encapsulation through structure
- ~ Encapsulation through classes
- ~ Classes and Objects
- ~ Access specifier
- ~ Instance Members
- ~ Static members
- ~ Function call by passing object and returning object
- ~ Function Polymorphism (Function overloading)
- ~ Constructor
- ~ Constructor overloading
- ~ Default constructor
- ~ Copy constructor
- ~ Destructor
- ~ Deep copy Vs shallow copy
- ~ Operator Overloading

- ~ Overloading of pre and post increment operator
- ~ Friend Function
- ~ Friend operator
- ~ Benefits of Friend function
- ~ Overloading of insertion and extraction operator
- ~ Abstraction in c++
- ~ Data hiding in c++
- ~ Private constructor in C++
- ~ Private Destructor in C++

#### => Memory Management with Pointers :

- ~ What is Memory Management?
- ~ Why is memory management required?
- ~ Object Pointer
- ~ The this pointer
- ~ New Operator
- ~ Delete Operator
- ~ Memory Leak

#### => Inheritance :

- ~ C++ Inheritance
- ~ Advantage of C++ Inheritance
- ~ Types Of Inheritance
- ~ Single inheritance
- ~ Multiple inheritance
- ~ Hierarchical inheritance
- ~ Multilevel inheritance
- ~ Hybrid inheritance
- ~ Visibility mode in inheritance
- ~ Public , private and protected
- ~ Constructor and destructor in inheritance
- ~ Diamond problem
- ~ Inheritance method
- ~ Function overriding
- ~ Function hiding
- ~ Base pointer
- ~ Virtual function
- ~ Pure Virtual function
- ~ Abstract class in c++
- ~ Virtual destructor

#### => Exception handling :

- ~ History of Exception handling
- ~ Error Vs exception
- ~ Run time Exception and compile time Exception
- ~ C++ Standard Exceptions
- ~ Demo of exception one by one
- ~ Try
- ~ Catch
- ~ Throw
- ~ Catch all
- ~ Define New Exceptions
- ~ Handle Any Type of Exceptions

#### => File Management in C++ :

- ~ What is file handling?
- ~ Introduction to stream
- ~ ofstream
- ~ ifstream
- ~ fstream
- ~ Txt file vs binary file
- ~ Opening a File
- ~ Mode of file opening
- ~ Writing data to a File
- ~ appending data to a file
- ~ Reading data from a File
- ~ Close a File
- ~ Object by object reading and writing
- ~ Renaming a file
- ~ Removing a file
- ~ File Position Pointers

#### => Standard Template Library :

- ~ Template
- ~ Overview of STL
- ~ Iterator
- ~ Types of Iterators
- ~ Sequence Containers
- ~ Vector
- ~ List
- ~ Deque
- ~ Arrays
- ~ forward\_list
- ~ Container Adaptor
- ~ Queue
- ~ Priority Queue
- ~ Stack
- ~ Associative Containers
- ~ Set
- ~ Multiset

- ~ Map
- ~ Multimap
- ~ Unordered Associative Containers
- ~ Unordered set
- ~ Unordered multiset
- ~ Unordered map
- ~ Unordered multimap
- ~ Functors
- ~ Function Pointers
- ~ lambda
- ~ String
- ~ Pair
- ~ Tuple
- ~ Algorithms

=> Data Structures and its Implementation in C++ :

- ~ Introduction to Data Structures
- ~ Why you should learn data structure?
- ~ use case of data structure
- ~ Why product based companies focus on data structure

=> All about Arrays :

- ~ Down side of using conventional arrays
- ~ Array data structure
- ~ Dynamic Arrays

=> Linked List :

- ~ Singly Linked List
- ~ Doubly Linked List
- ~ Circular Linked List
- ~ Circular Doubly Linked List

=> Stack and Queues :

- ~ Stack Introduction
- ~ Implementation of Stack using arrays
- ~ Implementation of Stack using Linked List
- ~ Queue Introduction
- ~ Implementation of Queue using arrays
- ~ Implementation of Queue using Linked List
- ~ Two way stack
- ~ Double Ended Queue
- ~ Priority Queue

=> Tree :

- ~ Tree Introduction
- ~ Binary Tree and its variations
- ~ Binary Search Tree
- ~ Implementation of BST

=> Graph :

- ~ Graph Introduction
- ~ Implementation of Graph

=> Competitive Programming :

- ~ Introduction to Competitive Programming
- ~ Develop solving approach with 20 examples

=> Project Work :

- ~ Number Guessing Game
- ~ Employee Record Management
- ~ Book Record Management
- ~ Library Management System
- ~ Quiz Master
- ~ Tic Tac Toe Game

=> Industry/IOT based Project Work :

- ~ Applications of C++ in IOT

=> Arduino Simulation Overview :

- ~ Arduino introduction
- ~ usages of Arduino in real time
- ~ basic component of Arduino
- ~ pin diagram of Arduino
- ~ introduction of sensors
- ~ environment setup for Arduino
- ~ Controlling Element
- ~ Blinking LED
- ~ Push Button
- ~ Potentiometer Controller
- ~ Servo Motor
- ~ DC Motor
- ~ Photo Resistor

=> Arduino Based Projects :

- ~ Project Detail description
- ~ business use case of this project
- ~ Project architecture
- ~ project setup
- ~ Component identification for project
- ~ Project 1 - Street Light Project
- ~ Project 2 - Intruder Buzzer System Design
- ~ Project 3 - Ice Cream Factor Sensor Design

- ~ Project 4 - Passive Infrared
- ~ Project 5 - Designing African Home with PIR Sensor
- ~ Project 6 - Agriculture Design - Moisture Sensor
- ~ Project 7 - Music Generator Sensor Design
- ~ conclusion of project
- ~ Production of project
- ~ integration in real time

=> FAANG interview Preparation :

- ~ Overview of FAANG companies
- ~ Interview Preparation guide for Amazon
- ~ Interview Preparation guide for Google
- ~ Interview Preparation Guide for Microsoft
- ~ Interview Preparation Pro Tips from Industrial Mentors
- ~ Interview Questions and their solutions
- ~ 100+ MCQs

=> Resume development :

- ~ Key points for your resume
- ~ Templates for resume
- ~ Project for your resume
- ~ prepare your GIT
- ~ prepare your social media profile
- ~ prepare your demo for your resume
- ~ Detail project report
- ~ resume verification
- ~ place where you can apply for job
- ~ Final touch
- ~ Go get your Dream job

=> 300+ Practice Problems :

- ~ Section -1: Introduction - 30 Problems

=> 301+ Practice Problems :

- ~ Section -2: Flow Control Statements - 80 Problems

=> 302+ Practice Problems :

- ~ Section -3: Functions and Recursion - 40 Problems

=> 303+ Practice Problems :

- ~ Section -4: Making the hard jargon simple - Pointers -10 Problems

=> 304+ Practice Problems :

- ~ Section -5: Arrays and Strings - 40 Problems

=> 305+ Practice Problems :

- ~ Section -6: Object Oriented Programming - 30 Problems

=> 306+ Practice Problems :

- ~ Section -7: Memory Management - 10 Problems

=> 307+ Practice Problems :

- ~ Section -8: Standard Template Library - 30 Problems

=> 308+ Practice Problems :

- ~ Section -1: DSA All about Arrays - 15 Problems

=> 309+ Practice Problems :

- ~ Section -2: DSA Linked List - 15 Problems

=> 310+ Practice Problems :

- ~ Section -3: DSA Stack and Queues - 15 Problems

=> 311+ Practice Problems :

- ~ Section -4: DSA Tree - 5 Problems

=> 312+ Practice Problems :

- ~ Section -5: DSA Graph - 5 Problems

=> 313+ Practice Problems :

- ~ Section -6: DSA Algorithms - 5 Problems

=> Mini-Challenges :

- ~ It will be assigned by mentor after every major module

# HTML CSS Coding Interview Preparation

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : WEB DEVELOPEMENT INTERVIEW

Course link : <https://ineuron.ai/course/HTML-CSS-Coding-Interview-Preparation>

## Course Description :-

This course is designed mostly for HTML CSS test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> HTML Theoretical Test

=> HTML Practical Test

=> HTML Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> HTML CSS Test :

- ~ HTML CSS Test 1
- ~ HTML CSS Test 2
- ~ HTML CSS Test 3
- ~ HTML CSS Test 4
- ~ HTML CSS Test 5
- ~ HTML CSS Test 6
- ~ HTML CSS Test 7
- ~ HTML CSS Test 8
- ~ HTML CSS Test 9
- ~ HTML CSS Test 10
- ~ HTML CSS Test 11
- ~ HTML CSS Test 12
- ~ HTML CSS Test 13
- ~ HTML CSS Test 14
- ~ HTML CSS Test 15
- ~ HTML CSS Test 16
- ~ HTML CSS Test 17
- ~ HTML CSS Test 18
- ~ HTML CSS Test 19
- ~ HTML CSS Test 20

# ParlAI Chatbot

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/ParlAI-Chatbot>

## Course Description :-

This program is specialized in providing theoretical as well as practical understanding of the ParlAI module which allows you to learn how to build dialog based model chatbot using python based parlAI framework. Course curriculum includes concepts about ParlAI framework, tutorial and much more!

## Course Features :-

- => Learning of different concepts of ParlAI framework
- => Self Paced Videos
- => Completion Certificate
- => Assignments
- => Practical Implementation
- => Quiz

## What you will learn :-

- => ParlAI framework in detail
- => Basic concepts
- => Installation
- => Tutorials
- => Creating a chatbot

## Requirements :-

- => Little prior knowledge in chatbot
- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Course Introduction :
  - ~ Syllabus overview Preview
  - ~ Introduction to ParlAI Preview
- => Basics :
  - ~ World, agents, teachers
  - ~ Action and observations
- => Installation :
  - ~ Installation
- => Practical Implementation :
  - ~ List of available tasks/datasets
  - ~ List of available agents
  - ~ Loading data
  - ~ Training model Preview
  - ~ Prediction

# Solidity Live Class

---

Topic Name : BLOCKCHAIN

Sub-topic Name : SOLIDITY

Course link : <https://ineuron.ai/course/Solidity-Live-Class>

## Course Description :-

Solidity & Solana Blockchain course is designed to provide an in depth knowledge on various aspects & concepts of blockchain & Solidity. A step by step learning will be help to focus on each & every parameter of Blockchain. This course will take you into a deep dive into the state of the art blockchain technology and how to go about writing smart contracts in the ethereal platform. Moreover, this is a project-ready course which will help you take whatever you learn and apply it into a real-world portfolio-ready app, which you can showcase to the world.

## Course Features :-

- => Onine Live Classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Solidity Fundamentals
- => Smart Contracts in Solidity
- => Smart Contract Best Practices
- => What are ICO and what are tokens
- => Understanding about ERC-20
- => Solana Blockchain
- => Web 3.0
- => Intro to IPFS
- => Oracles
- => DeFi
- => NFTs

## Requirements :-

- => Knowledge of Web Designing will be Advantageous.

## Instructors :-

- => Sanjeevan Thorat :

~ Data Scientist and Blockchain developer, with experience in developing and managing end to end solutions. I have hands-on experience in Python Programming Language, Machine Learning Deep Learning and Natural language processing. Blockchain development experience in smart contracts, Decentralised Finance applications, DAOs, NFTs and Oracles running on Ethereum and Polygon blockchains. I have worked with various clients for different industry projects in the blockchain space. I specialize in building smart contracts on the Ethereum blockchain along with JavaScript integration for enhancing user experience to generate maximum returns on investment.

## Curriculum details :-

- => Introduction :
  - ~ Introduction to course
- => Solidity Fundamentals :
  - ~ Smart Contracts in Solidity
  - ~ Basic-Smart-Contract-Part1
  - ~ Basic Smart Contract Part 2
  - ~ Data types and Variables - part 1
  - ~ Data types and Variables - part 2
  - ~ Functions
  - ~ Storage vs Memory
  - ~ Events and logs
  - ~ Factory contract
  - ~ Security Of Smart Contracts
  - ~ Inheritance
  - ~ Inline Assembly



~ *Application Binary Interface*

=> Smart Contracts Pitfalls, Testing and Debugging :

~ *Unit tests*

~ *Integration Tests*

~ *Javascript tests*

~ *Smart Contract Best Practices*

=> Creating our own cryptocurrency on Ethereum Network :

~ *What are ICO and what are tokens*

~ *Understanding about ERC-20*

~ *Writing code for our cryptocurrency*

~ *Safe Math*

~ *Creating the cryptocurrency*

~ *Deploying it to the network*

=> Solana Blockchain :

~ *Introduction to solana Blockchain*

~ *Creating our own cryptocurrency on the Solana Network using CLI - part 1*

~ *Creating our own cryptocurrency on the Solana Network using CLI - part 2*

~ *Creating our own cryptocurrency on the Solana Network using CLI - part 3*

~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 1*

~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 2*

~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 3*

~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 4*

=> Web 3.0 & Connecting everything into a project :

~ *What is Web 3.0 ?*

~ *iNeuron Marketplace - part1*

~ *iNeuron Marketplace - part2*

~ *iNeuron Marketplace - part3*

~ *iNeuron Marketplace - part4*

~ *iNeuron Marketplace - part5*

~ *iNeuron Marketplace - part6*

~ *iNeuron Marketplace - part7*

~ *iNeuron Marketplace - part8*

=> A little more about ethereum :

~ *Ethereum naming service*

~ *Intro to IPFS*

~ *Oracles*

~ *DeFi*

=> NFTs :

~ *What are NFTs and ERC721*

~ *Create Your own NFT part 1*

~ *Create Your own NFT part 2*

~ *Create Your own NFT part 3*

~ *Create Your own NFT part 4*

# Interview ready DSA course in Python

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA INTERVIEW

Course link : <https://ineuron.ai/course/Interview-ready-DSA-course-in-Python>

## Course Description :-

A comprehensive chase to excel any interview for the Data Structures and Algorithms. The course comprises 250 curated data structures problems grounded on personal interview experiences of students of preceding batches. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

## Course Features :-

- => Downloadable resources
- => Roadmap
- => Quizzes
- => Interview Questions
- => Resume Preparation
- => Completion Certificate
- => Various Assignments

## What you will learn :-

- => Basic and Complex data structures and algorithm concepts starting from linked list, array continuing to graph, trees, and ending on advanced programming concepts like dynamic programming, greedy algorithm, etc.
- => An optimised approach to solving complex problems.
- => Competitive programming from basic to advanced level.
- => Deep understanding of the implemented approach.
- => What data structure to use for what questions?
- => During the course journey, you can see yourself evolving and overcoming the usual mistakes you might have previously committed.
- => This course will help you initiate and work in the right direction for different projects that you can mention in your resume.
- => End to end understanding of the solved questions.
- => Solve questions with the least time complexity.
- => Most importantly, you will learn to face the most challenging problem-solving interviews.

## Requirements :-

- => Understanding of python programming language
- => A system with a decent internet connection

## Instructors :-

=> Akarsh Jaiswal :

~ A meticulous software engineer and impeccable educator, Akarsh Jaiswal has been supporting the community in various forms. He is currently working as a Full Stack Developer in a MNC. Being a software developer and data structures scholar himself, he has supported multiple startups and trained innumerable students, helping them shape their careers in IT industry. His main forte lies in competitive programming which has led him to win number of prestigious hackathons worldwide. Talking of the tech, he is well versed with Python & Java and has many running projects. He is a highly motivated individual and believes that every learning in Computer Science has the potential to open infinite avenues.

## Curriculum details :-

- => Recursion :
  - ~ Why do we need recursion? Preview
  - ~ What is a recursive tree? Preview
  - ~ Basic algorithms for recursion, how should we break the input? What is a base condition?
  - ~ Fibonacci series and difference between recursive and iterative method
  - ~ Print numbers from 1 to n using recursion
  - ~ Print numbers from n to 1 using recursion
  - ~ Factorial of a number
  - ~ Reverse an array, string using recursion
  - ~ Reverse a stack using recursion
  - ~ Sort an array using recursion
  - ~ Tower of Hanoi problem Preview
  - ~ Generating all subsets/powersets
  - ~ Generating all unique subsets/powersets
  - ~ Generating all permutation with spaces
  - ~ Generating all permutation with case change
  - ~ Power of Two (Recursion/Iterative)
  - ~ Reverse a given number for eg. 634 should be 436

- ~ Letter case Permutation
- ~ Check if number is palindrome or not ?
- ~ Check if a number is Armstrong or not?

=> Linked List :

- ~ What is a Linked List? Preview
- ~ Advantages/Disadvantages
- ~ Properties
- ~ What is a node, its structure?
- ~ Types of Linked List
- ~ Making a linked list
- ~ Insertion Singly Linked List at start
- ~ Deletion Singly Linked List from start
- ~ Insertion Singly Linked List at middle
- ~ Deletion Singly Linked List from middle
- ~ Insertion Singly Linked List at end
- ~ Deletion Singly Linked List from end
- ~ Print a Linked List
- ~ Length of Linked List iterative
- ~ Search in a linked list
- ~ Delete a Linked List
- ~ Length of a linked list recursive
- ~ Insertion doubly Linked List at start
- ~ Deletion doubly Linked List from start
- ~ Insertion doubly Linked List at middle
- ~ Deletion doubly Linked List from middle
- ~ Insertion doubly Linked List at end
- ~ Deletion doubly Linked List from end
- ~ Insertion Circular Linked List at start
- ~ Deletion Circular Linked List from start
- ~ Insertion Circular Linked List at middle
- ~ Deletion Circular Linked List from middle
- ~ Insertion Circular Linked List at end
- ~ Deletion Circular Linked List from end
- ~ Merge two sorted Linked List
- ~ Reverse a Linked List iterative
- ~ Reverse a Linked List recursive
- ~ Loop in linked list using hashing
- ~ Loop in linked list using tortoise and hare algorithm
- ~ Start of the loop
- ~ Length of loop
- ~ Middle of Linked List
- ~ Check whether a linked list has repeated elements or not?
- ~ Remove duplicates from a sorted linked list
- ~ Remove duplicates from an unsorted linked list
- ~ Delete a Linked List node without head pointer
- ~ Count of every element in a linked list
- ~ Check the length of the Linked List is even or odd without counting nodes.
- ~ The intersection of Two Linked List using stack
- ~ The intersection of Two Linked List without extra memory
- ~ Find Nth Node from End of Linked list
- ~ Reverse a doubly linked list recursive
- ~ Reverse a doubly linked list iterative

=> Trees :

- ~ What is a tree?
- ~ What are a node and its structure?
- ~ What is a Binary Tree
- ~ What is a Binary Search Tree Preview
- ~ Types of trees?
- ~ Construction of a Tree?
- ~ Insertion of a node
- ~ Inorder traversal iterative
- ~ Preorder traversal iterative
- ~ Postorder traversal iterative
- ~ Preorder traversal recursive
- ~ Inorder traversal recursive
- ~ Postorder traversal recursive
- ~ Level order traversal
- ~ Level order traversal line by line
- ~ Vertical order traversal
- ~ Search in a tree
- ~ Height of tree
- ~ Number of nodes
- ~ Number of leaf nodes
- ~ Sum of all nodes
- ~ Inorder successor
- ~ Inorder Predecessor
- ~ Delete a Binary Tree
- ~ Number of nodes at nth level
- ~ Number of leaf nodes, internal nodes, total node if height is h

=> Sorting :

- ~ Bubble
- ~ Selection
- ~ Insertion
- ~ Merge
- ~ Quick
- ~ Counting

=> Searching :

- ~ Linear search
- ~ Binary search
- ~ Order not known Search
- ~ Ternary Search
- ~ First and last occurrence of an element
- ~ Count of element in a sorted array
- ~ Floor/ceil of an element in array
- ~ Next permutation
- ~ Searching in infinite sorted array
- ~ Index of first one in a binary sorted array
- ~ Find Pivot in rotated and sorted array
- ~ Search in rotated and sorted array
- ~ Find the number of 1s in a sorted binary array

=> Heap :

- ~ What is a heap
- ~ Types of Heap
- ~ Implementing Heap
- ~ Heap operations
- ~ Heap Sort
- ~ MinHeap Implementation
- ~ Time & space complexity
- ~ Kth smallest element
- ~ Kth largest element
- ~ K largest and smallest elements
- ~ K closest numbers map
- ~ Implementing Max & Min Heap which is not atomic
- ~ K closest numbers Heap without map
- ~ Top k frequent numbers
- ~ Frequency Sort
- ~ K closest points to origin

=> Stack & Queue :

- ~ What is a stack?
- ~ Implementing stack and its operations
- ~ Implementing queue and its operations
- ~ Types of queue
- ~ Parenthesis checker
- ~ Next greater to right
- ~ Next greater to left
- ~ Next smaller to right
- ~ Next smaller to left
- ~ Stack using queue
- ~ Queue using stack
- ~ Implementing Stack using linked list
- ~ Implementing Queue using linked list

=> Greedy Approach :

- ~ What is a greedy approach?
- ~ N meeting in one room
- ~ Activity Selection
- ~ Greedy algorithm to find the minimum number of coins
- ~ Fractional Knapsack Problem
- ~ Minimum number of platforms required for a railway
- ~ Job sequencing Problem

=> Array :

- ~ What is an array
- ~ Find the minimum and maximum element in an array
- ~ Segregate 0's and 1's in an array
- ~ Sort the array of 0s, 1s, and 2s
- ~ Reverse the given input array
- ~ Find a most frequent element in an array
- ~ Find non repeated elements in an array of integers
- ~ Find duplicate elements in an array
- ~ Find the first repeating element in an array of integers
- ~ Find the first non-repeating element in an array of integers
- ~ Find unique features in an array
- ~ Check whether the given array is sorted or not.
- ~ Move all the negative elements to one side of the array
- ~ Merge two sorted arrays to form a single array
- ~ Find the missing number in an integer array of 1 to 100
- ~ Remove duplicates from an array of integer
- ~ Move all 0s to the end of the array
- ~ What is a two-dimensional array
- ~ nth row/col of a matrix
- ~ Diagonal of a matrix
- ~ Upper lower triangular matrix
- ~ Transpose of a Matrix

=> String :

- ~ What is a string
- ~ Reverse a string
- ~ Find a maximum occurring character in a given string
- ~ Remove a given character from the string
- ~ Print duplicate characters of the given string.
- ~ Remove all duplicates from the given string.
- ~ Check if a string is a substring of another or not.

- ~ Check if two strings are rotation of each other or not
- ~ check if two given string is an anagram of each other
- ~ Find first non-repeating character of the given string
- ~ Generate all substrings of a string
- ~ Check if the given string is palindrome or not
- ~ Check if the given string has all unique characters
- ~ Check if one string is a permutation of the other or not
- ~ Write a function to perform basic string compression, e.g. aabcccccaaa would become a2b1c5a3
- ~ Write a function to perform basic string expansion, e.g. a2b1c5a3 would become aabcccccaaa

=> Graphs :

- ~ What is a graph data structure and its examples ?
- ~ Order and degree of a graph
- ~ Types of graph
- ~ Classes of graph
- ~ Representation of graph
- ~ Implementation of graph
- ~ BFS Traversal
- ~ BFS Traversal for disconnected graph
- ~ DFS Traversal
- ~ DFS Traversal for disconnected graph

=> Time Complexity :

- ~ What is time complexity?
- ~ Sample problem 1
- ~ Sample problem 2
- ~ Sample problem 3
- ~ Sample problem 4
- ~ Sample problem 5
- ~ Sample problem 6
- ~ Time complexity of recursive functions

=> Space Complexity :

- ~ What is space complexity?
- ~ Sample problem 1
- ~ Sample problem 2
- ~ Sample problem 3
- ~ Sample problem 4
- ~ Sample problem 5
- ~ Space complexity of fibonacci series problem

=> Hashing :

- ~ What is hashing?
- ~ Two sum problem
- ~ Find all symmetric pairs.

=> Maths :

- ~ Prime number
- ~ Factors of a number
- ~ GCD
- ~ LCM
- ~ Trailing zeroes in a factorial
- ~ Search in a sorted 2D matrix
- ~ Power function
- ~ Majority element
- ~ Grid Unique Paths

=> Dynamic Programming :

- ~ What is DP?
- ~ Fibonacci series
- ~ 0-1 Knapsack Recursive Preview
- ~ 0-1 Knapsack Memorized
- ~ 0-1 Knapsack Tabular
- ~ Subset sum problem
- ~ Equal sum partition problem
- ~ Count subset with given sum
- ~ Unbounded Knapsack
- ~ Coin change max ways
- ~ Coin change min ways
- ~ Longest common subsequence Recursive
- ~ Longest common subsequence Memorized
- ~ Longest common subsequence Tabular
- ~ Printing LCS
- ~ Longest common substring  $O(n^2)$
- ~ Shortest common subsequence
- ~ Kadane's algorithm

=> Bit Manipulation :

- ~ Introduction to Bit Manipulation
- ~ Bitwise operators Preview
- ~ Bitwise operators Implementation
- ~ Power of Two
- ~ Number of 1 Bits or Hamming Weight
- ~ Missing Number in array
- ~ Find element which appears once in an array while all other appears twice

# AWS Architect for Real World

---

Topic Name : CLOUD

Sub-topic Name : AWS

Course link : <https://ineuron.ai/course/AWS-Architect-for-Real-World>

## Course Description :-

This Amazon AWS course will take you from AWS fundamentals to being a professional AWS cloud practitioner. From foundations to advanced topics, you will master general cloud computing principles and AWS. There are many hands-on activities that you may do with an Amazon Web Services (AWS) free tier account to gain expertise. This is the greatest approach to get started if you want to get into a high-paying profession working with cloud computing services. You'll go from beginning to advanced ideas, with lots of opportunity to put what you've learned into practise.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practicals on AWS
- => Course completion certificate

## What you will learn :-

- => AWS fundamentals
- => S3 buckets
- => Lambda functions
- => SES
- => API gateways
- => Load balancing
- => Migration
- => Container services
- => AWS serverless

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Course Intro :

*~ AWS Architect for real world*

=> Getting started with AWS and IAM :

- ~ FAQ for AWS architect course*
- ~ Getting started with AWS and expectation*
- ~ Tour of AWS console with ROOT user*
- ~ AWS Infra - Region and AZ*
- ~ Securing root account and MFA*
- ~ Custom signin link for IAM*
- ~ Why groups are created*
- ~ Creating groups and users*
- ~ What are roles in IAM*
- ~ Temporary security credentials in IAM*
- ~ Billing alarms in Cloudwatch*
- ~ Password compliance*
- ~ buying domain on Route 53*

=> Amazon Elastic Compute Cloud -EC2 :

- ~ What is Elastic Compute*
- ~ Instance types and limits*
- ~ your first EC2 instance*
- ~ In depth guide for EC2 options*

- ~ Connecting to cloud instance
- ~ Configure an AWS web server
- ~ Stress testing, Cloud watch alarms and clean up
- ~ What are user data scripts
- ~ What is instance meta-data
- ~ Docs and hands-on with Elastic IP
- ~ Custom network interface cards in AWS
- ~ creating custom AMI
- ~ Launch with custom image and clean up
- ~ Placement groups - Cluster, partition and Spread
- ~ EC2 pricing - OnDemand, spot and reserved
- ~ Just for Exam

#### => Virtual Private Cloud - VPC :

- ~ Why you should focus more on VPC
- ~ Understand the default VPC
- ~ Create diagram of default VPC
- ~ CIDR deep dive
- ~ Your custom VPC
- ~ Creating subnet in custom VPC
- ~ Internet Gateway and route table
- ~ Lets complete the diagram
- ~ VPC DNS hostname and resolution
- ~ updates from corporate in VPC structure
- ~ Clean up the resources
- ~ Security groups VS NACL
- ~ Understand the next diagram for VPC
- ~ Diagram 2 - VPC and subnets
- ~ Diagram 2 - Route tables and IGW
- ~ Configure a NAT gateway
- ~ SSH agent forwarding
- ~ Bastion host and testing of diagram
- ~ Bastion host and testing of diagram part 2
- ~ NAT instance and configurations
- ~ VPC peering connection
- ~ What are transit gateways
- ~ A use case of Endpoints in VPC
- ~ preparing logs for audit - flowLogs
- ~ Resources for hybrid cloud - VPN and more
- ~ Lets audit the logs with Athena and Glue
- ~ Egress gateway cloudhub and clean up

#### => Load Balancing and scalability :

- ~ What are load balancers
- ~ Type of Load Balancer
- ~ Prep work for load balancers
- ~ Configure target groups
- ~ Creating an Application load balancer
- ~ Path and HOST based routing on domain
- ~ Cross Zone load balancer
- ~ Case of Sticky session
- ~ Clean up for ALB
- ~ Network Load Balancer
- ~ Scaling - Horizontal and Vertical
- ~ Auto Scaling Group configuration
- ~ Clean up for ASG resources

#### => Route 53 in Depth :

- ~ Welcome to Route 53
- ~ What are hosted zone - Public and Private
- ~ AWS DNS records - A and Alias
- ~ Creating instance in multiple region
- ~ Route 53 Health Checks
- ~ Simple and weighted route policy
- ~ FailOver and latency based policies
- ~ Multi value and restricting content on geo location
- ~ Clean up for Route 53

#### => Storage in AWS - S3 :

- ~ lets start with AWS storage
- ~ Introduction to S3 buckets
- ~ Permissions in S3 buckets
- ~ Static website hosting in S3 buckets
- ~ S3 bucket - Versioning and encryption
- ~ S3 event notifications
- ~ Access log BILLS and requester pays
- ~ S3 storage class
- ~ Data replication - CRR and SRR
- ~ S3 Select, Athena and Redshift - Query
- ~ Data life cycle policy
- ~ Getting started with cloudfront and OAI
- ~ Setup a cloudfront and OAI for a website

#### => Storage - Block and Object :

- ~ Instance Store - ephemeral
- ~ Types of EBS volume and IOPS
- ~ Creating and mounting EBS volume
- ~ Getting a snapshot of EBS
- ~ Re attach EBS volume
- ~ Data migration between AZ and Region

- ~ RAID 0 and 1 config
- ~ Creating and mounting Elastic File Storage
- ~ FSx for Windows and Lustre
- ~ Storage Gateway - Hybrid cloud
- ~ Storage Gateway NOT by LCO

#### => Databases in AWS :

- ~ Introduction to Databases in AWS
- ~ OLTP vs OLAP
- ~ Production level RDS walkthrough
- ~ Create a Mysql db in AWS
- ~ Multi AZ replica RDS
- ~ Creating read replicas
- ~ Read Replica VS Multi AZ deployment
- ~ AWS aurora Docs walkthrough
- ~ Getting started with DynamoDB
- ~ Creating a table in DynamoDB
- ~ Reading the DAX Docs
- ~ ElasticCache memcached
- ~ ElasticCache Redis and Redis cluster
- ~ Redshift Overview

#### => Application integration in AWS :

- ~ Application integration services by AWS
- ~ Simple queue service
- ~ Creating our first queue service
- ~ FIFO vs standard queue
- ~ Delay, visibility and retention time
- ~ Dead letter queue
- ~ Long polling and short polling
- ~ Attaching lambda to SQS
- ~ Clean up all the SQS resources
- ~ Step function and simple workflow service
- ~ Amazon MQ, Rabbit MQ and other services

#### => PAAS and IAAS in AWS :

- ~ Getting started with PAAS and IAAS
- ~ Cloudformation inDepth guide
- ~ Beanstack application deployment

#### => Process and Migrate the Data :

- ~ Kinesis and shards
- ~ Kinesis analytics and firehose
- ~ What is Elastic MapReduce
- ~ What is Athena, Glue and Glue Studio
- ~ Import from other Virtualization Services
- ~ Database Migration service and Schema Conversion Tool

#### => Security Compliance :

- ~ Security and Compliance - SOX, PCI and more
- ~ Key Management Service
- ~ Hardware Security Module in Cloud
- ~ AWS WAF and shield service
- ~ Active Directory in AWS
- ~ What is AWS Cognito
- ~ AWS single sign on
- ~ AWS Directory service

#### => Container Service :

- ~ What are container service in AWS
- ~ What is Docker
- ~ What is Elastic Container Registry
- ~ What are microservices
- ~ What is Elastic Container service
- ~ What is Fargate
- ~ What is Elastic Kubernetes Service
- ~ AWS walkthrough for ECS and EKS

#### => AWS Serverless :

- ~ Getting started with AWS serverless
- ~ A common warning for AWS
- ~ Route 53
- ~ Get Started with S3 bucket
- ~ Struggle of web page hosting
- ~ Hosting with policies
- ~ GET vs POST and handling response
- ~ Your first lambda in AWS
- ~ Lambda permission and cloud watch
- ~ Introducing API gateway
- ~ Lambda for POST information
- ~ Post Data and CORS error
- ~ First look at SES
- ~ New user for SES and lambda
- ~ Sending email from SES and lambda



# Statistics

---

Topic Name : DATA SCIENCE

Sub-topic Name : STATS

Course link : <https://ineuron.ai/course/Statistics>

## Course Description :-

Statistics is the discipline that concerns the collection, organization, analysis, interpretation and presentation of data. Statistics is the foundation behind all the work you want to do regarding Data Science. So, you must know all the statistical concepts to learn data science well. In this course, you will learn all the statistical concepts in detail that will be highly beneficial for various fields of Data Science.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable Resources
- => Completion Certificates

## What you will learn :-

- => Understand what a Normal Distribution is.
- => Explain the difference between continuous and discrete variables
- => Understand the Central Limit Theorem
- => Use the Z-Score and Z-Tables
- => Understand the difference between a normal distribution and a t-distribution
- => Create confidence intervals
- => Understand standard deviations
- => Understand what a sampling distribution is
- => Apply Hypothesis Testing for Proportions
- => Use the t-Score and t-Tables

## Requirements :-

- => Basic understanding of Maths
- => A system with internet connection
- => Your dedication

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course introduction :

~ Introduction Preview

=> Stats Fundamental :

~ Statistics Preview

~ Inferential Statistics

~ Descriptive Statistics

~ Mean, Median and Mode

~ Population vs Sample

~ Gaussian or Normal Distribution

~ Log Normal Distribution

~ Covariance

~ Central Limit Theorem

~ Chebyshev's inequality

~ Pearson Correlation Coefficient

~ Spearman's Rank Correlation Coefficient

~ Standardization vs Normalization

=> Python :

~ Use of Python in Statistics

=> Representation and interaction with Data :

~ Data as a table

~ Pandas DataFrame

=> Hypothesis testing: Comparing two groups :

- ~ *Student's T-test*
- ~ *Paired test*

=> Linear models, multiple factors, and analysis of variance :

- ~ *Python formulas for specifying statistical models*
- ~ *Multiple Regression*
- ~ *Analysis of variance(ANOVA)*

=> Visualization: Statistical exploration using Seaborn :

- ~ *Pairplot: scatter matrices*
- ~ *Implot: plotting a univariate regression*

=> Testing for interactions

# Machine Learning Masters

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-Masters>

## Course Description :-

Machine Learning Masters

## Course Features :-

- => Machine Learning in depth from beginning to advance discussion and implementation with Deployment.
- => Deep learning in-depth topic wise discussion and implementation with the project.
- => Docker and Kubernetes end to end with CI/CD pipeline for machine learning.
- => End to End Model Deployment in Azure, GCP, AWS, and Pivotal Cloud.
- => Python spark implementation with the project.
- => Time Series end to end implementation in machine learning and deep learning.
- => 26 + hands-on industry real-time projects.
- => Power BI and Tableau self-placed course.
- => Machine Learning Deep Learning Masters Certificate
- => 200 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access.
- => Doubt clearing one to one
- => Doubt clearing through mail and support team
- => Assignment in all the module
- => 20+ use case of Machine learning
- => A live project with real-time implementation
- => Resume building
- => career guidance
- => interview Preparation
- => Regular assessment
- => Job alerts
- => Online Instructor-led learning: Live teaching by instructors
- => Product Demo

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Data analytics
- => Mock interview
- => Interview preparation
- => Resume building

## Requirements :-

- => Dedication
- => Laptop with internet connectivity

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving

challenging business problems.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course Introduction :

- ~ Introduction of Data science and its application in Day to Day life Preview
- ~ Course overview and Dashboard description Preview

=> Python Core :

- ~ Introduction of python and comparison with other
- ~ Programming language
- ~ Installation of Anaconda Distribution and other python
- ~ IDE Python Objects, Number & Booleans, Strings
- ~ Container objects, Mutability of objects
- ~ Operators Arithmetic, Bitwise, Comparison and Assignment operators, Operators Precedence and associativity
- ~ Conditions(If else, if elif else) Loops(While ,for)
- ~ Break and Continue statement and Range Function.

=> String Objects and collections :

- ~ String object basics
- ~ String methods
- ~ Splitting and Joining Strings
- ~ String format functions
- ~ List object basics
- ~ List as stack and Queues
- ~ List comprehensions

=> Tuples, Set ,Dictionaries Functions :

- ~ Tuples, Sets Dictionary Object basics, Dictionary Object methods, Dictionary View Objects.
- ~ Functions basics, Parameter passing, Iterators Generator functions
- ~ Lambda functions
- ~ Map , Reduce, Filter functions

=> OOPS concepts Working with Files :

- ~ OOPS basic concepts
- ~ Creating classes and Objects Inheritance
- ~ Multiple Inheritance
- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other File methods

=> Exception Handling :

- ~ Exceptions Handling with Try except

=> Api :

- ~ Flask introduction
- ~ Flask Application
- ~ Open link Flask
- ~ App Routing Flask
- ~ URL Building Flask
- ~ HTTP Methods Flask

=> Database :

- ~ Mongo DB SQL
- ~ Lite python SQL

=> Python pandas Modules :

- ~ Python Pandas Series
- ~ Python Pandas DataFrame
- ~ Python Pandas Panel
- ~ Python Pandas Basic functionality

=> Python Numpy :

- ~ NumPy Narray Object
- ~ NumPy Data Types
- ~ NumPy Array Attributes
- ~ NumPy Array Creation Routines
- ~ NumPy Array from Existing
- ~ Data Array From Numerical Ranges
- ~ NumPy Indexing & Slicing
- ~ NumPy Advanced Indexing
- ~ NumPy Broadcasting
- ~ NumPy Iterating Over Array
- ~ NumPy Array Manipulation
- ~ NumPy Binary Operators
- ~ NumPy String Functions
- ~ NumPy Mathematical Functions
- ~ NumPy Arithmetic Operations
- ~ NumPy Statistical Functions
- ~ Sort , Search & Counting Functions
- ~ NumPy Byte Swapping
- ~ NumPy Copies Views
- ~ NumPy Matrix Library

~ NumPy Linear Algebra

## => Exploratory Data Analysis :

- ~ Feature Engineering and Selection
- ~ Building Tuning and Deploying Models
- ~ Analyzing Bike Sharing Trends
- ~ Analyzing Movie Reviews Sentiment
- ~ Customer Segmentation and Effective Cross Selling
- ~ Analyzing Wine Types and Quality
- ~ Analyzing Music Trends and Recommendations
- ~ Forecasting Stock and Commodity Prices

## => Statistics :

- ~ Descriptive Statistics
- ~ Sample vs Population statistics Random Variables
- ~ Probability distribution function Expected value
- ~ Binomial Distribution
- ~ Normal Distribution z score
- ~ Central limit Theorem
- ~ Hypothesis testing Z Stats vs T stats
- ~ Type 1 type 2 error
- ~ Confidence interval
- ~ Chi Square test
- ~ ANOVA test
- ~ F stats

## => Machine Learning 1 :

- ~ Introduction
- ~ Supervised , Unsupervised, Semi supervised, Reinforcement Train , Test, Validation Split
- ~ Performance Overfitting , underfitting OLS.
- ~ Linear Regression assumption.
- ~ R square adjusted
- ~ R square Intro to Scikit learn
- ~ Training methodology
- ~ Hands on linear regression
- ~ Ridge Regression
- ~ Logistics regression
- ~ Precision Recall ROC curve
- ~ F Score

## => Machine Learning 2 :

- ~ Decision Tree Cross
- ~ Validation Bias vs Variance
- ~ Ensemble approach Bagging
- ~ Boosting Random
- ~ Forest Variable Importance

## => Machine Learning 3 :

- ~ XGBoost
- ~ Hands on XgBoost
- ~ K Nearest Neighbour
- ~ Lazy learners
- ~ Curse of Dimensionality
- ~ K NN Issues
- ~ Hierarchical clustering K Means
- ~ Performance measurement
- ~ Principal Component analysis
- ~ Dimensionality reduction
- ~ Factor Analysis

## => Machine Learning4 :

- ~ SVR
- ~ S V M
- ~ Polynomial Regression
- ~ Ada boost
- ~ Gradient boost
- ~ Gaussian mixture
- ~ Anomaly detection
- ~ Novelty detection algorithm Stacking
- ~ K NN regressor
- ~ Decision tree regressor DBSCAN

## => Natural Language Processing :

- ~ Text Analytics
- ~ Tokenizing , Chunking
- ~ Document term
- ~ Matrix TFIDF
- ~ Sentiment analysis hands on

## => Spark :

- ~ Spark overview.
- ~ Spark installation.
- ~ Spark RDD.
- ~ Spark dataframe .
- ~ Spark Architecture.
- ~ Spark ML lib.
- ~ Spark Nlp
- ~ Spark linear regression.
- ~ Spark logistic regression.
- ~ Spark Decision Tree.
- ~ Spark Naive Bayes

- ~ Spark xg boost
- ~ Spark time series.
- ~ Spark Deployment in local server
- ~ Spark job automation with scheduler.

#### => Deep Learning :

- ~ Deep Learning Introduction.
- ~ Neural Network Architecture.
- ~ Loss Function.
- ~ Cost Function.
- ~ Optimizers.
- ~ CNN architecture.
- ~ Build First Classifier in CNN.
- ~ Deploy Classifier over cloud.
- ~ RNN overview.
- ~ GRU.
- ~ LSTM.
- ~ Time Series using RNN LSTM.
- ~ Customer Feedback analysis using RNN LSTM.

#### => Time Series :

- ~ Arima
- ~ Sarima .
- ~ Auto Arima
- ~ Time series using RNN LSTM .
- ~ Prediction of NIFTY stock price.

#### => Deployment :

- ~ Deployment of all the project In cloudfoundary , AWS AZURE and Google cloud platform
- ~ Expose api to web browser and mobile application retraining a pproach of Machine learning model
- ~ Devops infrastructure for machine learning model
- ~ Data base integration and scheduling of machine learning model and retraining c ustom machine learning training approach.
- ~ AUTO ML
- ~ Discussion on infra cost and data volume
- ~ P rediction based on streaming data

#### => Extra session :

- ~ Discussion on project explanation in interview
- ~ Data scientist roles and responsibilities
- ~ Data scientist day to day work
- ~ Companies which hire a data scientist
- ~ Resume discussion with our team one to one

#### => Tableau and power Bi self placed session :

- ~ Business Intelligence (BI) Concepts.
- ~ Microsoft Power BI (MSPBI) introduction.
- ~ Connecting Power BI with Different Data sources.
- ~ Power Query for Data Transformation.
- ~ Data Modelling in Power BI.
- ~ Reports in Power BI Reports and Visualisation types in Power BI.
- ~ Dashboards in Power BI.
- ~ Data Refresh in Power BI.
- ~ Traditional Visualisation(Excel) vs Tableau.
- ~ About Tableau.
- ~ Tableau vs Other BI Tool Pricing.

#### => Tableau Interview Questions.

### Project details :-

#### => Python project :

- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis
- ~ Integration with web portal
- ~ Integration with rest a A pi W eb portal and Mongo DB on Azure
- ~ Deployment on web portal on Azure
- ~ Text mining
- ~ Social media data churn

#### => Chatbot Project :

- ~ Chatbot using Microsoft Luis
- ~ Chatbot using google Dialog flow
- ~ Chatbot using Amazon Lex
- ~ Chatbot using Rasa NLU
- ~ Deployemnt of chatbot with web , Telegram , Whatsapp , Skype

#### => Machine learning project :

- ~ Healthcare analytics prediction of medicines based on FIT BITband
- ~ Revenue forecasting for startups
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ Anamoly detection in inventory packaged material.
- ~ Fault detection in wafferes based on sensordata
- ~ Demand forecasting for FMCG product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with Zomato dataset.
- ~ Fault detection in wafferes based on sensor data.
- ~ Cement\_Strength \_ reg.
- ~ Credit Card Fraud.
- ~ Forest\_Cover\_Classification .
- ~ Fraud Detection.
- ~ Income Prediction.

- ~ *Mushroom classifier., Phising Classifier , Thyroid\_Detection .*
- ~ *Visibility climate.*

=> Deep Learning projects :

- ~ *Customer Feedback analysis using RNN LSTM.*
- ~ *Family member detection.*
- ~ *Industry financial growth prediction.*
- ~ *Speech recognition based attendance system.*
- ~ *Vehicle Number plate detection and recognition system.*

=> Tableau and power Bi Projects :

- ~ *Project 1. Project Sales.*
- ~ *Project 2. Financial Report.*
- ~ *Project 3. HealthCare.*
- ~ *Project 4. Procurement Spend Analysis.*
- ~ *Project 5. Human Resource Tableau*

# Class 10 Biology

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10-Biology>

## Course Description :-

Biology is the study of life. It is the study of living organisms and how they interact with the environment. Biology recognizes the cell as the basic unit of life, genes as the unit of heredity and evolution as an engine which boosts the formation of new species. The study of life has helped in shaping the world. It has credible answers to why things happen in a scientific manner. iNeuron allows you to explore all the content from NCERT physics in a proper manner and flow.

## Course Features :-

- => Self paced video session
- => Covered entire class 10th Biology syllabus
- => Solved questions chapter wise
- => Notes
- => Previous year solved questions

## What you will learn :-

- => Entire NCERT Class 10th Biology Syllabus
- => Chapter wise solution with detailed explanation

## Requirements :-

- => Computer with Internet Connectivity

## Curriculum details :-

### => LIFE PROCESSES :

- ~ Nutrition & its types
- ~ Respiration
- ~ Transport
- ~ Excretion

### => CONTROL & COORDINATION :

- ~ I. Tropic movements in plants
- ~ II. Introduction of plant hormones
- ~ III. Control and co-ordination in animals
- ~ IV. Nervous system
- ~ V. Voluntary, involuntary and reflex action
- ~ VI. Chemical co-ordination
- ~ VII. Animal Hormones

### => HOW DO ORGANISMS REPRODUCE :

- ~ I. Reproduction in animals and plants (asexual and sexual)
- ~ II. Types / Modes of Reproduction
- ~ III. Safe sex vs HIV/AIDS
- ~ IV. Child bearing and womens health

### => HEREDITY & EVOLUTION :

- ~ I. Heredity
- ~ II. Mendels contribution- Laws for inheritance of traits
- ~ III. Basic concepts of evolution
- ~ IV. Evolution
- ~ V. Sex determination
- ~ VI. Speciation

### => OUR ENVIRONMENT :

- ~ I. Eco-system
- ~ II. Environmental problems
- ~ III. Ozone depletion
- ~ IV. Waste production and their solutions
- ~ V. Biodegradable and non-biodegradable substances

### => SUSTAINABLE MANAGEMENT OF NATURAL RESOURCES :

- ~ I. Conservation and judicious use of natural resources
- ~ II. Forest and wild life
- ~ III. Coal and Petroleum conservation
- ~ IV. Big dams - advantages and limitations; alternatives
- ~ V. Harvesting of water
- ~ VI. Sustainability of natural resources



# Full Stack Java Developer

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Full-Stack-Java-Developer>

## Course Description :-

The Full Stack Java Developer Job Guarantee Program offers a comprehensive set of software development skills. This one-of-a-kind industry curriculum will help you learn the entire Full Stack Java Development process. Create industry-ready projects and be prepared to land opportunities in top organisations.

## Course Features :-

- => Full stack Java Developer certification
- => Job guarantee Program
- => Online Instructor-led learning: Live teaching by instructors
- => 250+ hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing through mail and discussion forum
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building
- => Career guidance
- => Interview Preparation
- => Regular assessment
- => Mock Interview
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certification

## What you will learn :-

- => Fundamentals of Programming
- => Core Java (Detailed)
- => JDBC
- => JEE (Servlets, JSP, and Thymleaf )
- => Hibernate and JPA specifications
- => Spring Core
- => Spring Boot
- => Spring JDBC
- => Spring ORM
- => Spring Data JPA
- => Spring AOP
- => Spring MVC
- => Spring REST
- => Microservices and Realtime tools(Maven, Gradle, Log4J, Junit, Splunk, Putty, Jacacco)
- => Docker and Kubernetes
- => Agile and Scrum
- => Git and Github
- => HTML and CSS
- => Javascript

- => React js
- => SQL - Mysql
- => NoSQL - MongoDB

### Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

### Instructors :-

=> Navin Reddy :

~ I am Corporate Java trainer. Since past few years successfully trained many professionals at JP Morgan, Accenture, Polaris and L&T infotech. My youtube channel "Telusko" presently has 1.7 million subscribers. Passionate about Java Technology for over a decade and moved on as a corporate trainer. I am certified blockchain developer and Currently, building Applications running on Blockchain (dapps).

=> Hyder Abbas :

~ Corporate Software Development Trainer with a demonstrated track record of success in the IT and Ed-tech industries. I started my career as a software developer and have since taught Java, Python, Javascript to hundreds of IT enthusiasts, including corporate professionals, throughout the years. I have been developing software for over 6 years.

=> Nitin M :

~ I began working for a multinational corporation as a developer, but teaching has always been my passion. I shifted to education technology and have five years of expertise instructing both college freshmen and Corporate Employees. My interests include Java, JEE, and frameworks, and I have developed numerous applications using SpringBoot and microservices. Last but not least, I want to code as well as teach and continue to teach forever.

### Curriculum details :-

=> Induction of Course :

~ Introduction to course and Q&A

=> Git and Github :

- ~ Git foundation
- ~ Setting, maintaining and tracking git repos
- ~ Git snapshots
- ~ Git for team management
- ~ Git branches
- ~ Git merging
- ~ Git and Github ecosystem

=> Software Installation :

- ~ Download and Install Java
- ~ Download and Install Eclipse
- ~ Download and Install Visual Studio Code

=> Fundamentals of Java :

- ~ Introduction to Programming
- ~ Basic Understanding of a Computer
- ~ Basic feature of Java
- ~ Main method
- ~ Classes and Objects(Basics)
- ~ Statically typed vs Dynamically typed Programming Language
- ~ Variables and Data type in Java
- ~ Naming Convention
- ~ Identifiers

=> Operators and Loops :

- ~ Operators in Java
- ~ Incrementation and Decrementation
- ~ Conditional statement
- ~ Ternary operator
- ~ Switch case
- ~ Loops intro
- ~ for - while - do while
- ~ More on loops
- ~ Scanner class and User input in Java
- ~ Pattern programs
- ~ Nested loops

=> OOps Fundamentals :

- ~ Object creation
- ~ Instance variable vs Local variables
- ~ Methods with memory maps (JVM data areas)
- ~ Method overloading

=> Mini Project :

- ~ Guesser Game Project

=> Array in Java :

- ~ Why array?
- ~ What is an Array?
- ~ How to create an array
- ~ 1D, 2D, 3D and Regular Array & Jagged Array with memory map
- ~ Buffer overrun and ArrayIndexOutOfBoundsException
- ~ Disadvantages of Array in Java
- ~ Few basic programming questions
- ~ Bubble Sort
- ~ Selection Sort

- ~ Merge Sort
- ~ Linear Search
- ~ Binary Search

=> String in Java :

- ~ String Introduction
- ~ Types of string
- ~ Immutable string
- ~ Ways to compare and memory map String constant pool
- ~ Inbuilt methods in String class
- ~ Concatination
- ~ Few Programming questions discussion
- ~ (Reversing String, Palindrome, Anagram, Pangram ....)
- ~ Mutable String
- ~ String Buffer vs String Builder
- ~ Inbuilt Methods

=> Static Keyword :

- ~ Static keyword
- ~ Class loading
- ~ Execution of a Java Program
- ~ static variables, static methods, static block
- ~ Differences b/w Non static and static

=> Encapsulation :

- ~ Need of Encapsulation
- ~ What is Encapsulation?
- ~ Private members
- ~ Shadowing problem and this keyword
- ~ Setters & Getters
- ~ Constructor
- ~ this()

=> Inheritance :

- ~ Inheritance introduction
- ~ extends keyword
- ~ Types of Inheritance
- ~ Important key points(5 keypoints)
- ~ Inherited methods, Overridden methods, Specialized methods
- ~ Rules to override method

=> Polymorphism and Abstraction :

- ~ What is polymorphism ?
- ~ How to achieve polymorphism
- ~ Runtime vs Compile time polymorphism
- ~ Abstract keyword and Abstraction
- ~ Abstract class and Abstract method

=> Final keyword in Java :

- ~ final class
- ~ final variable
- ~ final method

=> Interface :

- ~ What is interface
- ~ Need of Interface
- ~ Different use cases of Interface
- ~ Abstract vs interface
- ~ Additional features of Interface

=> Lambda Expression :

- ~ Functional Interface
- ~ What is Lambda Expression
- ~ Different ways to create Lambda Expression
- ~ Lambda Expression excercises

=> Exception Handling :

- ~ What is an Exception?
- ~ How to handle Exception (try catch)
- ~ Multiple catch block
- ~ Handling vs Ducking an Exception
- ~ Hierarchy of an Exception class
- ~ throw & throws keyword and Custom Exception
- ~ try with Resources

=> Core Java Project :

- ~ Assignment with mentor guidance - Food Delivery App

=> Multi-threading :

- ~ What is Thread & Need of multiple Threads
- ~ How to create multiple Threads
- ~ run() method
- ~ Race condition
- ~ Different states of Thread
- ~ Dead lock

=> Collection in Java :

- ~ Why Collection ?
- ~ ArrayList
- ~ LinkedList
- ~ PriorityQueue
- ~ ArrayDeque

- ~ TreeSet
- ~ HashSet
- ~ LinkedHashSet
- ~ Collection Hierarchy
- ~ Map
- ~ Map heirarchy
- ~ Stream API in Java

=> Annotations in Java :

- ~ Enums
- ~ What is Annotation
- ~ In Built Annotation
- ~ Custom Annotation

=> File Handling in Java :

- ~ Input Stream
- ~ Output Stream
- ~ File Operation in Java
- ~ Serialization
- ~ Deserialization

=> SQL - MySQL :

- ~ Basic Concepts of Advantages of DBMS
- ~ Exploring Relational DBMS
- ~ E-R Modeling and Diagram
- ~ Normalization
- ~ Introduction to SQL
- ~ DDL and DML Statements
- ~ Working with Queries (DQL)
- ~ CRUD operations
- ~ Aggregate Functions
- ~ Joins and Set Operations
- ~ Working with Constraints

=> MongoDB :

- ~ What is mongoDB
- ~ How does mongoDB works
- ~ What is mocha and need of mocha in mongodb
- ~ Big umbrella of MongoDB
- ~ How to install mongoDB on MAC
- ~ How to install mongoDB on Windows
- ~ Create and Read operation in MongoDB
- ~ ObjectId and BSON in mongoDB
- ~ CRUD operations in mongoDB
- ~ UpdateOne and DeleteOne in #mongoDB
- ~ UpdateMany and deleteMany in mongoDB
- ~ Database issues with Update in mongodb
- ~ Getting more data in #mongodb
- ~ Understanding objects structure in mongoDB
- ~ What is schema in mongoDB

=> JDBC :

- ~ Steps followed to write JDBC Code
- ~ Usage of Statement Object
- ~ Usage of Prepared Statement
- ~ Types of Driver available
- ~ Application using Statement and PreparedStatetement

=> Project - JDBC :

- ~ CRUD operation applicationin layered approach of Student table using Factory Desgin Pattern

=> HTML and CSS :

- ~ Collecting and installing developers tool
- ~ Structuring the files and creating first file
- ~ Text tags
- ~ List items
- ~ Divisions and Spans
- ~ Images and links
- ~ Challenge for links on images and solution
- ~ Tables in HTML
- ~ More about forms in HTML
- ~ Comparing HTML 4 semantics with HTML 5
- ~ Introduction to css and where to write it
- ~ Solving the color selection problem
- ~ Comming soon template and backgrounds
- ~ Box model and centering text
- ~ Google fonts and font awesome
- ~ Styling the links
- ~ Classes and ID in CSS
- ~ Designing a navigation bar from scratch
- ~ Color palletes and canva for design
- ~ Gradients in css
- ~ Check through css
- ~ box sizing in css

=> JavaScript :

- ~ What are JavaScript engines
- ~ What ES version of JavaScript is good for us
- ~ Variable and datatypes in JavaScript
- ~ Operators in JavaScript
- ~ What are conditionals in JavaScript

- ~ Logical conditional Login in JavaScript
- ~ Ternary operator in JavaScript
- ~ Switch for role-based access in JavaScript
- ~ Basics of functions in JavaScript
- ~ Functions in variable User Role in JavaScript
- ~ Understand the context in JavaScript
- ~ Code hoisting in JavaScript
- ~ Scope chaining in JavaScript
- ~ Light intro to THIS in JavaScript
- ~ Maps in JavaScript
- ~ Classes and module exports in JavaScript
- ~ Private props getters and setters in JavaScript
- ~ Inheritance in JavaScript
- ~ Event loop Will JavaScript wait
- ~ Promise async and await in JavaScript

=> React js :

- ~ What is react and myths
- ~ Tools that we need
- ~ Introduction of Virtual DOM.
- ~ Difference between JS and JSX.
- ~ React Components overview
- ~ Containers and components
- ~ Child Components
- ~ Namespaced components
- ~ JavaScript expressions available in JSX
- ~ Node setup
- ~ How to use NPM?
- ~ How to create package.json and purpose of it
- ~ Best IDE for React JS and How to write optimized code in React JS?
- ~ React JS browser plugins overview.
- ~ Create a React component with JSX template.
- ~ How to create Nested Components?
- ~ What is React JS render?
- ~ React Props overview.
- ~ Introduction of Props validation with data types.
- ~ Flow of States, Initialize states and update states.
- ~ Lists of Form components.
- ~ Setup Controlled and Uncontrolled form components.
- ~ Control Input elements.
- ~ How to set default values on all formats of Input elements.

React JS Form validations.

- ~ How to write Styles?
- ~ Initial Render
- ~ Props Change
- ~ Stage Change
- ~ Component willMount
- ~ Component didMount
- ~ Component Unmount
- ~ Overview of a single-page application.
- ~ How is React Router configured?
- ~ Background of Router
- ~ How Should Conditional Statements Be Handled in JSX?
- ~ onBlur, onKeyUp, onChange and other useful primary events in React JS.
- ~ How to Sharing events between the components?
- ~ Introduction to styled components
- ~ Styling the application using styled component
- ~ How to Load the router library?
- ~ Configure the React Router?
- ~ How to Pass and receive parameters?
- ~ Understanding Hooks
- ~ The useState hook
- ~ Side effects using the useEffect hook
- ~ The useContext hook
- ~ The useReducer hook
- ~ Writing your own hook
- ~ The React ecosystem

=> Servlet :

- ~ Types of application
- ~ Client Server Architecture
- ~ Different types of Server a. web server b. application server
- ~ Need of Servlet and Different ways of Creating a Servlet
- ~ Configuring Servlet in
- ~ XML and Annotation support
- ~ Difference b/w ServletConfig vs ServletContext object
- ~ HttpServletRequest, HttpServletResponse, RequestDispatching
- ~ SessionTracking Mechanism
- ~ HttpSessionTracking
- ~ Cookie
- ~ URL ReWriting
- ~ Hidden form Field
- ~ Filters,Listeners and One CRUD app using MVC Design pattern
- ~ Need of JSP, Usage of JSP, Implicit Objects
- ~ Type of Directives
- ~ Expression Language, JSTL Tags
- ~ MVC CRUD APP using Servlet, JSP

=> Project - JEE :

~ Building CustomerRelationship manager System using JDBC,Servlets and JSP and JSTL

=> Introduction to ORM(Hibernate and JPA Specifications) :

- ~ Drawbacks of JDBC
- ~ Hiberante
- ~ Advantages of Hibernate compared to JDBC
- ~ Introduction.
- ~ ORM (Object Relational Mapping)
- ~ Configuration xml file and Mapping xml file along with dtlds.
- ~ Hibernate architecture
- ~ Installation and Directory Structure
- ~ Hibernate Data Types.
- ~ First Application using Hibernate.
- ~ Hibernate API
- ~ CRUD operations
- ~ Primary key Generators
- ~ Hibernate Query Language (HQL)
- ~ Native SQL
- ~ Criteria API
- ~ Inheritance in Hibernate
- ~ Relations
- ~ (one to one, one to many, many to one, many to many)
- ~ Caching
- ~ Connecting with Multiple Databases
- ~ Integrating Hibernate with Servlets,JSP and with Spring
- ~ Hibernate Annotations
- ~ Performing BLOB/CLOB operation, Insertion of Date and Time to Database
- ~ Performing Object versioning TimeStamping and life cycle events of hibernate
- ~ ConnectionPooling in hibernate

=> Project - Hibernate :

~ Building CustomerRelationship manager System using ORM,Servlets and JSP and JSTL

=> SPRING BOOT :

- ~ What is Spring Framework
- ~ What is Spring Boot
- ~ Differences between Spring & Spring Boot
- ~ IOC container
- ~ Dependency Injection a) Setter Injection b) Constructor Injection c) Field Injection
- ~ Stereotype Annotations a) @Component b) @Service c) @Repository d) @Controller e) @Indexed
- ~ Spring Boot Overview
- ~ Pros & Cons of Spring Boot
- ~ Approaches to create Spring Boot Application
- ~ Spring Initializer (start.spring.io)
- ~ Spring Starter Wizard in STS IDE
- ~ Introduction to Spring Boot Starters
- ~ Spring Boot Parent Starter
- ~ Spring-boot-starter
- ~ Spring-boot-starter-web
- ~ Spring-boot-starter-webflux
- ~ Spring-boot-starter-data-jpa
- ~ Spring-boot-devtools
- ~ Spring-boot-starter-mail
- ~ Spring-boot-actuator
- ~ Spring-boot-starter-test etc.
- ~ What is Start Class in Spring Boot
- ~ @SpringBootApplication annotation internals
- ~ SpringApplication.run(..) method internals
- ~ Spring Boot Application Boot strapping
- ~ AutoConfiguration in Spring Boot

=> SPRING DATA JPA :

- ~ What is Persistence Layer
- ~ Best practises to follow in persistence layer
- ~ ORM Basics
- ~ Spring Data JPA Introduction
- ~ Differences between Spring ORM and Spring Data
- ~ CrudRepository introduction
- ~ CrudRepository methods for DB operations
- ~ Custom findByXXX method syntax
- ~ Custom Queries Execution in Data JPA
- ~ JpaRepository introduction
- ~ JpaRepository methods for DB operations
- ~ Pagination Using Data JPA methods
- ~ Sorting Using Data JPA Methods
- ~ Query By Example Executor
- ~ Generators
- ~ Custom Generators in Spring Data
- ~ Embedded Database Introduction
- ~ Embedded Database vs External Database
- ~ Application Development using Embedded Database (H2)
- ~ Application Development Using MYSQL Database
- ~ Application Development Using PostGreSQL Database
- ~ Application Development Using MongoDB
- ~ profiles in springboot

=> SPRING WEB MVC :

- ~ Spring Web MVC Introduction
- ~ Spring Web MVC Advantages
- ~ Spring MVC Architecture

- ~ Introduction to Front Controller
- ~ Controllers
- ~ Handler Mappers
- ~ View Resolvers
- ~ Web Application development using Spring Boot
- ~ Embedded HTTP Servers Introduction
- ~ a) Embedded Tomcat Server b) Embedded Jetty Server c) Embedded Undertow Server
- ~ Making Jetty as Default server
- ~ Web Application Deployment in External Server
- ~ Sending Data From UI to Controller
- ~ a) Query Param b) Path Param
- ~ Sending Data From Controller to UI a) Model b) ModelAndView
- ~ @RequestBody annotation 38) @ResponseBody annotation
- ~ Introduction to Spring MVC Form Tag library
- ~ Form Based application development using Spring Boot
- ~ Thymeleaf Introduction
- ~ Web Application with Thymeleaf
- ~ Sending Email using Spring Boot
- ~ Exception Handling in Spring Boot Web Application
- ~ Spring Boot Actuators
- ~ a) Health b) Info c) Heapdump d) Threaddump
- ~ e) Beans f) Httptrace g) Mappings h) Shutdown etc
- ~ Unit Testing for Spring Boot Application using JUnit with Mocking
- ~ Code Coverage using Jacoco

#### => SPRING REST :

- ~ Distributed Applications
- ~ Distributed Technologies
- ~ SOAP vs REST
- ~ RESTful Services Introduction
- ~ REST principles
- ~ XML
- ~ One Time operations
- ~ Run Time Operations a) Marshalling b) Un Marshalling
- ~ JAX-B Introduction JAX-B Architecture
- ~ Applications development with JAX-B
- ~ JSON Introduction
- ~ XML vs JSON
- ~ JACKSON API
- ~ Converting Java object to JSON and vice versa using Jackson API
- ~ GSON API
- ~ Converting Java Object to JSON and Vice Versa using GSON API
- ~ HTTP Protocol Details
- ~ HTTP Methods a) GET b) POST c) PUT d) DELETE
- ~ HTTP Status Codes
- ~ @RestController
- ~ @RequestBody
- ~ @ResponseBody
- ~ @RequestParam
- ~ @PathVariable
- ~ Media Types
- ~ Consumes
- ~ Produces
- ~ Accept Header
- ~ Content-Type head
- ~ REST API Development using Spring Boot
- ~ POSTMAN
- ~ SWAGGER & SWAGGER UI
- ~ Exception Handling in REST API
- ~ REST Security
- ~ a) HTTP Basic Auth
- ~ b) JWT
- ~ c) OAuth2.0
- ~ Mono Objects
- ~ Flux Objects
- ~ REST Client Introduction
- ~ RestTemplate
- ~ WebClient
- ~ RestTemplate vs WebClient
- ~ Reactive Programming
- ~ Synchronous vs Asynchronous Calls
- ~ Apache Kafka Integration with Spring Boot
- ~ Redis Cache Integration with Spring Boot

#### => Spring Boot Projects :

- ~ Building Student management System using SpringBoot
- ~ Building CustomerRelationship manager System using SpringMVC and Thymeleaf
- ~ Working with TicketManagement application using Spring datajpa and Spring ReSt with swagger integration

#### => Docker :

- ~ Docker & its architecture
- ~ Docker as a service
- ~ Docker CLI
- ~ Docker Volumes
- ~ Dockerizing a web application

#### => MICROSERVICES :

- ~ Monolith Architecture case study
- ~ Monolith Application Deployment Process
- ~ Load balancer (Cluster) case study

- ~ *Load Balancing Algorithms*
- ~ *a) Round Robin*
- ~ *b) IP Hashing*
- ~ *c) Sticky Session*
- ~ *Monolith Architecture Drawbacks*
- ~ *Micro services Introduction*
- ~ *Micro Services Advantages*
- ~ *Micro Services Dis-Advantages*
- ~ *Micro Services case study*
- ~ *Identifying Micro services boundaries*
- ~ *Micro services Architecture*
- ~ *Micro services Development*

=> *Agile and Scrum :*

- ~ *What is Agile?*
- ~ *What is Scrum?*
- ~ *Benefits of Agile*
- ~ *Scrum Artifacts*

=> *Final Project 1 :*

- ~ *Building StockMarket API integration with Eureka Client and hosting in PCF*

=> *Final Project 2 :*

- ~ *Buidling BookStock AP integration with MongoDB and making it as Eureka Client with swagger integration*

=> *Final Project 3 :*

- ~ *Capstone project of Insurance application which holds microservices and React integration*



# Complete Interview Preparation For Data Structure and Algorithm in Python

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA INTERVIEW

Course link : <https://ineuron.ai/course/Complete-Interview-Preparation-For-Data-Structure-and-Algorithm-in-Python>

## Course Description :-

This course will teach you about different data structures and algorithms. In addition, you'll be able to learn how to arrange your replies in-depth explanations using Python and discover how to understand frequently-asked technical interview questions.

## Course Features :-

- => 400+ Interview DSA questions with explanation and coding files
- => Explanation of every data structure and intuition behind algorithms
- => Understand which algorithms to use for a particular problem statement
- => Learn how to approach a problem and explain it to an interviewer
- => Build confidence for the development related interviews

## What you will learn :-

- => Arrays
- => Linked Lists
- => Matrix
- => Backtracking
- => Bit Manipulation
- => Binary Tree
- => Binary Search Tree
- => Divide and Conquer
- => Dynamic Programming
- => Graph
- => Heap
- => Queue
- => Stack
- => Sorting
- => String
- => Trie

## Requirements :-

- => A System with Internet Connection
- => Consistency and Dedication

## Instructors :-

- => Vaibhav Kumar :  
~ Keen Problem solver and a Competitive coder, Ex-Gainsight

## Curriculum details :-

- => Array :
  - ~ Find pair with given sum in the array Preview
  - ~ Check if subarray with 0 sum is exists or not Preview
  - ~ Print all sub-arrays with 0 sum
  - ~ Sort binary array in linear time
  - ~ Find a duplicate element in a limited range array
  - ~ Find maximum length sub-array having given sum
  - ~ Find maximum length sub-array having equal number of 0s and 1s
  - ~ Find maximum product of two integers in an array
  - ~ Sort an array containing 0s, 1s and 2s (dutch national flag problem)
  - ~ In place merge two sorted arrays
  - ~ Merge two arrays by satisfying given constraints
  - ~ Find index of 0 to replace to get maximum length sequence of continuous ones
  - ~ Shuffle a given array of elements (fisheryates shuffle)
  - ~ Rearrange the array with alternate high and low elements
  - ~ Find equilibrium index of an array
  - ~ Find largest sub-array formed by consecutive integers

- ~ Find majority element (boyermoore majority vote algorithm)
- ~ Move all zeros present in the array to the end
- ~ Replace each element of array with product of every other element without using / operator
- ~ Find longest bitonic subarray in an array
- ~ Longest increasing subsequence
- ~ Find maximum difference between two elements in the array by satisfying given constraints
- ~ Maximum sum subarray problem (kadaness algorithm)
- ~ Print continuous subarray with maximum sum
- ~ Maximum sum circular subarray
- ~ Find all distinct combinations of given length I
- ~ Find all distinct combinations of given length with repetition allowed
- ~ Find maximum sequence of continuous 1s formed by replacing at-most k zeroes by ones
- ~ Find minimum sum subarray of given size k
- ~ Find maximum product subarray in a given array
- ~ Find subarray having given sum in given array of integers
- ~ Find the length of smallest subarray whose sum of elements is greater than the given number
- ~ Find largest number possible from set of given numbers
- ~ Find the smallest window in array sorting which will make the entire array sorted
- ~ Find maximum sum path involving elements of given arrays
- ~ Maximum profit earned by buying and selling shares any number of times
- ~ Trapping rain water within given set of bars
- ~ Find minimum platforms needed in the station so to avoid any delay in arrival of any train
- ~ Decode the array constructed from another array
- ~ Sort an array using one swap
- ~ Find triplet with given sum in an array
- ~ Length of longest continuous sequence with same sum in given binary arrays
- ~ Reverse every consecutive m elements of the given subarray
- ~ Maximum product subset problem
- ~ Find pairs with given difference k in the array
- ~ Find pairs with given difference k in the array | constant space solution
- ~ 4 sum problem | quadruplets with given sum
- ~ Print all quadruplets with given sum | 4-sum problem extended
- ~ Quickselect algorithm
- ~ Rearrange array such that  $a[a[i]]$  is set to i for every element  $a[i]$
- ~ Print all triplets that forms arithmetic progression
- ~ Print all triplets that forms geometric progression
- ~ Print all combination of numbers from 1 to n having sum n
- ~ Replace each element of the array by its corresponding rank in the array
- ~ Print all triplets in an array with sum less than or equal to given number
- ~ Group elements of an array based on their first occurrence
- ~ Find minimum difference between index of two given elements present in the array
- ~ Find maximum absolute difference between sum of two non-overlapping sub-arrays
- ~ Find all symmetric pairs in an array of pairs
- ~ Partition an array into two sub-arrays with the same sum
- ~ Find count of distinct elements in every sub-array of size k
- ~ Find two numbers with maximum sum formed by array digits
- ~ Print all sub-arrays of an array having distinct elements
- ~ Find a triplet having maximum product in an array
- ~ Find minimum index of repeating element in an array
- ~ Generate random input from an array according to given probabilities
- ~ Find pair in an array having minimum absolute sum
- ~ Find index of maximum occurring element with equal probability
- ~ Check if an array is formed by consecutive integers
- ~ Find two non-overlapping pairs having same sum in an array
- ~ Add elements of two arrays into a new array
- ~ Find minimum product among all combinations of triplets in an array
- ~ Replace every element of an array with the least greater element on its right
- ~ Find all odd occurring elements in an array having limited range of elements
- ~ Count the distinct absolute values in the sorted array
- ~ Print all combinations of positive integers in increasing order that sum to a given number
- ~ Find all distinct combinations of given length II
- ~ Find subarrays with given sum in an array
- ~ Find the surpasser count for each element of an array
- ~ Find maximum length sequence of continuous ones (using sliding window)
- ~ Find maximum length sequence of continuous ones
- ~ Find index that divides an array into two non-empty subarrays of equal sum
- ~ Calculate frequency of all elements present in an array of specified range
- ~ Rearrange the array such that it contains positive and negative numbers at alternate positions
- ~ Find a sorted triplet in the given array
- ~ Shuffle an array according to the given order of elements
- ~ Count number of strictly increasing sub-arrays in an array
- ~ Find duplicates within given range k in an array
- ~ Longest alternating subarray problem
- ~ Find minimum range with at-least one element from each of the given arrays
- ~ Find longest subsequence formed by consecutive integers
- ~ Find all elements in an array that are greater than all elements present to their right
- ~ Find missing number in array without using extra space
- ~ Determine index of an element in given array which satisfies given constraints
- ~ Find minimum moves required for converting a given array to an array of zeroes
- ~ Left rotate an array
- ~ Right rotate an array k times
- ~ Find maximum profit earned from at most two stock transactions
- ~ Find frequency of each element in a sorted array containing duplicates
- ~ Find minimum and maximum element in an array using minimum comparisons
- ~ Difference between subarray, subsequence and subset
- ~ Find odd occurring element in an array in single traversal
- ~ Find odd occurring element in logarithmic time
- ~ Find two odd occurring elements in an array without using any extra space

- ~ Check if given array represents min heap or not
- ~ Find kth smallest element in an array
- ~ Find kth largest element in an array
- ~ Sort a k-sorted array
- ~ Merge M sorted lists of variable length
- ~ Find smallest range with at-least one element from each of the given lists
- ~ Merge M sorted lists each containing N elements
- ~ Find maximum sum of subsequence with no adjacent elements
- ~ Find ways to calculate a target from elements of specified array
- ~ Sort elements by their frequency and index
- ~ Sort an array based on order defined by another array
- ~ Inversion count of an array
- ~ Segregate positive and negative integers in linear time
- ~ Find number of rotations in a circularly sorted array
- ~ Search an element in a circular sorted array
- ~ Find first or last occurrence of a given number in a sorted array
- ~ Count occurrences of a number in a sorted array with duplicates
- ~ Find smallest missing element from a sorted array
- ~ Find floor and ceil of a number in a sorted array
- ~ Search in a nearly sorted array in logarithmic time
- ~ Find number of 1s in a sorted binary array
- ~ Find missing term in a sequence in logarithmic time
- ~ Find missing number and duplicate elements in an array
- ~ Find the peak element in an array
- ~ Find floor and ceil of a number in a sorted array (recursive solution)
- ~ Print all distinct subsets of a given set
- ~ Find two duplicate elements in a limited range array (using XOR)

=> Linked list :

- ~ Introduction to linked lists
- ~ Linked list implementation
- ~ Linked list | insertion at tail
- ~ Static linked list
- ~ Clone given linked list
- ~ Delete linked list
- ~ Pop operation in linked list
- ~ Insert given node into the correct sorted position in the given sorted linked list
- ~ Rearrange linked list in increasing order (sort linked list)
- ~ Split the nodes of the given linked list into front and back halves
- ~ Remove duplicates from a sorted linked list
- ~ Move front node of the given list to the front of the another list
- ~ Move even nodes to the end of the list in reverse order
- ~ Split given linked list into two lists where each list containing alternating elements from it
- ~ Construct a linked list by merging alternate nodes of two given lists
- ~ Merge sort algorithm for singly linked list
- ~ Merge two sorted linked lists into one
- ~ Merge K sorted linked lists
- ~ Intersection of two given sorted linked lists
- ~ Reverse linked list (iterative solution)
- ~ Reverse linked list (recursive solution)
- ~ Reverse every group of k nodes in given linked list
- ~ Find kth node from the end in a linked list
- ~ Merge alternate nodes of two linked lists into the first list
- ~ Merge two sorted linked lists from their end
- ~ Delete every N nodes in a linked list after skipping M nodes
- ~ Rearrange linked list in specific manner in linear time
- ~ Check if linked list is palindrome or not
- ~ Move last node to front in a given linked list
- ~ Rearrange the linked list in specific manner
- ~ Detect cycle in a linked list (floyds cycle detection algorithm)
- ~ Sort linked list containing 0s, 1s and 2s
- ~ Implement stack using linked list
- ~ Implement queue using linked list
- ~ Remove duplicates from a linked list
- ~ Rearrange the linked list so that it has alternating high, low values
- ~ Rearrange a linked list by separating odd nodes from the even ones
- ~ Calculate height of a binary tree with leaf nodes forming a circular doubly linked list
- ~ XOR linked list: overview and implementation
- ~ Convert a multilevel linked list to a singly linked list
- ~ Recursively check if linked list of characters is palindrome or not
- ~ Remove redundant nodes from a path formed by a linked list
- ~ Add a single-digit number to a linked list representing a number
- ~ Reverse every alternate group of k nodes in a linked list
- ~ Sort a doubly linked list using merge sort
- ~ Reverse a doubly linked list
- ~ Pairwise swap adjacent nodes of a linked list
- ~ Flatten a linked list
- ~ Check if a linked list of string is palindromic
- ~ Swap kth node from beginning with kth node from end in a linked list
- ~ Add two linked lists without using any extra space

=> Matrix :

- ~ Print matrix in spiral order
- ~ Create spiral matrix from given array
- ~ Shift all matrix elements by 1 in spiral order
- ~ Find shortest path from source to destination in a matrix that satisfies given constraints
- ~ Change all elements of row i and column j in a matrix to 0 if cell (i, j) has value 0
- ~ Print diagonal elements of the matrix having positive slope
- ~ Find all paths from first cell to last cell of a matrix

- ~ Replace all occurrences of 0 that are not surrounded by 1 in a binary matrix
- ~ In-Place rotate the matrix by 90 degrees in clock-wise direction
- ~ Count negative elements present in sorted matrix in linear time
- ~ Report all occurrences of an element in row wise and column wise sorted matrix in linear time
- ~ Calculate sum of all elements in a sub-matrix in constant time
- ~ Find maximum sum  $K \times K$  sub-matrix in a given  $M \times N$  matrix
- ~ Find maximum sum submatrix present in a given matrix
- ~ Count the number of islands
- ~ Flood fill algorithm
- ~ Find shortest safe route in a field with sensors present
- ~ Find all occurrences of given string in a character matrix
- ~ Shortest path in a maze | lee algorithm
- ~ Check if given matrix is toeplitz matrix or not
- ~ In-Place rotate the matrix by 180 degrees

=> Backtracking

=> Bit manipulation

=> Binary tree

=> Binary search tree

=> Divide and conquer

=> Dynamic programming

=> Graph

=> Heap

=> Queue

=> Stack

=> Sorting

=> String

=> Trie

# Pro Aptitude - DBMS

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Aptitude---DBMS>

## Course Description :-

This course is designed mostly for computer science subject Database Management Systems test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> DBMS Theoretical Test

=> DBMS Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Database Management Systems Test :

~ *DBMS Test 1*

~ *DBMS Test 2*

~ *DBMS Test 3*

~ *DBMS Test 4*

# Pro JavaScript Programming Language

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-JavaScript-Programming-Language>

## Course Description :-

This course is designed mostly for JavaScript test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> JavaScript Theoretical Test

=> JavaScript Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> JavaScript Test :

- ~ JavaScript Test 1
- ~ JavaScript Test 2
- ~ JavaScript Test 3
- ~ JavaScript Test 4
- ~ JavaScript Test 5
- ~ JavaScript Test 6
- ~ JavaScript Test 7
- ~ JavaScript Test 8
- ~ JavaScript Test 9
- ~ JavaScript Test 10
- ~ JavaScript Test 11
- ~ JavaScript Test 12
- ~ JavaScript Test 13
- ~ JavaScript Test 14

# MLOPS Live Class

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : <https://ineuron.ai/course/MLOPS-Live-Class>

## Course Description :-

Machine Learning can be found in a variety of places, including company financial performance, human health, social issues, and scientific discoveries and innovation. Machine Learning is no longer a differentiator in business; it is an issue of life. MLOPs (Machine Learning Operations) appear to improve the administration and operation of Machine Learning models and allow a smoother transition to operation, based on principles similar to DevOps, which likewise strive to combine and automate development and operation. This course will expose you to the realm of MLOps through a variety of hands-on exercises.

## Course Features :-

- => Online live classes
- => Doubt clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Learn what MLOps is and why it's important in machine learning.
- => Create Projects and Register Models with their Metrics and Hyperparameters using Python and MLOps.

## Requirements :-

- => Basic understanding of Python Language
- => Prior knowledge of Machine Learning (Not Mandatory)

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

=> AIOps introduction theory :

- ~ Introduction 1
- ~ Introduction 2
- ~ Introduction 3
- ~ Challenges
- ~ AIML generic steps
- ~ Level 0 workflow
- ~ Level 0 characteristics and observations
- ~ Level 1 workflow
- ~ Level 1 aim
- ~ Level 1 characteristics
- ~ Frequently used terms
- ~ Data validation
- ~ Model validation - Offline
- ~ Model validation - Online
- ~ Feature store
- ~ Metadata storage
- ~ Pipeline trigger
- ~ Final summary
- ~ Level 2 aim
- ~ Level 2 CI CD workflow detail discussion part 1
- ~ Level 2 CI CD workflow detail discussion part 2
- ~ Level 2 more on CI
- ~ Level 2 more on CD
- ~ Level 2 deployment types
- ~ Level 2 summary final

=> Linux introduction :

- ~ Introduction to Linux
- ~ What is Linux
- ~ Important pieces in Linux
- ~ Features of Linux

- ~ Evolution of Linux
- ~ Differences between Windows and Linux

#### => Setting up our Linux space :

- ~ Downloading necessary tools
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting up passwordless SSH

#### => Linux concepts :

- ~ What is Kernel
- ~ Types of Kernel
- ~ What is Shell
- ~ Types of Shell
- ~ Distros in Linux
- ~ Linux boot process
- ~ File system
- ~ Run levels in Linux
- ~ File types of Linux

#### => Package management :

- ~ Package management
- ~ Package managers & DPKG
- ~ Working with APT & APT GET
- ~ Apt-get advanced part - 1
- ~ Apt-get advanced part - 2

#### => Linux commands :

- ~ Linux commands part - 1
- ~ Linux commands part - 2
- ~ Linux commands part - 3
- ~ Linux commands part - 4
- ~ Cat command usages

#### => Working with terminal :

- ~ File archival
- ~ File compression
- ~ Files and patterns search
- ~ Input output redirection
- ~ Working with VI editor
- ~ Advanced VI editor part - 1
- ~ Advanced VI editor part - 2

#### => Permissions & security :

- ~ Types of account in Linux
- ~ User management
- ~ Group management
- ~ Files access controls
- ~ Linux file permissions
- ~ Modifying file ownership
- ~ Sudoers in Linux
- ~ Cronjobs
- ~ SCP
- ~ Special permissions
- ~ System management
- ~ System tools
- ~ Hard link and Soft link
- ~ Aliasing in Linux
- ~ Creating users in multiple ways

#### => Linux in AWS cloud deploy an app in EC2 :

- ~ Launching an Ubuntu VM and SSH setup
- ~ Package installation in VM
- ~ Running our calculator app
- ~ Unicorn & Nginx setup
- ~ Creating a Unicorn service
- ~ Attaching an Elastic IP
- ~ Attaching OpenSSL certificates for https

#### => GitHub introduction :

- ~ Git introduction
- ~ What is version control?
- ~ Types of version control
- ~ What is git?
- ~ Why git?
- ~ Git installation on Windows
- ~ Git installation on Linux
- ~ Git setup
- ~ Git terminologies
- ~ Repositories in GIT
- ~ Creating repository
- ~ Checking repository history
- ~ Doing commits
- ~ Git diff
- ~ Git restore
- ~ Gitignore
- ~ Tagging
- ~ Branching



- ~ Branching practicals
- ~ Merging
- ~ Merge conflicts
- ~ Remote repository
- ~ Cloning repository
- ~ Working with remote repository
- ~ Pushing to remote failed in GitHub
- ~ Personal access token setup in Windows
- ~ Personal access token setup in Linux
- ~ Pull request
- ~ Git fetch & pull
- ~ Fork
- ~ Rebasing
- ~ Interactive rebasing
- ~ Git rewrite history
- ~ Git rewrite history continued
- ~ Cherry picking
- ~ Modify recent commits
- ~ Git revert
- ~ Git checkout
- ~ Git reset
- ~ Git stash
- ~ Git reflog
- ~ Course outro

=> DVC introduction :

- ~ What is DVC?
- ~ Installation

=> Automate ML pipelines with DVC :

- ~ Workflow
- ~ Basic setup
- ~ Stage 01 implementation
- ~ Stage 01 added to dvc.yaml
- ~ Stage 02 implementation
- ~ Stage 02 added to dvc.yaml
- ~ Stage 03 implementation
- ~ Stage 03 added to dvc.yaml
- ~ Stage 04 implementation
- ~ Stage 04 added to dvc.yaml
- ~ Final update

=> Getting started with DVC :

- ~ Data versioning 01
- ~ Data versioning 02
- ~ Data versioning 03

=> Automate DL pipelines with DVC (dl-tensorflow) :

- ~ Workflow description
- ~ Creation of project skeleton
- ~ Stage 01 implementation
- ~ Stage 01 added to dvc.yaml
- ~ Stage 01 final update
- ~ Stage 02 main file creation
- ~ Stage 02 base model creation
- ~ Stage 02 python scripting
- ~ Stage 02 logging model summary in the logs
- ~ Stage 02 added to the dvc.yaml file
- ~ Stage 03 preparing directory creation for callbacks
- ~ Stage 03 adding callback utility
- ~ Stage 03 adding to dvc.yaml
- ~ Stage 04 loading binary file of callbacks
- ~ Stage 04 load untrained model and start training
- ~ Stage 04 training valid generator
- ~ Stage 04 model training added
- ~ Stage 04 save model

=> Docker :

- ~ Docker introduction
- ~ What is Docker?
- ~ Why Docker?
- ~ Benefits of Docker
- ~ What is container?
- ~ Containers vs VM
- ~ Containers vs image
- ~ Docker editions
- ~ What docker is not?
- ~ Important terminologies
- ~ Docker setup in Windows
- ~ Docker setup in Linux
- ~ Docker setup in Mac

=> Docker basic usage :

- ~ Docker basic commands part 1
- ~ Docker basic commands part 2

=> Docker run :

- ~ Docker run part 1
- ~ Docker run part 2

=> Docker images :

- ~ Docker images
- ~ Creating a new image
- ~ Environment variables
- ~ Commands & entrypoints

#### => Docker compose :

- ~ Docker compose
- ~ Voting application understanding
- ~ Docker compose versions
- ~ Docker compose networks
- ~ Voting application with docker run
- ~ Voting application with docker compose

#### => Docker concepts :

- ~ Docker engine
- ~ Docker storage
- ~ Docker networking
- ~ Docker registry

#### => Kubernetes :

- ~ Course introduction
- ~ What is Kubernetes?
- ~ Why Kubernetes?
- ~ Containers
- ~ Containers orchestration

#### => Kubernetes setup :

- ~ Kubernetes setup on Windows
- ~ Kubernetes setup in Linux
- ~ Kubernetes setup in Mac
- ~ Minikube
- ~ Kubectl
- ~ Kubernetes architecture

#### => Kubernetes concepts :

- ~ Pods
- ~ Node architecture
- ~ Replication controller
- ~ Deployments
- ~ Services
- ~ Labels
- ~ Healthchecks
- ~ Readiness probe
- ~ Pod state
- ~ Pod lifecycle
- ~ Secrets
- ~ Webui

#### => Advanced :

- ~ Service discovery
- ~ ConfigMap
- ~ Ingress controller
- ~ External DNS
- ~ Volumes
- ~ Volumes autoprovisioning
- ~ Pod presets
- ~ Statefulsets
- ~ Daemonsets
- ~ Resource usage monitoring
- ~ Autoscaling

#### => Deploying apps :

- ~ Microservices architecture
- ~ Deploying in Kubernetes
- ~ Deploying in kubernetes with deployments

#### => Packaging & deployment :

- ~ Introduction to Helm
- ~ Creating your own Helm charts
- ~ Setting up a Helm repository on S3
- ~ Building and deploying Helm charts with Jenkins

#### => MLFlow introduction :

- ~ What is MLFlow?
- ~ Installation
- ~ Where runs are recorded
- ~ How runs and artifacts are recorded
- ~ Scenario 1: MLFlow on localhost
- ~ Scenario 2: MLFlow on localhost with sqlite
- ~ Scenario 3: MLFlow on localhost with tracking server
- ~ Scenario 4: MLFlow with remote tracking server, backend and artifact stores
- ~ Logging data to runs
- ~ Logging functions
- ~ Launching multiple runs in one program
- ~ Performance tracking with metrics
- ~ Visualizing metrics

#### => Automatic logging :

- ~ Scikit-Learn
- ~ Tensorflow and Keras
- ~ Glueviz
- ~ Xgboost

~ Pytorch

=> MLFlow tracker :

- ~ Organizing runs in experiments
- ~ Managing experiments and runs with the tracking service API
- ~ Tracking UI
- ~ Querying runs programmatically
- ~ MLFlow tracking servers
- ~ Storage
- ~ Networking
- ~ Logging to a tracking server

=> MLFlow projects :

- ~ Overview
- ~ Specifying projects
- ~ Running projects
- ~ Iterating quickly
- ~ Building multistep workflows

=> MLFlow models :

- ~ Storage format
- ~ Model signature and input example
- ~ Model API
- ~ Built-In model flavors
- ~ Model customization
- ~ Built-In deployment tools
- ~ Deployment to custom targets

=> Model registry :

- ~ Model registry workflows
- ~ UI workflow
- ~ Registering a model
- ~ Using the model registry
- ~ API workflow
- ~ Adding an MLFlow model to the model registry
- ~ Fetching an MLFlow model from the model registry
- ~ Serving an MLFlow model from model registry
- ~ Adding or updating an MLFlow model descriptions
- ~ Renaming an MLFlow model
- ~ Transitioning an MLFlow models stage
- ~ Listing and searching MLFlow models
- ~ Archiving an MLFlow model
- ~ Deleting mlflow models

=> Mlflow integration with project :

- ~ Mlflow integration with project

=> Kubeflow foundation :

- ~ What is Kubeflow?
- ~ Core Kubeflow components
- ~ How to set up Kubeflow on Kubernetes
- ~ How to develop basic ML models in Kubeflow notebooks
- ~ How to train and deploy models in kubeflow
- ~ How to use Kubeflow pipelines
- ~ How to use kfserving to deploy models
- ~ How to manage logs with Kubeflow metadata component
- ~ Katib hyper parameter tuning
- ~ Kubeflow pipelines to kfserving

=> AWS MLOps :

- ~ Amazon sagemaker
- ~ Amazon s3
- ~ AWS codebuild
- ~ AWS codecommit
- ~ Sagemaker training job
- ~ Sage maker endpoint
- ~ Amazon api gateway
- ~ Sagemake model monitoring
- ~ Cloudwatch synthetics
- ~ Cloudwatch alarm

=> Azure MLOps :

- ~ Create an Azure machine learning workspace
- ~ Setup a new project in Azure DevOps
- ~ Import existing YAML pipeline to Azure DevOps
- ~ Declare variables for CI/CD pipeline
- ~ Create training compute
- ~ Train ML model
- ~ Register model
- ~ Deploy model in AKS

=> GCP MLOps :

- ~ Creating Flask application using Python
- ~ Best practices building Flask app
- ~ Understanding docker files and dependencies
- ~ Creating container image
- ~ Walkthrough of different deployment options
- ~ Serverless deep dive
- ~ Deploying on GCP app engine
- ~ Deploying on serverless framework
- ~ Hosted Kubeflow pipelines

- ~ *Start hosted pipelines*
- ~ *Cluster permissions*
- ~ *Development environment*
- ~ *Launch AI platform notebook*
- ~ *CI/CD production environment*
- ~ *Set up continuous integration (CI)*
- ~ *Verify CD*

=> Digital ocean :

- ~ *Droplets*
- ~ *File transfers*
- ~ *Gitops*
- ~ *Jenkins*
- ~ *Creating jobs*
- ~ *Creating pipelines in jenkins*
- ~ *Docker images*
- ~ *Kubernetes flow*
- ~ *Creating clusters*
- ~ *Load testing*

# Big Data Foundations

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA MASTERS

Course link : <https://ineuron.ai/course/Big-Data-Foundations>

## Course Description :-

Data is an essential part of any organization. Every organization generates a massive amount of real-time or batch data. This is where Big data plays a vital role irrespective of domain and industry. This complete course is designed to fulfill such requirements so that we will be able to work with a humongous amount of data. You will be able to create your Big Data Engine in your organization by implementing various big data stacks used across the industry.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Big Data Engine Creation
- => Streaming and Batch Processing of Data
- => Various SQL Databases
- => Various NOSQL Databases
- => Real-Time Implementation
- => Spark
- => Hive
- => Talend
- => Informatica
- => Hadoop Distributions
- => Deployment
- => DataBricks Implementation

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Introduction to Big data and Distributed Systems :

- ~ What Is Big Data?
- ~ Who Is Using Big Data?
- ~ Why Is Data So Important?
- ~ Big Bank: Big Challenge
- ~ Why Learn Big Data Technologies?
- ~ Pre-Requisite Data Scale
- ~ Common Problems
- ~ 3 Vs Of Big Data
- ~ Defining Big Data
- ~ Sources Of Data Flood
- ~ Exploding Data Problem
- ~ Redefining The Challenges Of Big Data
- ~ Possible Solutions: Scaling Up Vs. Scaling Out
- ~ Challenges Of Scaling Out

=> Hadoop and MapReduce :

- ~ Solution For Data Explosion-Hadoop

- ~ Hadoop: Introduction
- ~ Hadoop In Layman's Term
- ~ Hadoop Ecosystem
- ~ Evolutionary Features Of Hadoop
- ~ Hadoop Timeline
- ~ HDFS: Introduction
- ~ Design Of HDFS
- ~ Why Hadoop Cluster?
- ~ HDFS Blocks
- ~ Components Of Hadoop 3
- ~ NameNode And Hadoop Cluster
- ~ Arrangement Of Racks
- ~ Arrangement Of Machines And Racks
- ~ Local FS And HDFS
- ~ NameNode And Hadoop Cluster
- ~ Checkpointing
- ~ Replica Placement
- ~ Benefits-Replica Placement And Rack Awareness
- ~ URL And URN
- ~ HDFS Commands
- ~ Problems With HDFS In Hadoop 1.X
- ~ HDFS Federation
- ~ Anatomy Of File Read From HDFS
- ~ Important Java Classes To Write Data To HDFS
- ~ Anatomy Of File Write To HDFS
- ~ Writing File To HDFS: Steps
- ~ InputSplit
- ~ InputSplit And Data Blocks Difference
- ~ Why Is The Block Size 128 MB?
- ~ RecordReader
- ~ InputFormat
- ~ Default Inputformat : TextInputFormat
- ~ OutputFormat
- ~ Using A Different OutputFormat
- ~ Important Points
- ~ Partitioner
- ~ Using Partitioner
- ~ Map Only Job
- ~ Flow Of Operations In MapReduce
- ~ Serialization In MapReduce
- ~ Schedulers In YARN
- ~ FIFO Scheduler
- ~ Capacity Scheduler
- ~ Fair Scheduler
- ~ Differences Between Hadoop 1.X And Hadoop 2.X and hadoop 3.X

=> Hive and Advance Hive :

- ~ Introduction
- ~ Hive DDL
- ~ Demo: Databases.Ddl
- ~ Demo: Tables.Ddl
- ~ Hive Views
- ~ Demo: Views.Ddl
- ~ Architecture
- ~ Primary Data Types
- ~ Data Load
- ~ Demo: ImportExport.Dml
- ~ Demo: HiveQueries.Dml
- ~ Demo: Explain.Hql Table Types
- ~ Demo: ExternalTable.Ddl
- ~ Complex Data Types
- ~ Demo: Working With Complex Datatypes
- ~ Hive Variables
- ~ Demo: Working With Hive Variables
- ~ Hive Variables And Execution Customisation
- ~ Working With Arrays
- ~ Sort By And Order By
- ~ Distribute By And Cluster By
- ~ Partitioning
- ~ Static And Dynamic Partitioning
- ~ Bucketing Vs Partitioning
- ~ Joins And Types
- ~ Bucket-Map Join
- ~ Sort-Merge-Bucket-Map Join
- ~ Left Semi Join
- ~ Demo: Join Optimisations
- ~ File Formats In Hive
- ~ Input Formats In Hive
- ~ Sequence Files In Hive
- ~ RC File In Hive
- ~ ORC Files In Hive
- ~ Inline Index In ORC Files
- ~ ORC File Configurations In Hive
- ~ SerDe In Hive
- ~ Demo: CSVSerDe
- ~ JSONSerDe
- ~ RegexSerDe
- ~ Analytic And Windowing In Hive
- ~ Demo: Analytics.Hql

- ~ Hcatalog In Hive
- ~ Demo: Using\_HCatalog
- ~ Accessing Hive With JDBC
- ~ Demo: HiveQueries.Java
- ~ HiveServer2 And Beeline
- ~ Demo: Beeline
- ~ UDF In Hive
- ~ Demo: ToUpper.Java And Working\_with\_UDF
- ~ Optimizations In Hive
- ~ Demo: Optimizations

=> Nosql database :

- ~ Challenges With Traditional RDBMS
- ~ Features Of NoSQL Databases
- ~ NoSQL Database Types
- ~ CAP Theorem

=> Hbase :

- ~ What Is HBase Regions
- ~ HBase HMaster ZooKeeper
- ~ HBase First Read
- ~ HBase Meta Table
- ~ Region Split
- ~ Apache HBase Architecture Benefits
- ~ HBase Vs. RDBMS
- ~ Shell Commands

=> Cassandra :

- ~ Cassandra Introduction
- ~ Cassandra Installation in local system
- ~ DATASTAX Cassandra setup
- ~ Cassandra Architecture Cassandra Queries

=> mongodb :

- ~ MondoDB Introduction
- ~ MondoDB Compass Setup
- ~ MongoDB Atlas Setup
- ~ MondoDB Architecture
- ~ MondoDB Queries

=> kafka :

- ~ Introduction To Kafka
- ~ Kafka Architecture
- ~ Kafka Key Concepts/Fundamentals
- ~ Overview Of Zookeeper And Its Role In Kafka Cluster
- ~ Cluster, Nodes, Brokers, Topics Consumer, Producers, Logs, Partitions Concept Of Consumer Groups
- ~ Leader & Follower Partition
- ~ Installing One Node Kafka Cluster On Local Installing Multinode Kafka Cluster On Local Command Line Producer And Consumer Replication Concept For Fault Tolerance How Data
- ~ Is Stored In Brokers
- ~ Log Segments, Message Offsets, Message Index
- ~ Isr List / Minimum Isr
- ~ Committed Vs Uncommitted Messages Writing A Kafka Producer In Java Writing A Kafka Consumer In Java Scaling Up The Kafka Cluster Achieving Exactly Once Semantics
- ~ Integrating Kafka With Spark Structured Streaming.

=> Airflow :

- ~ Introduction To Airflow And Its Usage What Is Workflow
- ~ Cron-Job Creation Example Airflow Additional Features
- ~ Airflow Architecture And Components Airflow Installation Demo
- ~ Dags-Creating A Simple Helloworld Dag Introduction To Tasks And Operators
- ~ Viewing The DAG In Ui-Graph View, Tree View, Logs Viewing
- ~ Example Showcasing Bash Operators Usage Setting Precedence Among Various Tasks Lifecycle OfATask-Understanding Various Stages About Trigger\_rules & Understanding With
- ~ Example Airflow Artifact - More On Operators
- ~ Writing Our Own Custom Operators Walkthrough Of Airflow UI
- ~ Connections To Various Datastores & Variables
- ~ Working With Connections, Understanding Sensors Demo
- ~ Building an end-to-end customer-360 pipeline using Airflow involving data collection from various sources, processing in spark, loading the processed data in hive and uploading the same to HBase and generating a notification about success of the pipeline to the downstream applications.

=> spark :

- ~ Introduction To Apache Spark
- ~ Map Reduce Limitations
- ~ RDD's
- ~ Spark Context - SQLContext And HiveContext
- ~ Programming With RDD's
- ~ Creating RDD's From Text-Files
- ~ Transformations And Actions
- ~ How Does Spark Execution Work
- ~ RDD API's - Filter
- ~ FlatMap
- ~ Fold
- ~ Foreach
- ~ Glom
- ~ GroupBy
- ~ Map
- ~ ReduceByKey
- ~ Zip
- ~ Persist
- ~ Unpersist

- ~ Read/Write From Storage
- ~ RDD Examples Internal Of Spark Workings
- ~ RDD API's - Aggregate
- ~ Cartesian
- ~ Checkpoint
- ~ Coalesce
- ~ Repartition
- ~ Cogroup
- ~ CollectAsMap
- ~ CombineByKey
- ~ Count And CountApprox Functions
- ~ More RDD Examples
- ~ Schema - StructType
- ~ StructFields
- ~ DataType
- ~ DataFrame API's And Examples
- ~ Create Temporary Tables
- ~ SparkSQL
- ~ Spark Dataset
- ~ Parquet Vs Avro
- ~ Examples And Problem Solving On Real Data Using RDD And Converting The Same To Dataframe
- ~ Create A Spark Project
- ~ SBT / Maven
- ~ How Do Maven Repo Work
- ~ Accumulators
- ~ Broadcast Variables
- ~ Query Execution Plan

=> big data on cloud :

- ~ AWS EMR (Elastic MapReduce):
- ~ What is a VM (Virtual Machine) On-Premise vs Cloud Setup
- ~ Major Vendors of Hadoop Distribution Why Cloud & Big Data on Cloud Major Cloud Providers of Bigdata What is EMR
- ~ Hdfs vs S3 What Is S3
- ~ Important Instances in AWS Kinds of Nodes in Cluster
- ~ Transient vs Long Running Cluster Running Spark Code on Emr
- ~ How to Track Your Job
- ~ Copy File From S3 to Local Zeppelin Notebook
- ~ Types of EC2 Instances How to Create a VM What is a Keypair Elastic IP
- ~ AWS Storage, Networking & CLI Instance Store
- ~ S3 & EBS
- ~ Public ip Vs Private Ip Network Switches Security Group
- ~ Aws Command Line Interface
- ~ Launch A Emr Cluster Using Advanced Options
- ~ AWS Athena
- ~ What is Athena?
- ~ When do we require Athena What problem Athena Solve How Athena Works
- ~ Athena Pricing
- ~ Athena Practical Demonstration
- ~ How to create a normal table manually on csv data residing in s3
- ~ How to minimize data scanning in Athena How to create partition table on Parquet file
- ~ Inferring Schema automatically using AWS Glue
- ~ AWS Glue
- ~ What is AWS Glue? Introduction To Glue Features of Glue AWS Glue Benefits
- ~ AWS Glue Terminology
- ~ Pointing to Specific Data Stores and Endpoints Glue Data Catalogue
- ~ Crawlers
- ~ Connecting to Your Data Store Using Crawlers for Catalogue Tables
- ~ Overview and Working of Glue Jobs Adding New Jobs in Glue
- ~ Triggering Jobs and Their Scheduling
- ~ AWS Redshift
- ~ Database vs Data Warehouse vs Data Lake Introduction to Amazon Redshift
- ~ Benefits of Amazon Redshift Use Cases of Amazon Redshift
- ~ Redshift Master Slave Architecture Types of Nodes
- ~ Redshift Spectrum Redshift Fault Tolerance Redshift Sort Keys
- ~ Redshift Distribution Styles Practical Demonstration

=> Enterprise Big Data ETL Tools :

- ~ Introduction to ETL from Talend Studio- Integration with HDFS, Hive, Sqoop, Spark etc
- ~ Introduction to ETL from Informatica BDM- Integration with HDFS, Hive, Sqoop, Spark etc

=> databricks :

- ~ Databricks Introduction
- ~ Databricks Setup
- ~ Databricks Integration with cloud
- ~ Databricks OPS Pipeline
- ~ Databricks in Production

=> python :

- ~ Python Core
- ~ Introduction of python and comparison with other
- ~ Programming language
- ~ Installation of Anaconda Distribution and other python
- ~ IDE Python Objects, Number & Booleans, Strings
- ~ Updating and deleting the data
- ~ Container objects, Mutability of objects
- ~ Operators Arithmetic, Bitwise, Comparison and Assignment operators, Operators Precedence and associativity
- ~ Conditions (If else, if elif else) Loops (While, for)
- ~ Break and Continue statement and Range Function.
- ~ String Objects And Collections
- ~ String object basics



- ~ String methods
- ~ Splitting and Joining Strings
- ~ String format functions
- ~ List object basics
- ~ List as stack and Queues
- ~ List comprehensions
- ~ Tuples, Set, Dictionaries Functions
- ~ Tuples, Sets Dictionary Object basics, Dictionary Object methods, Dictionary View Objects.
- ~ Functions basics, Parameter passing, Iterators Generator functions
- ~ Lambda functions
- ~ Map, Reduce, Filter functions
- ~ OOPS Concepts Working With Files
- ~ OOPS basic concepts
- ~ Creating classes and Objects Inheritance
- ~ Multiple Inheritance
- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other File methods
- ~ Exception Handling Database Programming
- ~ Creating, inserting and retrieving Table
- ~ Using Standard Module
- ~ Creating new modules
- ~ Exceptions Handling with Try except

=> sql :

- ~ Installing and configuring MySQL
- ~ Install and Configure MySQL Client
- ~ DDL - Create database/table, Drop, Alter, etc
- ~ DML - INSERT, DELETE, UPDATE, MERGE etc
- ~ DML - INSERT, DELETE, UPDATE, MERGE etc
- ~ DQL - SELECT, etc
- ~ JOINS - One Many, Many Many
- ~ DISTINCT
- ~ ORDER BY
- ~ LIMIT
- ~ WILD CARDS
- ~ LOGICAL OPERATORS - LIKE, EQUAL, AND, OR etc
- ~ STRING Functions
- ~ DATE Functions
- ~ MATH Functions
- ~ COUNT, MIN and MAX
- ~ SUM
- ~ AVG
- ~ LAG and LEAD function Examples
- ~ Top N Analysis
- ~ ROW\_NUMBER
- ~ RANK AND DENSE\_RANK
- ~ CASE WHEN
- ~ PIVOT
- ~ LISTAGG
- ~ UNION
- ~ Sub-Queries
- ~ EXISTS
- ~ NOT EXISTS
- ~ WITH CLAUSE
- ~ Recursive WITH & CTE
- ~ Regular Expressions in SQL

# JavaScript Coding Interview Preparation

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : WEB DEVELOPEMENT INTERVIEW

Course link : <https://ineuron.ai/course/JavaScript-Coding-Interview-Preparation>

## Course Description :-

This course is designed mostly for JavaScript test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> JavaScript Theoretical Test

=> JavaScript Practical Test

=> JavaScript Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> JavaScript Test :

- ~ JavaScript Test 1
- ~ JavaScript Test 2
- ~ JavaScript Test 3
- ~ JavaScript Test 4
- ~ JavaScript Test 5
- ~ JavaScript Test 6
- ~ JavaScript Test 7
- ~ JavaScript Test 8
- ~ JavaScript Test 9
- ~ JavaScript Test 10
- ~ JavaScript Test 11
- ~ JavaScript Test 12
- ~ JavaScript Test 13
- ~ JavaScript Test 14
- ~ JavaScript Test 15
- ~ JavaScript Test 16
- ~ JavaScript Test 17
- ~ JavaScript Test 18
- ~ JavaScript Test 19
- ~ JavaScript Test 20

# R Programming

---

Topic Name : PROGRAMMING

Sub-topic Name : R

Course link : <https://ineuron.ai/course/R-Programming>

## Course Description :-

R is a programming language used for statistical computing, data manipulation, data visualization and implementing Machine Learning algorithms. It is widely used by researchers from diverse disciplines to estimate and analyze the result. Moreover, R programming is a wonderful statistical tool extensively used in the industry for the task which requires exhaustive research to analyze and understand the data. In this course, you will be going to learn R programming thoroughly and implement some widely used Machine Learning algorithms.

## Course Features :-

- => Implementation of widely used Machine Learning algorithms.
- => Exhaustive learning from basics to advance
- => Quizzes
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Introduction to basics of R
- => R Matrices and Arrays
- => R Data Frames
- => Data Input and Output
- => Operators and Conditional Statements
- => Advanced R Programming
- => Data Preparation
- => Statistics Overview
- => Hypothesis Testing
- => Data Visualization
- => Machine Learning with R

## Requirements :-

- => No prior programming experience
- => A system with Internet Connection
- => Interest to learn
- => Your dedication

## Instructors :-

- => Shlok Pandey :
  - ~ Content Developer (Deep Learning)

## Curriculum details :-

- => Course Introduction :
  - ~ Overview of the course Preview
  - ~ Installation and Setup
  - ~ Guide to R Studio
- => Introduction to Basics of R :
  - ~ Introduction to R programming Preview
  - ~ Arithmetics in R
  - ~ Variables and Data Types
  - ~ Basics of Vector
  - ~ Vector Operations
  - ~ Vector Indexing and Slicing
  - ~ Comparison Operators
- => R Matrices and Arrays :
  - ~ Introduction to R Matrices and Arrays Preview
  - ~ Creating a Matrix
  - ~ Matrix Arithmetic
  - ~ Matrix Operations
  - ~ Matrix Selection and Indexing
  - ~ Factor and Categorical Matrices

- ~ *Creating Multidimensional Arrays*
- ~ *Indexing Multidimensional Arrays*

#### => R Data Frames :

- ~ *Introduction to R Data Frames*
- ~ *Data Frame Basics*
- ~ *Data Frame Indexing and Selection*
- ~ *Data Frame Operations*

#### => Lists, Strings and Regular Expressions :

- ~ *Introduction to lists*
- ~ *Introduction to strings and their creation*
- ~ *Printing strings*
- ~ *String concatenation*
- ~ *String Manipulation-1*
- ~ *String Manipulation-2*
- ~ *Regular Expressions-1*
- ~ *Regular Expressions-2*

#### => Data Input and Output :

- ~ *Introduction to data input and output*
- ~ *CSV files in R*
- ~ *Excel files in R*

#### => Operators and Conditional Statements :

- ~ *Introduction to Operators and Conditional Statements*
- ~ *Logical Operators*
- ~ *If, else, and if-else statements*
- ~ *While loops*
- ~ *For loops*
- ~ *Functions*

#### => Advanced R Programming :

- ~ *Introduction to Advanced R programming*
- ~ *Built-in R Features*
- ~ *Understanding apply function:*
  - a. *lapply*
  - b. *sapply*
  - c. *vapply*
- ~ *Math functions in R*
- ~ *Dates and Timestamps*

#### => Data Preparation :

- ~ *Introduction to Data Preparation*
- ~ *Guide to dplyr package*
- ~ *Pipe operator*
- ~ *Guide to use tidyr*
- ~ *Dealing with missing data*
- ~ *Replacing missing data*

#### => Statistics Overview :

- ~ *Introduction to Statistics*
- ~ *Mean, Median, and Mode*
- ~ *Variance, standard deviation, and coefficient of variability*
- ~ *Covariance and correlation*

#### => Hypothesis Testing :

- ~ *Introduction to Hypothesis testing*
- ~ *Standard errors and Confidence intervals*
- ~ *Hypothesis testing*
- ~ *Type-1 and type-2 error*
- ~ *P-value*

#### => Data Visualization :

- ~ *Introduction to ggplot*
- ~ *Histograms*
- ~ *Scatterplots*
- ~ *Barplots*
- ~ *Boxplots*
- ~ *Coordinates and Faceting*
- ~ *Themes*
- ~ *Plotly and interactive visualization*

#### => Machine Learning with R :

- ~ *Introduction to Machine Learning*
- ~ *Linear Regression*
- ~ *Logistic Regression*
- ~ *K-Nearest Neighbours*
- ~ *Decision Trees*
- ~ *Random Forests*
- ~ *Support Vector Machines*
- ~ *K-Means Clustering*

#### => Project: Customer Churn Classification :

- ~ *Using the customer churn dataset, we will classify whether the customer will purchase Internet services based on various parameters.*

# AWS Sagemaker

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/AWS-Sagemaker>

## Course Description :-

Well groomed knowledge of the complete AWS Machine Learning ecosystem is required and SageMaker is one of the Most Important component of it. This course includes real world Projects which enables you to learn and solidify your concept on SageMaker.

## Course Features :-

- => Self-Paced Classes
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Learn about various Algorithms like XgBoost ,Deep AR , Linear Learner , Factorization Machines on SageMaker
- => Learn To Deploy custom Machine Learning Algorithms on SageMaker
- => Learn To implement Real world Machine Learning Problem on SageMaker
- => Learn To do Hyper Parameter Tuning on SageMaker

## Requirements :-

- => Prior knowledge in AWS
- => Little bit of Machine Learning knowledge
- => An AWS account

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => AWS Sagemaker :
  - ~ Introduction Preview
- => General overview of sagemaker :
  - ~ Introduction to AWS Sagemaker Preview
  - ~ Instance Types
  - ~ Build in algorithm
  - ~ Ground Truth and NEO
  - ~ Different API Levels
- => Prerequisite of Sagemaker :
  - ~ Making S3 Bucket
  - ~ Spinning Jupyter Notebook in Sagemaker part 1
  - ~ Spinning Jupyter Notebook in Sagemaker part 2
- => Basic of implementing ML Model on Sagemaker :
  - ~ Sagemaker ML Model Overview
  - ~ Sagemaker NEO
  - ~ Sagemaker Security
- => Amazon SageMaker to build, train, and deploy a machine learning (ML) model :
  - ~ Demo part 1 Preview
  - ~ Demo part 2
  - ~ Demo part 3
- => Object Detection :
  - ~ Intro to object detection
  - ~ Downloading Data
  - ~ Model Building and Training

# Manual Testing Kickstart

---

Topic Name : TESTING

Sub-topic Name : MANUAL TESTING

Course link : <https://ineuron.ai/course/Manual-Testing-Kickstart>

## Course Description :-

The goal of this course is to learn the fundamental ideas and methods of Software Testing. Software testing Basics, SDLC Models, Waterfall, V, Spiral, and Agile model, STLC, Phases & Types of Testing, Black Box & White Box Testing, Smoke & Sanity Testing, Regression & Retesting will all be covered in the training. You will learn about, Test Planning, Test case identification & Creation, Test case execution, Bug Management, and Bug Tracking tools like Jira, and Test Reporting. MySQL for backend data validation will also be covered.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Basics of Testing
- => Principles Of Testing
- => SDLC
- => Waterfall
- => Spiral
- => Test Estimation & Test Management
- => TestCase Development
- => Testing Types
- => Testing Techniques

## Requirements :-

- => Knowledge of C++
- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Kiran Sahu :

~ QA Manager with 12+ years of professional experience, worked in Brands like Infosys, Delhivery, Mydala, Aurea, Jive, Crossover, Agama Solutions & OSTC, have experience of working in global platforms and with multinational professionals. Strong domain knowledge on Retail, Logistics, Banking, Trading, Ecommerce Applications. Experience in Training and Mentoring Candidates all across the globe on Software Testing, MySQL and Agile.

## Curriculum details :-

=> Introduction to Manual Testing :

- ~ Basics of Testing
- ~ Principles Of Testing

=> Manual Testing :

- ~ SDLC
- ~ STLC SDLCvsSTLC
- ~ Waterfall
- ~ SoftwareTesting V
- ~ Spiral
- ~ Agile
- ~ Test Estimation & Test Management
- ~ TestPlan
- ~ TestCase Development
- ~ RTM

=> Manual Testing Phases :

- ~ Phases of testingUnitIntegration
- ~ Phases of testingSystem Testing
- ~ Phases of TestingUAT

=> Testing Types :

- ~ Types Of Testing SmokeandSanity
- ~ Types Of Testing Regression vs Retesting
- ~ Types Of Testing Func NonFunc Testing
- ~ Types Of Testing WhiteBox BlackBox Testing

=> Testing Techniques :

- ~ Testing Techniques Equivalence Partitioning And BVA
- ~ Testing TechniquesDecisionTable

=> Manual Testing MySQL :

- ~ Mysql create insertselect
- ~ Mysql orderby
- ~ Mysql groupby
- ~ Mysql wildcards
- ~ Mysql nullalter drop rename
- ~ Mysql limitddldmldcl
- ~ Mysql keys
- ~ Mysql definingKeys
- ~ Mysql join
- ~ Mysql subquery

# Azure Fundamentals

---

Topic Name : CLOUD

Sub-topic Name : AZURE

Course link : <https://ineuron.ai/course/Azure-Fundamentals>

## Course Description :-

The use of cloud computing is growing rapidly across all industries, opening up a slew of new job prospects. A wide range of positions is available, from developers and architects to security specialists and data scientists. As a result of this course, you will be prepared to begin your Microsoft Azure certification journey confidently. For the AZ-900 certification exam and your cloud career, you'll need a solid foundation of core knowledge from our Microsoft Azure Fundamentals AZ-900 Exam Prep Specialization. This program's content purely matches the AZ-900 test objectives.

## Course Features :-

- => Real-Time implementation
- => ML/DL model testing and monitoring
- => Scenario-based questions
- => Challenges
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => You will understand cloud concepts
- => Learn about core Azure services
- => Master security, privacy
- => Compliance and trust
- => Understand Azure pricing and support
- => Students who wish to start learning how to use the Azure platform

## Requirements :-

- => Computer with i3 and above configuration
- => Azure free or paid account
- => Your dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Introduction :
  - ~ Introduction to cloud computing Preview
  - ~ Cloud models
  - ~ Different cloud providers
- => Regions and Availability Zones :
  - ~ Understanding regions and availability zones in Azure Preview
  - ~ Creating Microsoft Azure account
- => Azure Virtual Machines :
  - ~ Topics covered in this section
  - ~ Getting started with Azure virtual machines
  - ~ Creating your first virtual machine in azure
  - ~ Connecting to the Azure virtual machine and running commands
  - ~ Connecting SSH
  - ~ Understanding Azure VM-key concepts
  - ~ Installing Nginx on Azure virtual machine
  - ~ Simplifying installing software on the Azure virtual machine
  - ~ Increasing availability for azure VM
  - ~ Virtual machine scale sets
  - ~ Exploring scaling and load balancing
  - ~ Static IP, monitoring and reducing costs
  - ~ Designing a good solution with Azure VM
  - ~ Exploring Azure virtual machine scenarios
- => Compute Services :
  - ~ Topics covered in this section
  - ~ Getting started with Azure app services



- ~ Creating your first Azure web app using app service
- ~ Exploring Azure app service
- ~ Getting started with containers
- ~ Azure container instances
- ~ Getting started with serverless
- ~ Azure functions
- ~ Building workflows logic apps
- ~ Azure responsibility model
- ~ Exploring Azure cloud service categories scenarios
- ~ Review Azure compute services
- ~ Deleting resource group

#### => Storage Service :

- ~ Getting started with Azure storage
- ~ Managed and unmanaged block storage in Azure
- ~ Getting started with Azure file storage
- ~ Exploring Azure file storage

#### => Databases :

- ~ Getting started with databases
- ~ Snapshot and standby
- ~ Availability and durability
- ~ How to achieve availability and durability
- ~ Understanding database fundamentals RTO and RPO
- ~ Data consistency
- ~ Understanding how to choose a database
- ~ Relational databases
- ~ Exploring OLTP relational databases in azure
- ~ Creating mysql server in azure
- ~ Playing with mysql server in azure
- ~ Azure synapse analytics
- ~ Azure nosql database
- ~ Azure cosmos DB
- ~ Azure cache for redis
- ~ Review databases in azure

#### => Networking :

- ~ Understanding need for azure network
- ~ Understanding need for subnets
- ~ Playing with azure virtual network
- ~ Azure virtual network important things to remember
- ~ Exploring azure Ddos Services
- ~ Exploring Azure Firewall
- ~ Network security groups
- ~ Security Best practices
- ~ Deployment models review
- ~ VPN and ExpressRoute

#### => Resource Hierarchy :

- ~ Understanding Resource Hierarchy
- ~ Demo on Resource Hierarchy
- ~ Resource groups, subscription and management groups

#### => Azure Active Directory :

- ~ Active Directory part 1
- ~ Active Directory part 2

#### => Security and identity management :

- ~ Azure security center
- ~ Azure Sentinel
- ~ Azure Key Vault
- ~ Azure key vault demo part 1
- ~ Azure key vault demo part 2
- ~ Role based Access control (RBAC)
- ~ Role based Access control (RBAC) demo

#### => Compliance, privacy and governance :

- ~ Azure Resource Locks
- ~ Azure resource Tags
- ~ Azure policy
- ~ Azure Blueprint
- ~ Cloud adoption framework for azure
- ~ Core tenets of security, privacy, compliance

#### => Managing costs :

- ~ Cost affecting factors
- ~ Cost reduction methods, reservations
- ~ Azure cost management

#### => Azure support and guarantees :

- ~ Exploring azure support plans
- ~ Automate azure recommendations with advisor
- ~ Monitoring azure with azure monitor
- ~ Monitoring azure service status with service health
- ~ Exploring azure management services
- ~ SLA and composite SLA in Azure
- ~ Service lifecycle in azure

# Pro Max Interview Preparation Edition 1

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Max-Interview-Preparation-Edition-1>

## Course Description :-

Pro Max Edition 1. These are interview preparation tests with a singular goal, to make sure you get a little better in real-world interviews. Leaderboards are ranked based on 1st attempt.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => Interview Preparation Theoretical Test
- => Interview Preparation Practical Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

- => Interview Preparation Test :
  - ~ Interview Preparation Test 1
  - ~ Interview Preparation Test 2
  - ~ Interview Preparation Test 3
  - ~ Interview Preparation Test 4
  - ~ Interview Preparation Test 5
  - ~ Interview Preparation Test 6

# RPA - UiPath

---

Topic Name : RPA

Sub-topic Name : UIPATH

Course link : <https://ineuron.ai/course/RPA---UiPath>

## Course Description :-

Learn and master UiPath Studio and then build state-of-the-art software robots from scratch. UiPath has evolved to become the only RPA platform in the market to support the full automation lifecycle from discovery to measurement. Its product portfolio continues to stay at the forefront of innovation, continuously expanding its traditional RPA offering capabilities to include tools like process mining, embedded analytics, improved AI fabric components, SaaS-based RPA, and test automation. UiPath is considered one of the fastest RPA solutions in the industry as well often 3-4x faster than other RPA products.

## Course Features :-

- => Online Live Classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to Course
- => What is RPA
- => ControlFlows in UiPath
- => Variables and Arguments in UiPath
- => Module 1: Input Activities and Methods in UiPath
- => Module 2: Type of Workflows in UiPath
- => Module 3: Selectors and Recordings in UiPath
- => Module 4: Error Handling and Debugging in UiPath
- => Module 5: Unstructured and Data Scrapping methods in UiPath
- => Module 6: Excel, PDF and Email activities in UiPath
- => Module 7 : Synchronization activities in UiPath
- => Module 8 : Connecting with Git, TFS and SVN in UiPath
- => Module 9 : Orchestrators in UiPath
- => Module 10 : ReFramework in UiPath

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction :
  - ~ Introduction Preview
  - ~ Topics to be covered Preview
- => What is RPA? :
  - ~ What is RPA? Preview
  - ~ Products of UiPath
  - ~ UiPath Installation Preview
  - ~ UiPath Overview
  - ~ Market leaders
  - ~ Our Purpose
- => ControlFlows in UiPath :
  - ~ ControlFlow Part 1
  - ~ ControlFlow Part 2
  - ~ ControlFlow Part 3

=> Variables and Arguments in UiPath :

- ~ Variables
- ~ Arguments

=> Module 1: Input Activities and Methods in UiPath :

- ~ Input Activities
- ~ Scope and Storing value in the output
- ~ Input methods
- ~ Properties

=> Module 2: Type of Workflows in UiPath :

- ~ Project Layout Part 1
- ~ Project Layout Part 2

=> Module 3: Selectors and Recordings in UiPath :

- ~ Selector Part 1
- ~ Selector Part 2
- ~ Recordings

=> Module 4: Error Handling and Debugging in UiPath :

- ~ Errorhandling and Debugging

=> Module 5: Unstructured and Data Scrapping methods in UiPath :

- ~ Unstructured Data
- ~ Data Scraping

=> Module 6: Excel, PDF and Email activities in UiPath :

- ~ Excel Activities
- ~ Pdf Activities
- ~ Email Activities
- ~ What is Citrix
- ~ ForEachLoop In the datatable

=> Module 7 : Synchronization activities in UiPath :

- ~ Synchronization Activities Part 1
- ~ Synchronization Activities Part 2

=> Module 8 : Connecting with Git, TFS and SVN in UiPath :

- ~ External Codes

=> Module 9 : Orchestrators in UiPath :

- ~ Assets
- ~ Queues part 1
- ~ Queues part 2
- ~ Queues part 3
- ~ Publish and How to run through Orchestrator
- ~ Logs
- ~ UiPath Assistant Preview
- ~ Orchestrator final Overview

=> Module 10 : ReFramework in UiPath :

- ~ ReFramework Intro
- ~ ReFramework Part 1
- ~ ReFramework Part 2
- ~ ReFramework Part 3
- ~ ReFramework part 4

=> State Machine Demo :

- ~ State Machine Demo Introduction
- ~ State Machine Demo Part 1
- ~ State Machine Demo Part 2
- ~ State Machine Demo Part 3

=> Excel Calculation Demo :

- ~ Excel Calculation Demo Introduction
- ~ Excel Calculation Demo Part 1
- ~ Excel Calculation Demo Part 2
- ~ Excel Calculation Demo Part 3
- ~ Excel Calculation Demo Part 4

=> PDF Extraction Demo :

- ~ PDF Extraction Introduction
- ~ PDF Extraction Part 1
- ~ PDF Extraction Part 2 and 3
- ~ PDF Extraction Part 4
- ~ PDF Extraction Part 5

=> Reading Data from Outlook :

- ~ Reading Data From Outlook Introduction
- ~ Reading Data From Outlook Part 1
- ~ Reading Data From Outlook Part 2

=> RPA Challenge Website :

- ~ RPA Challenge Introduction
- ~ RPA Challenge Part 1
- ~ RPA Challenge Part 2
- ~ RPA Challenge Part 3
- ~ RPA Challenge Part 4

=> RPA Challenge Website using ReFramework :

- ~ ReFramework RPA Challenge Introduction
- ~ ReFramework RPA Challenge Part 1
- ~ ReFramework RPA Challenge Part 2

- ~ *ReFramework RPA Challenge Part 3*
- ~ *ReFramework RPA Challenge Part 4*

=> Performer Model using ReFramework :

- ~ *PerFormer Model Using ReFramework Introduction*
- ~ *PerFormer Model Using ReFramework Part 1*
- ~ *PerFormer Model Using ReFramework Part 2*
- ~ *PerFormer Model Using ReFramework Part 3*
- ~ *PerFormer Model Using ReFramework Part 4*
- ~ *PerFormer Model Using ReFramework Part 5*

=> Thank You :

- ~ *Thank You*

# Class 6th Math

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Class-6th-Math>

## Course Description :-

This course is useful for Grade 6 students. In this course, entire NCERT will be covered, various questions from NCERT, NCERT exemplar and previous year will also be discussed. Dedicated doubt clearing sessions will also be conducted for helping the students regularly throughout their learning journey.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Mensuration
- => Statistics
- => Algebra
- => Numbers
- => Geometry
- => Ratio Proportion

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jawala Prakash :
- ~

## Curriculum details :-

- => Knowing Our Numbers :
  - ~ LECTURE 1 Comparing Numbers Preview
  - ~ LECTURE 2 How Many Numbers Can You Make Preview
  - ~ LECTURE 3 Introducing Large Numbers Preview
  - ~ LECTURE 4 International System of Numeration
  - ~ LECTURE 5 Exercise 11
  - ~ LECTURE 6 Large Numbers in Practice, Exercise 12 Question 1 to 3
  - ~ LECTURE 7 Exercise 12 Question 4 to 9
  - ~ LECTURE 7.1 Exercise 12 Question 10 to 12
  - ~ LECTURE 8 Estimation
  - ~ LECTURE 9 EXERCISE 13 Question 1 to 3
  - ~ LECTURE 10 Bodmas
- => Whole Numbers
  - :
  - ~ Lecture 1 : NCERT Solutions Ex2.1 Question 1-4
  - ~ Lecture 4 : NCERT Solutions Ex2.2 Question 4
- => Whole Numbers :
  - ~ Lecture 2 : NCERT Solutions Ex2.1 Question 4-8
  - ~ Lecture 3 : NCERT Solutions Ex2.2 Question 1-3
  - ~ Lecture 5 : NCERT Solutions Ex2.2 Question 5-7
  - ~ Lecture 6 : NCERT Solutions Ex2.3
- => Playing with Numbers :
  - ~ Lecture 1 : Introduction
  - ~ Lecture 3 : NCERT Solutions Ex 3.1
- => Playing with Numbers :
  - ~ Lecture 2 : Introduction
- => Basic Geometrical Ideas :
  - ~ Lecture 1 : Introduction
  - ~ Lecture 2 : NCERT Solutions Ex 4.1
  - ~ Lecture 3 : NCERT Solutions Ex 4.2
  - ~ NaN
  - ~ Lecture 5 : NCERT Solutions Ex 4.4
  - ~ Lecture 6 : NCERT Solutions Ex 4.5

~ Lecture 7 : NCERT Solutions Ex 4.6

## => Understanding Elementary Shapes :

- ~ LECTURE 1 Introduction
- ~ LECTURE 2 Measuring Line Segment
- ~ LECTURE 3 Exercise 31 Question 1 to 7
- ~ LECTURE 4 Right Angles
- ~ LECTURE 5 Exercise 5.2 Question 1 to 3
- ~ LECTURE 6 Exercise 5.2 Question 4 to 7
- ~ LECTURE 7 Angles Acute Obtuse and Reflex
- ~ LECTURE 8 Exercise 5.4 Question 1 to 2
- ~ LECTURE 9 Measuring Angles Using Protractor
- ~ LECTURE 10 Exercise 5.4 Question 1 to 7
- ~ LECTURE 11 Exercise 5.4 Question 8 to 11
- ~ LECTURE 12 Perpendiculars and Perpendicular Bisectors
- ~ LECTURE 13 Exercise 5.5 Question 1 to 4
- ~ LECTURE 14 Classification of Triangles
- ~ LECTURE 14.1 Classification of Triangles
- ~ LECTURE 15 Exercise 5.6 Question 1 to 4
- ~ LECTURE 16 Quadrilaterals
- ~ LECTURE 17 Exercise 5.7 Question 1 to 3
- ~ LECTURE 18 Polygons Exercise 5.8 Question 1 to 4
- ~ LECTURE 19 Three Dimensional Shape
- ~ LECTURE 20 Exercise 5.9

## => INTEGERS :

- ~ Lecture 1 : Introduction, Why we need Integers
- ~ Lecture 2 : Representation of Integers on Number Line
- ~ Lecture 3 : EX 6.1 Q1 to 10
- ~ Lecture 4 : Operations on Integers, Addition of Integers on Number Line, Rules for Addition of Integers, Additive Inverse
- ~ Lecture 5 : EX 6.2, Q 1 to 5
- ~ Lecture 6 : subtraction of Integers with the help of Number Line, EX 6.3 Q 1 to 4

## => FRACTIONS :

- ~ lecture 1 : Understanding the Concept of fraction, EX 7.1 Q1 to 2
- ~ lecture 2 : EX 7.1 Q 3 to 11
- ~ lecture 3 : Proper fraction, Improper and Mixed fraction, Converting Improper fraction into Mixed Fraction, Converting Mixed Fraction into improper fraction
- ~ lecture 4 : Representing Fractions on Number Line, Representing Mixed Fraction On Number Line
- ~ lecture 5 : EX 7.2 Q 1 to 3
- ~ lecture 6 : Equivalent Fraction, Simplest form of a Fraction
- ~ lecture 7 : EX 7.3 Q 1 to 9
- ~ lecture 8 : Like fractions, Unlike fractions, Comparing Like fractions, Comparing Unlike Fractions, General Method of Comparing two fractions (Method of Cross Multiplication and Method of Converting Given Fraction into Like fraction)
- ~ lecture 9 : Method of Converting given fractions into Like Fraction, EX 7.4 Q 1
- ~ lecture 10 : EX 7.4 Q 2 to 7
- ~ lecture 11 : EX 7.4 Q 8 to 10
- ~ lecture 12 : Addition and Subtraction of Fraction, Subtraction of Unlike Fraction
- ~ Lecture 13 : How do we Add and Subtract Mixed Fraction, EX 7.5 Q 1 to 5
- ~ Lecture 14 : EX 7.5 Q 1 to 4
- ~ Lecture 15 : EX 7.6 Q 1 to 4
- ~ Lecture 16 : EX 7.6 Q 5 to 9

## => Decimals :

- ~ Lecture 1 : NCERT Solutions Ex 8.1 Question 1-4
- ~ Lecture 2 : NCERT Solutions Ex 8.1 Question 5-10
- ~ Lecture 3 : NCERT Solutions Ex 8.2 Question 1&2
- ~ Lecture 4 : NCERT Solutions Ex 8.2 Question 3-6
- ~ Lecture 5 : NCERT Solutions Ex 8.2 Question 7
- ~ Lecture 6 : NCERT Solutions Ex 8.3
- ~ Lecture 7 : NCERT Solutions Ex 8.4
- ~ Lecture 8 : NCERT Solutions Ex 8.5
- ~ Lecture 9 : NCERT Solutions Ex 8.6 Question 1&2
- ~ Lecture 10 : NCERT Solutions Ex 8.6 Question 3-7

## => Data Handling :

- ~ Lecture 1 : NCERT Solutions Ex 9.1 Question 1&2
- ~ Lecture 2 : NCERT Solutions Ex 9.1 Question 3-6
- ~ Lecture 3 : NCERT Solutions Ex 9.1 Question 6&7
- ~ Lecture 4 : NCERT Solutions Ex 9.2 Question 1
- ~ Lecture 5 : NCERT Solutions Ex 9.3
- ~ Lecture 6 : NCERT Solutions Ex 9.4 Question 1-2
- ~ Lecture 7 : NCERT Solutions Ex 9.4 Question 3-4

## => Mensuration :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Perimeter of closed Figures
- ~ Lecture 3 : Perimeter of Standard Shapes
- ~ Lecture 4 : Ex 10.1 Q 1 to 7
- ~ Lecture 5 : EX 10.1 Q 8 to 17
- ~ Lecture 6 : Area
- ~ Lecture 7 : Estimating Area of any geometric figure on Graph sheet
- ~ Lecture 8 : Ex 10.3 Q 1 to 7
- ~ Lecture 9 : Ex 10.3 Q 8 to 12

## => Algebra :

- ~ Lecture1\_Introduction\_&\_Course\_Contents
- ~ Lecture2\_Matchstick\_Pattern\_&\_Variable
- ~ Lecture3\_More\_Concept\_On\_Matchstick\_Patterns\_&\_Variables
- ~ Lecture4\_NCERT\_EX11.1\_PROBLEM\_DISCUSSION
- ~ Lecture5\_Use\_Of\_Variables\_In\_Common\_Rules

- ~ Lecture6\_NCERT\_EX\_11.2\_PROBLEM\_DISCUSSION
- ~ Lecture7\_Forming\_Expressions\_Using\_Variables
- ~ Lecture8\_NCERT\_EX11.3\_PROBLEM\_DISCUSSION
- ~ Lecture9\_Using\_Expressions\_Practically
- ~ Lecture10\_NCERT\_EX11.4\_PROBLEM\_DISCUSSION
- ~ Lecture11\_What\_Is\_An\_Equation
- ~ Lecture12\_Solving\_An\_Equation
- ~ Lecture13\_NCERT\_EX11.5\_PROBLEMS\_DISCUSSION

=> Ratio & Proportion :

- ~ Lecture 1 : Introduction, Comparing Quantities
- ~ Lecture 2 : EX 12.1 Q 1 to 5
- ~ Lecture 3 : EX Q 6 to 10
- ~ Lecture 4 : EX Q 11 to 16
- ~ Lecture 5 : Understanding Proportion
- ~ Lecture 6 : Unitary Method
- ~ Lecture 7 : EX 12.2 Q 1 to 4
- ~ Lecture 8 : EX 12.3 Q 1 to 5
- ~ Lecture 9 : EX 12.3 Q 6 to 11

=> Symmetry :

- ~ Lecture 1 NCERT Solutions Ex 13.1 Question 1-3
- ~ NaN
- ~ Lecture 3 NCERT Solutions Ex13.1 Question 5&6
- ~ Lecture 4 NCERT Solutions Ex 13.2 Question 1
- ~ Lecture 5 NCERT Solutions Ex 13.2 Question 2-4

=> Practical Geometry :

- ~ LECTURE 1 Tools Used and Construction of Circles
- ~ LECTURE 2 EXERCISE 141 Question 1 to 5
- ~ LECTURE 3 Exercise 142 Question 1 to 5
- ~ LECTURE 4 Constructing Copy of a Given Line Exercise 143 Question 1 to 2
- ~ LECTURE 5 Perpendiculars
- ~ LECTURE 6 Exercise 144 Question 1 to 3
- ~ LECTURE 8 Angles ,Constructing Copy of Angles, Angle Bisector, constructing some Special Angles
- ~ LECTURE 9 Exercise 146 Question 1 to 9



# Mastering DSA with Java

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH JAVA

Course link : <https://ineuron.ai/course/Mastering-DSA-with-Java>

## Course Description :-

This course has been designed to help you become a complete and professional Java developer at the conclusion of the course, rather than only teaching essential Java skills.. After completing this course, you will have a thorough understanding of various Data Structures and Algorithms in Java which will further enhance your career as a java developer.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Problem Solving
- => Time-based DSA
- => Big O notation
- => Time and space complexity
- => Recursion
- => Power program theory
- => Combination theory
- => Stacks,queues,linked lists,trees
- => Searching, sorting, hashing

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to DSA :
  - ~ Why we need Data structures and algorithms
  - ~ Time based approach
  - ~ Concept of Big O and graphs
  - ~ Data Structures and Algorithms HB
- => Problem Solving :
  - ~ Start with a challenge - reverse string
  - ~ Reverse a string - solution
  - ~ Interview approach to solve a problem
  - ~ Classic interview steps for DSA problems
- => Data Structure Introduction :
  - ~ Memory process - Stack and Heap
  - ~ Physical and logical data structures
  - ~ Abstract Data Types - ADT
- => Recursion in depth :
  - ~ Introduction to recursion
  - ~ Tracing the recursion tree
  - ~ Trace tree assignment
  - ~ Trace tree solution
  - ~ Types of Recursion

- ~ Complex recursion tree
- ~ What is Factorial
- ~ DSA08 Factorial program in JAVA
- ~ Fibonacci series THEORY
- ~ Fibonacci series and its version JAVA Code
- ~ What is Power Program
- ~ Power Program JAVA code
- ~ What is a Combination Program
- ~ Combination Program JAVA code
- ~ Classic Tower of Hanoi problem
- ~ Classic Tower of Hanoi JAVA code

=> Linked List in depth :

- ~ Introduction to Linked List
- ~ Add value in linked list - cases
- ~ Push Append and insertat in LinkedList - JAVA code
- ~ Deletion of linked list THEORY.
- ~ Deletion in linked list JAVA code
- ~ Delete complete linked list JAVA code
- ~ Count all nodes in linkedlist JAVA code
- ~ Reversing a linked list THEORY
- ~ Reversing a linked list JAVA code

=> Circular Linked List in Depth :

- ~ Circular linked list THEORY
- ~ Circular Linked List push JAVA code
- ~ Traverse a circular linked list JAVA code
- ~ Deletion in circular linked list JAVA code
- ~ count nodes in circular linked list JAVA code
- ~ convert linked list to circular linked list JAVA code

=> Doubly Linked List in Depth :

- ~ Theory for doubly linked list
- ~ Doubly linked list push JAVA code
- ~ Insert After in doubly linked list JAVA code
- ~ add to last in doubly linked list JAVA code
- ~ Traverse a doubly linked list JAVA code
- ~ Deleting a node in doubly linked list JAVA code

=> Stack and Queue :

- ~ Stack - Push and Pop operation THEORY
- ~ Stack operations with JAVA code
- ~ Queue concept THEORY
- ~ Queue implementation in JAVA code
- ~ Circular queue THEORY
- ~ Circular queue JAVA code

=> Binary Search Tree :

- ~ What is Binary Search tree and creation THEORY update
- ~ Insertion and Deletion in BST THEORY
- ~ InOrder Traversal of BST THEORY
- ~ Pre Order traversal in BST THEORY
- ~ Post order traversal in BST THEORY
- ~ Creating a Binary Search tree JAVA code
- ~ search a key in BST JAVA code
- ~ Insertion in BST JAVA code
- ~ deletion of key in BST JAVA code
- ~ inorder preorder and postorder traversal in BST JAVA code

=> Hashing :

- ~ What is Hashing THEORY
- ~ Hash chaining with linked list
- ~ Linear Hash Shifting
- ~ Square hash shifting

=> AVL Tree :

- ~ What is AVL tree and height
- ~ Finding balance factor
- ~ Left Left and Right Right Rotation in AVL Tree
- ~ LR and RL rotation with 1 trick
- ~ Creating a AVL tree - Important
- ~ Deletion in AVL Tree.

=> HEAP :

- ~ Heap - Max and min Heap
- ~ Insertion and deletion in HEAP

=> Sorting algorithms :

- ~ Categories of sorts
- ~ Selection sort - Theory
- ~ Selection sort - Java Code
- ~ Bubble Sort - Theory
- ~ Bubble Sort - Java Code
- ~ Insertion sort - Theory
- ~ Insertion sort - Java Code
- ~ Quick Sort - Theory
- ~ Quick Sort - Theory part 2
- ~ Counting Sort - Theory
- ~ Merge Sort Theory
- ~ Merge sort JAVA code
- ~ Counting Sort - Java Code

# Snowflake Advanced

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Snowflake-Advanced>

## Course Description :-

This course will help you to learn the advanced concepts of Snowflake.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Loading Data
- => Copy Options
- => Loading unstructured data
- => Performance optimization
- => Loading from AWS
- => Loading from Azure
- => Loading from GCP
- => Snowpipe
- => Time Travel
- => Fail Safe

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Loading Data :
  - ~ Loading Methods
  - ~ Understanding Stages
  - ~ Creating stage
  - ~ Copy command
  - ~ Transforming Data
  - ~ More Transformations
  - ~ Copy options and ON\_ERROR
  - ~ File format object
  - ~ Summary
- => Copy Options :
  - ~ Understanding copy options
  - ~ Validation\_mode
  - ~ Working with rejected records
  - ~ size\_limit
  - ~ return\_failed\_only
  - ~ Truncatecolumns
  - ~ Force
  - ~ Load history
- => Loading unstructured data :
  - ~ High level steps
  - ~ our data
  - ~ Creating stage and raw file
  - ~ Parsing JSON
  - ~ Handling nested data

- ~ Dealing with hierarchy
- ~ Insert final data
- ~ Querying Parquet data
- ~ Loading Parquet data

=> Performance optimization :

- ~ Performance in Snowflake
- ~ Create dedicated virtual warehouse
- ~ Implement dedicated virtual warehouse
- ~ Scaling up
- ~ Scaling out
- ~ Caching
- ~ Maximize Caching
- ~ Clustering
- ~ Clustering Demo

=> Loading from AWS :

- ~ Sign up for free trial
- ~ Creating S3 bucket
- ~ Upload files in S3
- ~ Creating policy
- ~ Creating integration object
- ~ Loading from S3 part 1
- ~ Loading from S3 part 2
- ~ Loading from S3 part 3

=> Loading from Azure :

- ~ Sign up for free trial
- ~ Create a storage account
- ~ Create a Container
- ~ Create integration object
- ~ Create stage object
- ~ Load CSV file
- ~ Load JSON file

=> Loading from GCP :

- ~ Create a bucket
- ~ Create integration object
- ~ Create stage
- ~ Query and load data

=> Snowpipe :

- ~ What is Snowpipe
- ~ High level steps
- ~ Creating Stage
- ~ Creating pipe
- ~ Configure pipe and notifications
- ~ Error handling
- ~ Manage pipes

=> Time Travel :

- ~ Using time travel
- ~ Restoring data
- ~ UNDROP tables
- ~ Retention time
- ~ Time travel cost

=> Fail Safe :

- ~ Understanding Fail Safe
- ~ Fail Safe Storage

# AIOPS Crash Course

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : <https://ineuron.ai/course/AIOPS-Crash-Course>

## Course Description :-

Artificial Intelligence Operations (AIOps) is the most in demand technical skill these days. It helps to incorporate DevOps principle in AI product development. It's a live instructor-led certification program provided by iNeuron intelligence private limited.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable resources

## What you will learn :-

- => AIOps
- => Linux foundation
- => GIT foundation
- => GitHub

## Requirements :-

- => Minimum System requirement: Intel Core i3 processor and 4GB RAM or higher
- => Decent internet connection
- => Your Dedication

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

- => AI Ops Theory :
  - ~ Introduction to AI/MLOps Preview
  - ~ MLOps level1 workflow
  - ~ MLOps level2 workflow

# MLDL

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/MLDL>

## Course Description :-

This is Machine Learning masters and Deep Learning, where you will learn various things from beginning like python , API , deployment in Aws , azure , GCP , Heroku , database , various modules in statistics ,all machine learning algorithm , four mode of Chabot live Dialog flow , Amazon Lex , Azure Luis and RASA NLU , and 15+ live project all together in live instructor led class along with various mode of support and services and doubt clearing session.

## Course Features :-

- => Machine Learning in depth from beginning to advance discussion and implementation with Deployment.
- => Deep learning in-depth topic wise discussion and implementation with the project.
- => Docker and Kubernetes end to end with CI/CD pipeline for machine learning.
- => End to End Model Deployment in Azure, GCP, AWS, and Pivotal Cloud.
- => Python spark implementation with the project.
- => Time Series end to end implementation in machine learning and deep learning.
- => 26 + hands-on industry real-time projects.
- => Power BI and Tableau self-placed course.
- => Machine Learning Deep Learning Masters Certificate
- => 200 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access.
- => Doubt clearing one to one
- => Doubt clearing through mail and support team
- => Assignment in all the module
- => 20+ use case of Machine learning
- => A live project with real-time implementation
- => Resume building
- => career guidance
- => interview Preparation
- => Regular assessment
- => Job alerts
- => Online Instructor-led learning: Live teaching by instructors
- => Product Demo

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Data analytics
- => Mock interview
- => Interview preparation
- => Resume building

## Requirements :-

- => Dedication
- => Laptop with internet connectivity

## Instructors :-

- => krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course Introduction :

- ~ Introduction of Data science and its application in Day to Day life
- ~ Course overview and Dashboard description

=> Python Core :

- ~ Introduction of python and comparison with other
- ~ Programming language
- ~ Installation of Anaconda Distribution and other python
- ~ IDE Python Objects, Number & Booleans, Strings
- ~ Container objects, Mutability of objects
- ~ Operators Arithmetic, Bitwise, Comparison and Assignment operators, Operators Precedence and associativity
- ~ Conditions(if else, if elif else) Loops(While ,for)
- ~ Break and Continue statement and Range Function.

=> String Objects and collections :

- ~ String object basics
- ~ String methods
- ~ Splitting and Joining Strings
- ~ String format functions
- ~ List object basics
- ~ List as stack and Queues
- ~ List comprehensions

=> Tuples, Set, Dictionaries & Functions :

- ~ Tuples, Sets Dictionary Object basics, Dictionary Object methods, Dictionary View Objects.
- ~ Functions basics, Parameter passing, Iterators Generator functions
- ~ Lambda functions
- ~ Map , Reduce, Filter functions

=> OOPS Concepts & Working With Files :

- ~ OOPS basic concepts
- ~ Creating classes and Objects Inheritance
- ~ Multiple Inheritance
- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other File methods

=> Exception Handling :

- ~ Exceptions Handling with Try except

=> API :

- ~ Flask introduction
- ~ Flask Application
- ~ Open linkFlask
- ~ App RoutingFlask
- ~ URL BuildingFlask
- ~ HTTP MethodsFlask

=> Database :

- ~ Mongo DB SQL
- ~ Lite python SQL

=> Python pandas Modules :

- ~ Python Pandas Series
- ~ Python Pandas DataFrame
- ~ Python Pandas Panel
- ~ Python Pandas Basic functionality

=> Python Numpy :

- ~ NumPy Narray Object
- ~ NumPy Data Types
- ~ NumPy Array Attributes
- ~ NumPy Array Creation Routines
- ~ NumPy Array from Existing
- ~ Data Array From Numerical Ranges
- ~ NumPy Indexing & Slicing
- ~ NumPy Advanced Indexing
- ~ NumPy Broadcasting
- ~ NumPy Iterating Over Array
- ~ NumPy Array Manipulation
- ~ NumPy Binary Operators
- ~ NumPy String Functions
- ~ NumPy Mathematical Functions
- ~ NumPy Arithmetic Operations
- ~ NumPy Statistical Functions
- ~ Sort , Search & Counting Functions
- ~ NumPy Byte Swapping
- ~ NumPy Copies Views
- ~ NumPy Matrix Library
- ~ NumPy Linear Algebra

=> Exploratory Data Analysis :

- ~ Feature Engineering and Selection
- ~ Building Tuning and Deploying Models

- ~ *Analyzing Bike Sharing Trends*
- ~ *Analyzing Movie Reviews Sentiment*
- ~ *Customer Segmentation and Effective Cross Selling*
- ~ *Analyzing Wine Types and Quality*
- ~ *Analyzing Music Trends and Recommendations*
- ~ *Forecasting Stock and Commodity Prices*

=> Statistics :

- ~ *Descriptive Statistics*
- ~ *Sample vs Population statistics Random Variables*
- ~ *Probability distribution function Expected value*
- ~ *Binomial Distribution*
- ~ *Normal Distribution z score*
- ~ *Central limit Theorem*
- ~ *Hypothesis testing Z Stats vs T stats*
- ~ *Type 1 type 2 error*
- ~ *Confidence interval*
- ~ *Chi Square test*
- ~ *ANOVA test*
- ~ *F stats*

=> Machine Learning 1 :

- ~ *Introduction*
- ~ *Supervised , Unsupervised, Semi supervised, Reinforcement Train , Test, Validation Split*
- ~ *Performance Overfitting , underfitting OLS.*
- ~ *Linear Regression assumption.*
- ~ *R square adjusted*
- ~ *R square Intro to Scikit learn*
- ~ *Training methodology*
- ~ *Hands on linear regression*
- ~ *Ridge Regression*
- ~ *Logistics regression*
- ~ *Precision Recall ROC curve*
- ~ *F Score*

=> Machine Learning 2 :

- ~ *Decision Tree Cross*
- ~ *Validation Bias vs Variance*
- ~ *Ensemble approach Bagging*
- ~ *Boosting Random*
- ~ *Forest Variable Importance*

=> Machine Learning 3 :

- ~ *XGBoost*
- ~ *Hands on XgBoost*
- ~ *K Nearest Neighbour*
- ~ *Lazy learners*
- ~ *Curse of Dimensionality*
- ~ *K NN Issues*
- ~ *Hierarchical clustering K Means*
- ~ *Performance measurement*
- ~ *Principal Component analysis*
- ~ *Dimensionality reduction*
- ~ *Factor Analysis*

=> Machine Learning 4 :

- ~ *SVR*
- ~ *S V M*
- ~ *Polynomial Regression*
- ~ *Ada boost*
- ~ *Gradient boost*
- ~ *Gaussian mixture*
- ~ *Anomaly detection*
- ~ *Novelty detection algorithm Stacking*
- ~ *K NN regressor*
- ~ *Decision tree regressor DBSCAN*

=> Natural Language Processing :

- ~ *Text Analytics*
- ~ *Tokenizing , Chunking*
- ~ *Document term*
- ~ *Matrix TFIDF*
- ~ *Sentiment analysis hands on*

=> Spark :

- ~ *Spark overview.*
- ~ *Spark installation.*
- ~ *Spark RDD.*
- ~ *Spark dataframe .*
- ~ *Spark Architecture.*
- ~ *Spark ML lib.*
- ~ *Spark Nlp*
- ~ *Spark linear regression.*
- ~ *Spark logistic regression.*
- ~ *Spark Decision Tree.*
- ~ *Spark Naive Bayes*
- ~ *Spark xg boost*
- ~ *Spark time series.*
- ~ *Spark Deployment in local server*
- ~ *Spark job automation with scheduler.*



#### => Deep Learning :

- ~ Deep Learning Introduction.
- ~ Neural Network Architecture.
- ~ Loss Function.
- ~ Cost Function.
- ~ Optimizers.
- ~ CNN architecture.
- ~ Build First Classifier in CNN.
- ~ Deploy Classifier over cloud.
- ~ RNN overview.
- ~ GRU.
- ~ LSTM.
- ~ Time Series using RNN LSTM.
- ~ Customer Feedback analysis using RNN LSTM.

#### => Time series :

- ~ Arima
- ~ Sarima .
- ~ Auto Arima
- ~ Time series using RNN LSTM .
- ~ Prediction of NIFTY stock price.

#### => Deployment :

- ~ Deployment of all the project In cloudfoundry , AWS AZURE and Google cloud platform
- ~ Expose api to web browser and mobile application retraining a pproach of Machine learning model
- ~ Devops infrastructure for machine learning model
- ~ Data base integration and scheduling of machine learning model and retraining c ustom machine learning training approach.
- ~ AUTO ML
- ~ Discussion on infra cost and data volume
- ~ P rediction based on streaming data

#### => Extra session :

- ~ Discussion on project explanation in interview
- ~ Data scientist roles and responsibilities
- ~ Data scientist day to day work
- ~ Companies which hire a data scientist
- ~ Resume discussion with our team one to one

#### => Tableau and PowerBI self Paced Session :

- ~ Business Intelligence (BI) Concepts.
- ~ Microsoft Power BI (MSPBI) introduction.
- ~ Connecting Power BI with Different Data sources.
- ~ Power Query for Data Transformation.
- ~ Data Modelling in Power BI.
- ~ Reports in Power BI Reports and Visualisation types in Power BI.
- ~ Dashboards in Power BI.
- ~ Data Refresh in Power BI.
- ~ Traditional Visualisation(Excel) vs Tableau.
- ~ About Tableau.
- ~ Tableau vs Other BI Tool Pricing.

#### => Tableau Interview Question

### Project details :-

#### => Python projects :

- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis
- ~ Integration with web portal
- ~ Integration with rest API with Web portal and Mongo DB on Azure
- ~ Deployment on web portal on Azure
- ~ Text mining
- ~ Social media data churn

#### => Chatbot Projects :

- ~ Chatbot using Microsoft Luis
- ~ Chatbot using google Dialog flow
- ~ Chatbot using Amazon Lex
- ~ Chatbot using Rasa NLU
- ~ Deployemnt of chatbot with web , Telegram , Whatsapp , Skype

#### => Machine Learning Projects :

- ~ Healthcare analytics prediction of medicines based on FIT BITband
- ~ Revenue forecasting for startups
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ Anamoly detection in inventory packaged material.
- ~ Fault detection in wafferes based on sensordata
- ~ Demand forecasting for FMCG product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with Zomato dataset.
- ~ Fault detection in wafferes based on sensor data.
- ~ Cement\_Strength \_ reg.
- ~ Credit Card Fraud.
- ~ Forest\_Cover\_Classification .
- ~ Fraud Detection.
- ~ Income Prediction.
- ~ Mushroom classifier., Phising Classifier , Thyroid\_Detection .
- ~ Visibility climate.

#### => Deep Learning projects :

- ~ Customer Feedback analysis using RNN LSTM.

- ~ *Family member detection.*
- ~ *Industry financial growth prediction.*
- ~ *Speech recognition based attendance system.*
- ~ *Vehicle Number plate detection and recognition system.*

=> Tableau and PowerBI Projects :

- ~ *Project 1. Project Sales.*
- ~ *Project 2. Financial Report.*
- ~ *Project 3. HealthCare.*
- ~ *Project 4. Procurement Spend Analysis.*
- ~ *Project 5. Human Resource Tableau*

# SPARK Augmented Reality Live Class

---

Topic Name : AR VR

Sub-topic Name : SPARK AR

Course link : <https://ineuron.ai/course/SPARK-Augmented-Reality-Live-Class>

## Course Description :-

If you're interested in learning about Augmented Reality, you've come to the perfect spot. Learn the concepts and programming skills needed to create fully functional Augmented Reality apps for Android and iOS. You will need an Android or iPhone to complete this course.

## Course Features :-

- => Online Live Classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Mac or Windows laptop
- => Understanding the fundamentals of Unity and C# programming will be quite beneficial.

## Requirements :-

- => A System with Windows or Mac
- => An iPhone or Android Smartphone
- => Basic Knowledge of Programming is required

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

=> Introduction :

- ~ Introduction to Augmented and Virtual Reality
- ~ Introducing the Platform(Navigation and Keyboard shortcuts)

=> Templates :

- ~ Briefing all the templates available
- ~ Asset library
- ~ Object Manipulation

=> Simple Mask :

- ~ Face tracker
- ~ Face meshes
- ~ Alpha issues
- ~ Layers
- ~ Lut filter

=> 3D object :

- ~ Animation and Transition
- ~ Dynamic Text

=> 2D object :

- ~ Working with canvases and rectangles

=> Segmentation :

- ~ Body segmentation
- ~ Hair segmentation

=> Tracking :

- ~ Plane tracking
- ~ Hand tracking
- ~ Body tracking
- ~ Target tracking

=> Option Picker :

- ~ *Ui Picker*
- ~ *Slider patch*

=> Render passes :

- ~ *Scene, shader and face render pass*

=> Makeup :

- ~ *Retouching*
- ~ *Deformation*
- ~ *Eye color, Lashes*
- ~ *Blush, Lip color*

=> World effect :

- ~ *Particle systems*

=> Interactions :

- ~ *Patch editor*
- ~ *Face interactions*
- ~ *Screen interactions*

=> Shaders :

- ~ *Basics of creating shaders*

=> Scripting :

- ~ *Basics*

=> Miscellaneous :

- ~ *Audio and Music*
- ~ *Lights*
- ~ *SDF textures*
- ~ *Multipeer effect*
- ~ *Filter games*

=> Publish and Export :

- ~ *Optimisation*

=> Misc Projects :

- ~ *NaN*

# Be A DevOps Pro

---

Topic Name : DEVOPS

Sub-topic Name : DEVOPS MASTERS

Course link : <https://ineuron.ai/course/Be-A-DevOps-Pro>

## Course Description :-

DevOps, which is a mix of cultural principles, practices, and technologies such as Linux foundations, Docker, Kubernetes, Ansible, Terraform, ArgoCD, AWS Cloud, Git, Git, Prometheus, and others, increases an organization's capacity to build applications and services at high velocity. Become a DevOps master and increase the velocity of production now.

## Course Features :-

- => Online Instructor-led learning
- => Doubt Clearing
- => Proper Roadmap for DevOps
- => Lifetime Dashboard access
- => Recording of Live Class
- => Material
- => Interview Questions
- => Resume Building
- => Career Guidance
- => Quiz in every module - Based on Real Certification Questions Based
- => Certificate
- => Industry Level Projects and Case studies
- => Major Projects
- => Weekly Assignments
- => AWS Cloud Practitioner certification prep
- => AWS Cloud Architect certification Prep
- => Exam Questions Preparation
- => Mock Examinations

## What you will learn :-

- => Devops
- => Linux
- => Git
- => AWS
- => Docker
- => Ansible
- => Kubernetes
- => Terraform
- => CI CD Pipelines
- => Argo CD
- => Prometheus
- => Grafana

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

=> Saksham Choudhary :

~ Hello folks, I am AWS certified Cloud Architect Engineer. With having 5+ years of Experience in Teaching, I am currently providing cloud solutions for various products via my strong hands on DevOps Skills. I am a tech youtuber with 120k + subscriber and has taught 30,000 + students, Narcotics,

## Curriculum details :-

### => Linux Setting up an Environment :

- ~ Vagrant : What and Why?
- ~ Setting Up CentOS 7 via Vagrant in VBox
- ~ Basic Vagrant commands for virtual machine management
- ~ Vagrant Bridge Network
- ~ Update, Upgrade and Package Installations

### => Understanding Linux :

- ~ What is Linux?
- ~ Unix Vs Linux
- ~ Linux Distro & Applications
- ~ Cent OS vs Cent OS Stream
- ~ Significance of Symbol keys in Linux

### => System Access & File System :

- ~ Introduction to Command Prompt
- ~ Accessing Linux System
- ~ Introduction to Network Related commands
- ~ Connect Linux VM
- ~ Linux File System and Structure
- ~ Introduction to Root?
- ~ Absolute and Relative Paths
- ~ Directory Listing Attributes
- ~ Playing with File & Directories
- ~ Copying Directories
- ~ Different file types in Linux
- ~ Find and Locate - Files and Directories
- ~ Changing Password
- ~ Wildcards (\*,?,^,[])
- ~ Inode, Softlink & Hard Link

### => SysAdmin :

- ~ Linux File Editors
- ~ Stream Editing Commands
- ~ UAM (useradd, groupadd, usermod, userdel, groupdel)
- ~ Enable password aging
- ~ Switching user with sudo module
- ~ User communication (Users, wall, write)
- ~ Active directory commands (LDAP, IDM, WinBind, OpenLDAP)
- ~ System Utility commands (Date, Uptime, Hostname, Uname, which, cal)
- ~ Jobs and Scheduling
- ~ Systemctl and top command
- ~ Kill Commands
- ~ Crontab
- ~ at commands
- ~ Additional cronjobs scheduling
- ~ Process management
- ~ Sys Monitoring
- ~ Log analysis
- ~ Taking control on (Shutdown, init, reboot, halt)
- ~ Hostname management (hostnamectl, uname, dmidecode)
- ~ Sys Architecture
- ~ Terminal controls
- ~ Password recovery
- ~ setuid & setgid
- ~ Environment variables

### => Software Management :

- ~ GNU Project
- ~ Compiling software from code
- ~ Repositories
- ~ Apt Cache
- ~ Updating & Upgrading softwares
- ~ Uninstalling a software

### => Shell Scripting 1st Use case :

- ~ How is Shell Scripting in Linux Environment?
- ~ Getting started with Shell Scripting
- ~ Naming
- ~ Permissions
- ~ Variables
- ~ Builtins
- ~ Special Variables
- ~ Pseudocode
- ~ Command substitutions
- ~ If Statements
- ~ Conditionals
- ~ Exit statuses
- ~ Return codes
- ~ String test conditionals
- ~ Use Case : Reading standard input, creating accounts, Username Conventions

### => Shell Scripting 2nd Use case :

- ~ Random data
- ~ Cryptographic Hash Functions
- ~ Text and String Manipulations
- ~ Positional Parameters

- ~ Arguments
- ~ For Loops
- ~ Special Parameters
- ~ While Loops
- ~ Infinite Loop
- ~ Shifting
- ~ Sleeping
- ~ Use Case : Password Generation

=> Shell Scripting 3rd Use case :

- ~ Case Statements
- ~ Functions
- ~ Parsing command line
- ~ Parsing command line functions part 2
- ~ Finding file
- ~ Playing around with Userdel commands
- ~ Exploring archives with tar
- ~ Use Case: Deleting and Disabling User Accounts

=> Shell Scripting 4th Use case :

- ~ Cut and Awk
- ~ Cut and Awk Demonstration Script
- ~ Open Network Ports
- ~ Sort
- ~ Uniq
- ~ Use Case : Parsing log files

=> Networking :

- ~ TCP/IP
- ~ Dns
- ~ Hostnames
- ~ DHCP
- ~ Dynamic & Static addressing
- ~ Troubleshooting part 1
- ~ Troubleshooting part 2

=> Final Automation of Systems :

- ~ Configuring Mini Network
- ~ Scripting for Remote systems
- ~ Introduction to Scripting Remote commands
- ~ Scripting remote commands Advance part 1
- ~ Scripting remote commands Part 2

=> Why we need GIT :

- ~ Why GIT is important
- ~ Collaboration and Forking
- ~ Installation of GIT
- ~ Autocompletion of GIT

=> Git foundation :

- ~ GIT Architecture and Github Gitlab and bitbucket
- ~ Initializing and exploring GIT
- ~ First commit and log messages
- ~ Git checksum and SHA-1
- ~ Understanding HEAD and Checksum

=> Getting into files :

- ~ Lets do it again
- ~ Track difference between files
- ~ Delete from repos
- ~ repo reset and unstaging
- ~ Can we reset commits?

=> Git Snapshots :

- ~ checkout with previous versions
- ~ Soft, Mixed and Hard reset
- ~ Ignoring the files
- ~ What to ignore
- ~ Gitignore will not listen
- ~ This is not even in git docs

=> GIT for team managements :

- ~ Git tree listing
- ~ Git log in detail

=> GIT branches :

- ~ Git Branching basics
- ~ Creating a new branch in GIT
- ~ Checkout branches
- ~ RD of branches in GIT

=> GIT Merging :

- ~ Basics of Merging in GIT
- ~ Fast forward
- ~ Conflicts and merging in GIT
- ~ Stashing a branch
- ~ Stashing in multiple branches
- ~ Clean your stash

=> GIT and GitHub :

- ~ GIT online hosting
- ~ Creating a repo at Github

- ~ Uploading local repo to remote repo
- ~ Push for a remote collab
- ~ Merging from origin master
- ~ Assisting on open source projects

#### => Tags and Tickets :

- ~ Concepts of tickets and tags
- ~ Pushing tags to github

#### => Course Intro :

- ~ AWS Architect for real world

#### => Getting started with AWS and IAM :

- ~ FAQ for aws architect course
- ~ Getting started with AWS and expectation
- ~ Tour of AWS console with ROOT user
- ~ AWS Infra - Region and AZ
- ~ Securing root account and MFA
- ~ Custom signin link for IAM
- ~ Why groups are created
- ~ Creating groups and users
- ~ What are roles in IAM
- ~ Temporary security credentials in IAM
- ~ Billing alarms in Cloudwatch
- ~ Password compliance
- ~ buying domain on Route 53

#### => Amazon Elastic Compute Cloud -EC2 :

- ~ What is Elastic Compute
- ~ Instance types and limits
- ~ your first EC2 instance
- ~ In depth guide for EC2 options
- ~ Connecting to cloud instance
- ~ Configure an AWS web server
- ~ Stress testing, Cloud watch alarms and clean up
- ~ What are user data scripts
- ~ What is instance meta-data
- ~ Docs and hands on with Elastic IP
- ~ Custom network interface cards in AWS
- ~ creating custom AMI
- ~ Launch with custom image and clean up
- ~ Placement groups - Cluster, partition and Spread
- ~ EC2 pricing - OnDemand, spot and reserved
- ~ Just for Exam

#### => Virtual Private Cloud - VPC :

- ~ Why you should focus more on VPC
- ~ Understand the default VPC
- ~ Create diagram of default VPC
- ~ CIDR deep dive
- ~ Your custom VPC
- ~ Creating subnet in custom VPC
- ~ Internet Gateway and route table
- ~ Lets complete the diagram
- ~ VPC DNS hostname and resolution
- ~ updates from corporate in VPC structure
- ~ Clean up the resources
- ~ Security groups VS NACL
- ~ Understand the next diagram for VPC
- ~ Diagram 2 - VPC and subnets
- ~ Diagram 2 - Route tables and IGW
- ~ Congigure a NAT gateway
- ~ SSH agent forwarding
- ~ Bastion host and testing of diagram
- ~ Bastion host and testing of diagram part 2
- ~ NAT instance and configurations
- ~ VPC peering connection
- ~ What are transit gateways
- ~ A use case of Endpoints in VPC
- ~ preparing logs for audit - flowLogs
- ~ Resources for hybrid cloud - VPN and more
- ~ Lets audit the logs with Athena and Glue
- ~ Egress gateway cloudhub and clean up

#### => Load Balancing and scalability :

- ~ What are load balancers
- ~ Type of Load Balancer
- ~ Prep work for load balancers
- ~ Configure target groups
- ~ Creating an Application load balancer
- ~ Path and HOST based routing on domain
- ~ Cross Zone load balancer
- ~ Case of Sticky session
- ~ Clean up for ALB
- ~ Network Load Balancer
- ~ Scaling - Horizontal and Vertical
- ~ Auto Scaling Group configuration
- ~ Clean up for ASG resources

#### => Route 53 in Depth :

- ~ Welcome to Route 53



- ~ What are hosted zone - Public and Private
- ~ AWS DNS records - A and Alias
- ~ Creating instance in multiple region
- ~ Route 53 Health Checks
- ~ Simple and weighted route policy
- ~ FailOver and latency based policies
- ~ Multi value and restricting content on geo location
- ~ Clean up for Route 53

#### => Storage in AWS - S3 :

- ~ lets start with AWS storage
- ~ Introduction to S3 buckets
- ~ Permissions in S3 buckets
- ~ Static website hosting in S3 buckets
- ~ S3 bucket - Versioning and encryption
- ~ S3 event notifications
- ~ Access log BILLs and requester pays
- ~ S3 storage class
- ~ Data replication - CRR and SRR
- ~ S3 Select, Athena and Redshift - Query
- ~ Data life cycle policy
- ~ Getting started with cloudfront and OAI
- ~ Setup a cloudfront and OAI for a website

#### => Storage - Block and Object :

- ~ Instance Store - ephemeral
- ~ Types of EBS volume and IOPS
- ~ Creating and mounting EBS volume
- ~ Getting a snapshot of EBS
- ~ Re attach EBS volume
- ~ Data migration between AZ and Region
- ~ RAID 0 and 1 config
- ~ Creating and mounting Elastic File Storage
- ~ FSx for Windows and Lustre
- ~ Storage Gateway - Hybrid cloud
- ~ Storage Gateway NOT by LCO

#### => Databases in AWS :

- ~ Introduction to Databases in AWS
- ~ OLTP vs OLAP
- ~ Production level RDS walkthrough
- ~ Create a mysql db in AWS
- ~ Multi AZ replica RDS
- ~ Creating read replicas
- ~ Read Replica VS Multi AZ deployment
- ~ AWS aurora Docs walkthrough
- ~ Getting started with DynamoDB
- ~ Creating a table in DynamoDB
- ~ Reading the DAX Docs
- ~ ElasticCache memcached
- ~ ElasticCache Redis and Redis cluster
- ~ Redshift Overview

#### => Application integration in AWS :

- ~ Application integration services by AWS
- ~ Simple queue service
- ~ Creating our first queue service
- ~ FIFO vs standard queue
- ~ Delay, visibility and retention time
- ~ Dead letter queue
- ~ Long polling and short polling
- ~ Attaching lambda to SQS
- ~ Clean up all the sqs resources
- ~ Step function and simple workflow service
- ~ Amazon MQ, Rabbit MQ and other services

#### => PAAS and IAAS in AWS :

- ~ Getting started with PAAS and IAAS
- ~ Cloudformation inDepth guide
- ~ Beanstack application deployment

#### => Process and Migrate the Data :

- ~ Kinesis and shards
- ~ Kinesis analytics and firehose
- ~ What is Elastic MapReduce
- ~ What is Athena, Glue and Glue Studio
- ~ Import from other Virtualization Services
- ~ Database Migration service and Schema Conversion Tool

#### => Security Compliance :

- ~ Security and Compliance - SOX, PCI and more
- ~ Key Management Service
- ~ Hardware Security Module in Cloud
- ~ AWS WAF and shield service
- ~ Active Directory in AWS
- ~ What is AWS Cognito
- ~ AWS single sign on
- ~ AWS Directory service

#### => Container Service :

- ~ What are container service in AWS

- ~ What is Docker
- ~ What is Elastic Container Registry
- ~ What are microservices
- ~ What is Elastic Container service
- ~ What is Fargate
- ~ What is Elastic Kubernetes Service
- ~ AWS walkthrough for ECS and EKS

#### => AWS Serverless :

- ~ Getting started with AWS serverless
- ~ A common warning for AWS
- ~ Route 53
- ~ Get Started with S3 bucket
- ~ Struggle of web page hosting
- ~ Hosting with policies
- ~ GET vs POST and handling response
- ~ Your first lambda in AWS
- ~ Lambda permission and cloud watch
- ~ Introducing API gateway
- ~ Lambda for POST information
- ~ Post Data and CORS error
- ~ First look at SES
- ~ New user for SES and lambda
- ~ Sending email from SES and lambda

#### => Cracking AWS Certificate :

- ~ How to crack AWS Certification Exams

#### => Preparing for CCP :

- ~ How to crack AWS Certification CCP Exam

#### => Preparing for Associate/Architect Exam :

- ~ How to crack AWS Associate/Architect Certification Exam

#### => First interaction with python basics :

- ~ Indents and comments
- ~ take input from user and challenge
- ~ input challenge solution
- ~ getting started with variables in python

#### => Operations in Python :

- ~ Arithmetic and comparison operators in python
- ~ Logical operations in python
- ~ Membership and identity operations in python

#### => Conditionals and loops :

- ~ Introduction to conditionals
- ~ Design a rating system in python
- ~ While - Getting started with loops in python

#### => Detail analysis of data types :

- ~ Randomness in python
- ~ Using math library in python
- ~ String are powerful in python

#### => Functions Files and Exceptions :

- ~ getting started with functions in python
- ~ Multiple arguments in python
- ~ lambda in python
- ~ design custom modules in python

#### => Python challenges for fun :

- ~ Prime number and challenges
- ~ range of prime numbers
- ~ finding factorials
- ~ Get matrix input and print it

#### => Object Oriented programming in python :

- ~ Introduction to class
- ~ objects and constructor in python
- ~ Getters and setters in python
- ~ Inheritance from Samsung to iphone

#### => Database TODO App :

- ~ Read sqlite3 documentation first
- ~ Database helper in sqlite3 part 1
- ~ database helper file part 2
- ~ Debugging and finishing the app

#### => Advance side of python :

- ~ Iterator and generators in python
- ~ Maps and sets in python
- ~ All and any functions in python
- ~ Collections and deque

#### => Handling API in Python :

- ~ Requests and JSON handling in python
- ~ Get a unique user every time - Project

#### => Docker Installation Basics :

- ~ What is Docker?
- ~ How to install Docker and Hello World
- ~ What is container in Docker

- ~ Docker vs Virtual Machine
- ~ First interaction with busy box image

#### => Fundamentals of docker :

- ~ Docker lifecycle and PS
- ~ Start and delete a container
- ~ Getting a mongodb container for fun
- ~ Exploring exec command
- ~ Multiple ways to get inside a container

#### => Custom Docker images :

- ~ Analogy for custom docker image
- ~ Our first base image and custom image
- ~ Behind the scene for custom image

#### => Project and Docker :

- ~ Introduction to node project for docker
- ~ Introduction to node project for docker part 2
- ~ Containerize a node application
- ~ Performance upgrade in node project container

#### => Multi container setup :

- ~ Introduction to multi docker container
- ~ A mini mongo connector project
- ~ Put your node code in a container

#### => Ngnix - production grade deployment :

- ~ Ngnix A production grade docker
- ~ Attaching volumes in Docker
- ~ Types of docker files
- ~ Dev test and production stages

#### => Docker AWS and Travis CI :

- ~ Docker CI and AWS
- ~ What is CI CD Jenkins vs Travis CI
- ~ Moving to AWS Elastic Beanstalk
- ~ Moving project to github repo
- ~ Reading Travis CI documentation

#### => What is Kubernetes? :

- ~ What is Kubernetes?
- ~ Introduction to Kubernetes
- ~ Kubernetes History
- ~ Kubernetes Architecture
- ~ Kubernetes Architecture - In-depth

#### => Provisioning Infrastructure :

- ~ Provisioning Kubernetes Infrastructure on AWS
- ~ Provisioning Kubernetes Infrastructure on GCP
- ~ Installing Kubernetes using kubeadm
- ~ Setting up K8 using kubeadm

#### => Installing kubectl and minikube :

- ~ What is minikube?
- ~ What is kubectl?
- ~ Install minikube and kubectl

#### => Installing Kubernetes Using microk8s :

- ~ Setting up K8 using microk8's

#### => Installing Kubernetes Using K3s :

- ~ Setting up K8's using K3's

#### => Kubernetes Components :

- ~ Node & Pod
- ~ Service & Ingress
- ~ ConfigMap & Secret
- ~ Volumes
- ~ Deployment & StatefulSet

#### => Create and start a minikube cluster in the local environment Kubernetes CLI :

- ~ Commands with Example (kubectl)
- ~ Create a pod/deployment
- ~ Change the pod/deployment configuration
- ~ Debugging pods
- ~ Delete pod/deployment
- ~ Kubernetes YAML Configuration
- ~ Different attributes of a Kubernetes config file
- ~ Creating config files

#### => Kubernetes Namespace :

- ~ What is a Namespace?
- ~ 4 Default Namespaces
- ~ Create a Namespace and resources
- ~ Why use Namespaces?

#### => Kubernetes Healthchecks :

- ~ What is Ingress?
- ~ Creating YAML Config Files for Ingress
- ~ How to configure Ingress in your cluster?
- ~ What is Ingress Controller?

#### => Statefulset in Kubernetes :

- ~ What is StatefulSet?

- ~ *Deployment of Stateful and Stateless Application*
- ~ *Deployment vs StatefulSet*

#### => Kubernetes Services :

- ~ *What is a Service?*
- ~ *ClusterIP Services*
- ~ *Headless Services*
- ~ *NodePort Services*
- ~ *LoadBalancer Services*

#### => Volumes in Kubernetes :

- ~ *Persistent Volume (PV)*
- ~ *Persistent Volume Claim (PVC)*
- ~ *Storage Class (SC)*

#### => Deploying Microservices App to Kubernetes Cluster :

- ~ *Microservice Overview*
- ~ *Adding Dockerfile and Dockerfile Plugins*
- ~ *Adding configurations for Service Registry*
- ~ *Creating Kubernetes Config files (YAML)*

#### => Ansible :

- ~ *Getting started with Ansible*
- ~ *PlayBook Run and Lab Configurations*
- ~ *Ansible Modules Yaml Syntax*
- ~ *Variables*

#### => Terraform :

- ~ *Getting started with Terraform*
- ~ *Understand Infrastructure as Code (IaC) concepts*
- ~ *Terraform Provider Basics*
- ~ *Variables, Resource Attributes and Dependencies*
- ~ *Terraform State*
- ~ *Intro to alternatives*

#### => Pulumi :

- ~ *Getting started with pulmi*
- ~ *Syntax understanding*
- ~ *laac with python aws*

#### => CI-CD :

- ~ *Jenkins*
- ~ *Github Actions*
- ~ *Argo CD*

#### => Jenkins :

- ~ *Getting started with Jenkins*
- ~ *Jenkins Plugins and Integrations*
- ~ *Jenkins UI*
- ~ *Systems Administration With Jenkins*
- ~ *Pipeline*
- ~ *Conclusion*

#### => Github workflows and Actions :

- ~ *Getting started with Github*
- ~ *Events*
- ~ *Schedulers*
- ~ *External Triggers*
- ~ *Environment Variables*

#### => Prometheus :

- ~ *Getting started with Prometheus*
- ~ *Architecture of Prometheus server*
- ~ *Installation*
- ~ *Exporters*
- ~ *Alerting*

#### => Grafana :

- ~ *Introduction , Setup and Configuration*
- ~ *Grafana UI Tour*
- ~ *Integration with different data sources*
- ~ *Security and Administration of Grafana*

# House Price Prediction

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING PROJECT

Course link : <https://ineuron.ai/course/House-Price-Prediction>

## Course Description :-

The selling price of a property in a specific area can be determined with the use of house price predictions, and consumers can choose the ideal moment to purchase a home. In this project, "House Price Prediction Using Machine Learning," our goal is to develop a machine learning model to forecast house prices in the State of California using data from the census.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => House Price Prediction
- => Preparing Dataset And Basic Analysis
- => Preparing Dataset For Model Training
- => Training the Model
- => Performance Metrics
- => Creating A Flask Web Application
- => Deployment

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- House Price Prediction :
  - ~ Understanding the dataset
  - ~ Preparing Dataset And Basic Analysis
  - ~ Preparing Dataset For Model Training
  - ~ Training the Model
  - ~ Performance Metrics
  - ~ Prediction Of New Data
  - ~ Pickling the model File
  - ~ Setting up Github And VS Code
  - ~ Tools And Softwares Required
  - ~ Creating A New Environment
  - ~ Setting up Git
  - ~ Creating A Flask Web Application
  - ~ Running And Testing Our Application
  - ~ Prediction From Front End Application
  - ~ Procfile For Heroku Deployment
  - ~ Deploying App To Heroku
  - ~ Deploying the App Using Dockers

# Data Structure and Algorithms with competitive programming

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/Data-Structure-and-Algorithms-with-competitive-programming>

## Course Description :-

Data Structure and Algorithms for Beginners to Advance entire course will be discussed in python language and all the implementation and project will be done by using python .

## Course Features :-

- => Online Instructor-led learning: Live teaching by instructors
- => Every week doubt clearing session after the live classes
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Assignment in all the module
- => Quiz in every module
- => Everything will be discussed with python

## What you will learn :-

- => Analysis in Algorithms
- => Divide and Conquer
- => Greedy Technique
- => Dynamic Programming
- => Arrays
- => Linked List
- => Skip List
- => Hashing
- => Tree
- => Graph Traversal
- => Tree Traversal
- => Programming
- => Stack
- => Queue
- => String Matching
- => NP-Hard and NP-Complete Problems

## Requirements :-

- => Dedication
- => PC with internet connectivity

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving an analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

=> Analysis in Algorithms :

- ~ Introduction to Algorithms Preview
- ~ Analyzing Algorithm Preview
- ~ Asymptotic Notation Preview
- ~ Big O
- ~ Omega
- ~ Theta
- ~ Recurrence Relation Solving
- ~ Substitution Method
- ~ Recursive Tree Method
- ~ Master's Theorem

=> Divide and Conquer :

- ~ Introduction to Divide and Conquer
- ~ Discussion of applications of Divide and Conquer
- ~ Finding of maxima and minima

- ~ *Finding Power of an Element*
- ~ *Binary Search*
- ~ *Quicksort*
- ~ *Mergesort*
- ~ *Strassen's Matrix Multiplication*
- ~ *Maximum-subarray problem*
- ~ *Finding of number of inversions*

#### => Greedy Technique :

- ~ *Introduction to Greedy Techniques*
- ~ *Discussion of applications of Greedy Technique*
- ~ *Knapsack Problem*
- ~ *Job Sequencing with deadline*
- ~ *Huffman Coding*
- ~ *Optimal Merge Pattern*
- ~ *Minimum Cost Spanning Tree*
- ~ *Kruskal Algorithm*
- ~ *Prim's Algorithm*
- ~ *Single Source Shortest Path*
- ~ *Dijkstra's Algorithm*
- ~ *Bellmanford Algorithm*

#### => Dynamic Programming :

- ~ *Introduction to Dynamic Programming*
- ~ *Discussion of applications of Dynamic Programming*
- ~ *Fibonacci Series*
- ~ *Longest Common Subsequence*
- ~ *0/1 Knapsack*
- ~ *Sum of Subset*
- ~ *All Shortest Path*
- ~ *Matrix Chain Multiplication*

#### => Arrays :

- ~ *Introduction to Arrays*
- ~ *One Dimensional Array - How to find the address of an element in an array*
- ~ *Two Dimensional Array*
- ~ *Row Major Order*
- ~ *Column Major Order*
- ~ *Searching in an array*
- ~ *Linear Search*
- ~ *Binary Search(Discussed in DAC)*
- ~ *Sorting of an array*
- ~ *Comparison Sort*
- ~ *Selection Sort*
- ~ *Bubble Sort*
- ~ *Insertion Sort*
- ~ *Quicksort(Discussed in DAC)*
- ~ *Mergesort(Discussed in DAC)*
- ~ *Non Comparison Sort*
- ~ *Radix Sort*
- ~ *Bucket Sort*
- ~ *Count Sort*

#### => Linked List :

- ~ *Introduction to Linked List*
- ~ *Searching in Linked List*
- ~ *Deleting from a Linked List*
- ~ *Doubly Linked List*
- ~ *Reversal in linked list*

#### => Skip List :

- ~ *Introduction to Skip List*
- ~ *Operations and Randomization in Skip Lists*
- ~ *Insertion and Deletion in Skip Lists*
- ~ *Complexity analysis*

#### => Hashing :

- ~ *Introduction to Hashing*
- ~ *Hash Tables*
- ~ *Hash Functions*
- ~ *Collision Resolution Techniques*
- ~ *Chaining*
- ~ *Open Addressing*
- ~ *Linear Probing*
- ~ *Quadratic Probing*
- ~ *Double Hashing*
- ~ *Perfect Hashing*
- ~ *Analysis of Chaining*
- ~ *Analysis of Open Addressing*
- ~ *Application of Hashing : Bloom Filters Discussion*

#### => Tree :

- ~ *Introduction to Binary Tree*
- ~ *Binary Search Tree*
- ~ *AVL Tree - Creation , Insertion, Deletion*
- ~ *Red Black Tree - Creation , Insertion, Deletion*
- ~ *BTree and B+ Tree - Creation , Insertion, Deletion*

#### => Graph Traversal :

- ~ *Breadth First Search*
- ~ *Depth First Search*

=> Tree Traversal :

- ~ Preorder Traversal
- ~ Postorder Traversal
- ~ Inorder Traversal

=> Programming :

- ~ Static and Dynamic Scoping
- ~ Static Variable
- ~ Pointers

=> Stack :

- ~ Introduction to Stack Data Structure
- ~ Implementation of Stack Using Arrays
- ~ Implementation of Stack Using Linked List
- ~ Average Stack Lifetime of an element
- ~ Implementing multiple stacks in single array
- ~ Applications of Stack
- ~ Recursion
- ~ Tail Recursion
- ~ Non-Tail Recursion
- ~ Nested Recursion
- ~ Indirect Recursion
- ~ Infix to Postfix
- ~ Prefix to Postfix
- ~ Postfix Evaluation
- ~ Towers of Hanoi
- ~ Fibonacci Series

=> Queue :

- ~ Introduction to Queue Data Structure
- ~ Implementation of Queue Using Arrays
- ~ Implementation of Queue Using Linked List
- ~ Circular Queue
- ~ Priority Queue
- ~ Implementation of Stack using Queue

=> String Matching :

- ~ Naive String Matching Algorithms
- ~ Rabin-Karp Algorithm
- ~ String Matching with finite automata

=> NP-Hard and NP-Complete Problems :

- ~ NP-Hard
- ~ NP-Complete Problem



# Be A DevOps Pro Tech Neuron

---

Topic Name : DEVOPS

Sub-topic Name : DEVOPS MASTERS

Course link : <https://ineuron.ai/course/Be-A-DevOps-Pro-Tech-Neuron>

## Course Description :-

DevOps, which is a mix of cultural principles, practices, and technologies such as Linux foundations, Docker, Kubernetes, Ansible, Terraform, ArgoCD, AWS Cloud, Git, Git, Prometheus, and others, increases an organization's capacity to build applications and services at high velocity. Become a DevOps master and increase the velocity of production now.

## Course Features :-

- => Online Instructor-led learning
- => Doubt Clearing
- => Proper Roadmap for DevOps
- => Lifetime Dashboard access
- => Recording of Live Class
- => Material
- => Interview Questions
- => Resume Building
- => Career Guidance
- => Quiz in every module - Based on Real Certification Questions Based
- => Certificate
- => Industry Level Projects and Case studies
- => Major Projects
- => Weekly Assignments

## What you will learn :-

- => Devops
- => Linux
- => Git
- => AWS
- => Docker
- => Ansible
- => Kubernetes
- => Terraform
- => CI CD Pipelines
- => Argo CD
- => Prometheus
- => Grafana

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

=> Saksham Choudhary :

~ Hello folks, I am AWS certified Cloud Architect Engineer. With having 5+ years of Experience in Teaching, I am currently providing cloud solutions for various products via my strong hands on DevOps Skills. I am a tech youtuber with 120k + subscriber and has taught 30,000 + students, Narcotics, Custom duty officers, Police officials and Corporate candidates.

=> Ritesh Yadav :

~ Ritesh is truly passionate about data science, machine learning and DevOps in general, he likes what he does, and is keen to learn. Currently, He is working as a Jr. Data Scientist at Ineuron.ai. He also loves to Contribute to Open Source Projects, which are mainly under CNCF Landscape. Ritesh loves to work in Cloud-Native technologies and Golang ( Go ). Apart from this, Ritesh has been actively involved in the open-source community for over a year, helping many open-source DevOps tools and CNCF Projects like Porter, Meshery, Keptn, TensorFlow, and Thanos through his contributions.

## Curriculum details :-

### => Linux Setting up an Environment :

- ~ *Vagrant : What and Why?*
- ~ *Setting Up CentOS 7 via Vagrant in VBox*
- ~ *Basic Vagrant commands for virtual machine management*
- ~ *Vagrant Bridge Network*
- ~ *Update, Upgrade and Package Installations*

### => Understanding Linux :

- ~ *What is Linux?*
- ~ *Unix Vs Linux*
- ~ *Linux Distros & Applications*
- ~ *Cent OS vs Cent OS Stream*
- ~ *Significance of Symbol keys in Linux*

### => System Access & File System :

- ~ *Introduction to Command Prompt*
- ~ *Accessing Linux System & configuring Putty*
- ~ *Introduction to Network Related commands*
- ~ *Connect Linux VM via Putty*
- ~ *Linux File System and Structure*
- ~ *Introduction to Root?*
- ~ *Absolute and Relative Paths*
- ~ *Directory Listing Attributes*
- ~ *Playing with File & Directories*
- ~ *Copying Directories*
- ~ *Different file types in Linux*
- ~ *Find and Locate - Files and Directories*
- ~ *Changing Password*
- ~ *Wildcards (\*,?,^,[],)*
- ~ *Inode, Softlink & Hard Link*

### => SysAdmin :

- ~ *Linux File Editors*
- ~ *Stream Editing Commands*
- ~ *UAM (useradd, groupadd, usermod, userdel, groupdel)*
- ~ *Enable password aging*
- ~ *Switching user with sudo module*
- ~ *User communication (Users, wall, write)*
- ~ *Active directory commands (LDAP, IDM, WinBind, OpenLDAP)*
- ~ *System Utility commands (Date, Uptime, Hostname, Uname, which, cal)*
- ~ *Jobs and Scheduling*
- ~ *Systemctl and top command*
- ~ *Kill Commands*
- ~ *Crontab*
- ~ *at commands*
- ~ *Additional cronjobs scheduling*
- ~ *Process management*
- ~ *Sys Monitoring*
- ~ *Log analysis*
- ~ *Taking control on (Shutdown, init, reboot, halt)*
- ~ *Hostname management (hostnamectl, uname, dmidecode)*
- ~ *Sys Architecture*
- ~ *Terminal controls*
- ~ *Password recovery*
- ~ *setuid & setgid*
- ~ *Environment variables*

### => Software Management :

- ~ *GNU Project*
- ~ *Compiling software from code*
- ~ *Repositories*
- ~ *Apt Cache*
- ~ *Updating & Upgrading softwares*
- ~ *Uninstalling a software*

### => Shell Scripting 1st Use case :

- ~ *How is Shell Scripting in Linux Environment?*
- ~ *Getting started with Shell Scripting*
- ~ *Naming*
- ~ *Permissions*
- ~ *Variables*
- ~ *Builtins*
- ~ *Special Variables*
- ~ *Pseudocode*
- ~ *Command substitutions*
- ~ *If Statements*
- ~ *Conditionals*
- ~ *Exit statuses*
- ~ *Return codes*
- ~ *String test conditionals*
- ~ *Use Case : Reading standard input, creating accounts, Username Conventions*

### => Shell Scripting 2nd Use case :

- ~ *Random data*
- ~ *Cryptographic Hash Functions*
- ~ *Text and String Manipulations*
- ~ *Positional Parameters*
- ~ *Arguments*

- ~ For Loops
- ~ Special Parameters
- ~ While Loops
- ~ Infinite Loop
- ~ Shifting
- ~ Sleeping
- ~ Use Case : Password Generation

#### => Shell Scripting 3rd Use case :

- ~ Case Statements
- ~ Functions
- ~ Parsing command line
- ~ Parsing command line functions part 2
- ~ Finding file
- ~ Playing around with Userdel commands
- ~ Exploring archives with tar
- ~ Use Case: Deleting and Disabling User Accounts

#### => Shell Scripting 4th Use case :

- ~ Cut and Awk
- ~ Cut and Awk Demonstration Script
- ~ Open Network Ports
- ~ Sort
- ~ Uniq
- ~ Use Case : Parsing log files

#### => Networking :

- ~ TCP/IP
- ~ Dns
- ~ Hostnames
- ~ DHCP
- ~ Dynamic & Static addressing
- ~ Troubleshooting part 1
- ~ Troubleshooting part 2

#### => Final Automation of Systems :

- ~ Configuring Mini Network
- ~ Scripting for Remote systems
- ~ Introduction to Scripting Remote commands
- ~ Scripting remote commands Advance part 1
- ~ Scripting remote commands Part 2

#### => Why we need GIT :

- ~ Why GIT is important
- ~ Collaboration and Forking
- ~ Installation of GIT
- ~ Autocompletion of GIT

#### => Git foundation :

- ~ GIT Architecture and Github Gitlab and bitbucket
- ~ Initializing and exploring GIT
- ~ First commit and log messages
- ~ Git checksum and SHA-1
- ~ Understanding HEAD and Checksum

#### => Getting into files :

- ~ Lets do it again
- ~ Track difference between files
- ~ Delete from repos
- ~ repo reset and unstaging
- ~ Can we reset commits?

#### => Git Snapshots :

- ~ checkout with previous versions
- ~ Soft, Mixed and Hard reset
- ~ Ignoring the files
- ~ What to ignore
- ~ Gitignore will not listen
- ~ This is not even in git docs

#### => GIT for team managements :

- ~ Git tree listing
- ~ Git log in detail

#### => GIT branches :

- ~ Git Branching basics
- ~ Creating a new branch in GIT
- ~ Checkout branches
- ~ RD of branches in GIT

#### => GIT Merging :

- ~ Basics of Merging in GIT
- ~ Fast forward
- ~ Conflicts and merging in GIT
- ~ Stashing a branch
- ~ Stashing in multiple branches
- ~ Clean your stash

#### => GIT and GitHub :

- ~ GIT online hosting
- ~ Creating a repo at Github
- ~ Uploading local repo to remote repo

- ~ Push for a remote collab
- ~ Merging from origin master
- ~ Assisting on open source projects

#### => Tags and Tickets :

- ~ Concepts of tickets and tags
- ~ Pushing tags to github

#### => Course Intro :

- ~ AWS Architect for real world

#### => Getting started with AWS and IAM :

- ~ FAQ for aws architect course
- ~ Getting started with AWS and expectation
- ~ Tour of AWS console with ROOT user
- ~ AWS Infra - Region and AZ
- ~ Securing root account and MFA
- ~ Custom signin link for IAM
- ~ Why groups are created
- ~ Creating groups and users
- ~ What are roles in IAM
- ~ Temporary security credentials in IAM
- ~ Billing alarms in Cloudwatch
- ~ Password compliance
- ~ buying domain on Route 53

#### => Amazon Elastic Compute Cloud -EC2 :

- ~ What is Elastic Compute
- ~ Instance types and limits
- ~ your first EC2 instance
- ~ In depth guide for EC2 options
- ~ Connecting to cloud instance
- ~ Configure an AWS web server
- ~ Stress testing, Cloud watch alarms and clean up
- ~ What are user data scripts
- ~ What is instance meta-data
- ~ Docs and hands on with Elastic IP
- ~ Custom network interface cards in AWS
- ~ creating custom AMI
- ~ Launch with custom image and clean up
- ~ Placement groups - Cluster, partition and Spread
- ~ EC2 pricing - OnDemand, spot and reserved
- ~ Just for Exam

#### => Virtual Private Cloud - VPC :

- ~ Why you should focus more on VPC
- ~ Understand the default VPC
- ~ Create diagram of default VPC
- ~ CIDR deep dive
- ~ Your custom VPC
- ~ Creating subnet in custom VPC
- ~ Internet Gateway and route table
- ~ Lets complete the diagram
- ~ VPC DNS hostname and resolution
- ~ updates from corporate in VPC structure
- ~ Clean up the resources
- ~ Security groups VS NACL
- ~ Understand the next diagram for VPC
- ~ Diagram 2 - VPC and subnets
- ~ Diagram 2 - Route tables and IGW
- ~ Congigure a NAT gateway
- ~ SSH agent forwarding
- ~ Bastion host and testing of diagram
- ~ Bastion host and testing of diagram part 2
- ~ NAT instance and configurations
- ~ VPC peering connection
- ~ What are transit gateways
- ~ A use case of Endpoints in VPC
- ~ preparing logs for audit - flowLogs
- ~ Resources for hybrid cloud - VPN and more
- ~ Lets audit the logs with Athena and Glue
- ~ Egress gateway cloudhub and clean up

#### => Load Balancing and scalability :

- ~ What are load balancers
- ~ Type of Load Balancer
- ~ Prep work for load balancers
- ~ Configure target groups
- ~ Creating an Application load balancer
- ~ Path and HOST based routing on domain
- ~ Cross Zone load balancer
- ~ Case of Sticky session
- ~ Clean up for ALB
- ~ Network Load Balancer
- ~ Scaling - Horizontal and Vertical
- ~ Auto Scaling Group configuration
- ~ Clean up for ASG resources

#### => Route 53 in Depth :

- ~ Welcome to Route 53
- ~ What are hosted zone - Public and Private

- ~ AWS DNS records - A and Alias
- ~ Creating instance in multiple region
- ~ Route 53 Health Checks
- ~ Simple and weighted route policy
- ~ FailOver and latency based policies
- ~ Multi value and restricting content on geo location
- ~ Clean up for Route 53

#### => Storage in AWS - S3 :

- ~ Lets start with AWS storage
- ~ Introduction to S3 buckets
- ~ Permissions in S3 buckets
- ~ Static website hosting in S3 buckets
- ~ S3 bucket - Versioning and encryption
- ~ S3 event notifications
- ~ Access log BILLS and requester pays
- ~ S3 storage class
- ~ Data replication - CRR and SRR
- ~ S3 Select, Athena and Redshift - Query
- ~ Data life cycle policy
- ~ Getting started with cloudfront and OAI
- ~ Setup a cloudfront and OAI for a website

#### => Storage - Block and Object :

- ~ Instance Store - ephemeral
- ~ Types of EBS volume and IOPS
- ~ Creating and mounting EBS volume
- ~ Getting a snapshot of EBS
- ~ Re attach EBS volume
- ~ Data migration between AZ and Region
- ~ RAID 0 and 1 config
- ~ Creating and mounting Elastic File Storage
- ~ FSx for Windows and Lustre
- ~ Storage Gateway - Hybrid cloud
- ~ Storage Gateway NOT by LCO

#### => Databases in AWS :

- ~ Introduction to Databases in AWS
- ~ OLTP vs OLAP
- ~ Production level RDS walkthrough
- ~ Create a mysql db in AWS
- ~ Multi AZ replica RDS
- ~ Creating read replicas
- ~ Read Replica VS Multi AZ deployment
- ~ AWS aurora Docs walkthrough
- ~ Getting started with DynamoDB
- ~ Creating a table in DynamoDB
- ~ Reading the DAX Docs
- ~ ElasticCache memcached
- ~ ElasticCache Redis and Redis cluster
- ~ Redshift Overview

#### => Application integration in AWS :

- ~ Application integration services by AWS
- ~ Simple queue service
- ~ Creating our first queue service
- ~ FIFO vs standard queue
- ~ Delay, visibility and retention time
- ~ Dead letter queue
- ~ Long polling and short polling
- ~ Attaching lambda to SQS
- ~ Clean up all the sqs resources
- ~ Step function and simple workflow service
- ~ Amazon MQ, Rabbit MQ and other services

#### => PAAS and IAAS in AWS :

- ~ Getting started with PAAS and IAAS
- ~ Cloudformation inDepth guide
- ~ Beanstack application deployment

#### => Process and Migrate the Data :

- ~ Kinesis and shards
- ~ Kinesis analytics and firehose
- ~ What is Elastic MapReduce
- ~ What is Athena, Glue and Glue Studio
- ~ Import from other Virtualization Services
- ~ Database Migration service and Schema Conversion Tool

#### => Security Compliance :

- ~ Security and Compliance - SOX, PCI and more
- ~ Key Management Service
- ~ Hardware Security Module in Cloud
- ~ AWS WAF and shield service
- ~ Active Directory in AWS
- ~ What is AWS Cognito
- ~ AWS single sign on
- ~ AWS Directory service

#### => Container Service :

- ~ What are container service in AWS
- ~ What is Docker

- ~ *What is Elastic Container Registry*
- ~ *What are microservices*
- ~ *What is Elastic Container service*
- ~ *What is Fargate*
- ~ *What is Elastic Kubernetes Service*
- ~ *AWS walkthrough for ECS and EKS*

#### => AWS Serverless :

- ~ *Getting started with AWS serverless*
- ~ *A common warning for AWS*
- ~ *Route 53*
- ~ *Get Started with S3 bucket*
- ~ *Struggle of web page hosting*
- ~ *Hosting with policies*
- ~ *GET vs POST and handling response*
- ~ *Your first lambda in AWS*
- ~ *Lambda permission and cloud watch*
- ~ *Introducing API gateway*
- ~ *Lambda for POST information*
- ~ *Post Data and CORS error*
- ~ *First look at SES*
- ~ *New user for SES and lambda*
- ~ *Sending email from SES and lambda*

#### => Cracking AWS Certificate :

- ~ *How to crack AWS Certification Exams*

#### => Preparing for CCP :

- ~ *How to crack AWS Certification CCP Exam*

#### => Preparing for Associate/Architect Exam :

- ~ *How to crack AWS Associate/Architect Certification Exam*

#### => Introduction and installation of python :

- ~ *Introduction to python course*
- ~ *Python Installation*
- ~ *Pycharm Installation on Windows*
- ~ *Installation of python on MAC*
- ~ *Installing Pycharm in MAC*
- ~ *Using VSCode for python- optional*

#### => First interaction with python basics :

- ~ *Indents and comments*
- ~ *take input from user and challenge*
- ~ *input challenge solution*
- ~ *getting started with variables in python*
- ~ *numbers and strings basics in python*
- ~ *Lists and tuples basics in python*
- ~ *Dictionary in python*

#### => Operations in Python :

- ~ *Arithmetic and comparison operators in python*
- ~ *Logical operations in python*
- ~ *Membership and identity operations in python*

#### => Conditionals and loops :

- ~ *Introduction to conditionals*
- ~ *Design a rating system in python*
- ~ *While - Getting started with loops in python*
- ~ *First step to read documentation*
- ~ *For loop in python*
- ~ *Break keyword in python loops*
- ~ *continue and pass keywords in python*

#### => Detail analysis of data types :

- ~ *Randomness in python*
- ~ *Using math library in python*
- ~ *String are powerful in python*
- ~ *Detail talk about lists in python*
- ~ *Tuples and dictionary talks in python*

#### => Functions Files and Exceptions :

- ~ *getting started with functions in python*
- ~ *Multiple arguments in python*
- ~ *lambda in python*
- ~ *design custom modules in python*
- ~ *Find the day assignment in python*
- ~ *Main method and file handling in python*
- ~ *Exception handling*

#### => Python challenges for fun :

- ~ *Prime number and challenges*
- ~ *range of prime numbers*
- ~ *finding factorials*
- ~ *Get matrix input and print it*

#### => Object Oriented programming in python :

- ~ *Introduction to class*
- ~ *objects and constructor in python*
- ~ *Getters and setters in python*
- ~ *Inheritance from Samsung to iphone*
- ~ *Method overriding in python*

## => Database TODO App :

- ~ *Read sqlite3 documentation first*
- ~ *Database helper in sqlite3 part 1*
- ~ *database helper file part 2*
- ~ *Debugging and finishing the app*

## => Advance side of python :

- ~ *Iterator and generators in python*
- ~ *Maps and sets in python*
- ~ *All and any functions in python*
- ~ *Collections and deque*

## => Handling API in Python :

- ~ *Requests and JSON handling in python*
- ~ *Get a unique user every time - Project*

## => Docker Installation Basics :

- ~ *What is Docker?*
- ~ *How to install Docker and Hello World*
- ~ *What is container in Docker*
- ~ *Docker vs Virtual Machine*
- ~ *First interaction with busy box image*

## => Fundamentals of docker :

- ~ *Docker lifecycle and PS*
- ~ *Start and delete a container*
- ~ *Getting a mongodb container for fun*
- ~ *Exploring exec command*
- ~ *Multiple ways to get inside a container*

## => Custom Docker images :

- ~ *Analogy for custom docker image*
- ~ *Our first base image and custom image*
- ~ *Behind the scene for custom image*
- ~ *Creating a custom mongodb image*
- ~ *Concept of caching in docker*
- ~ *Provide a custom name for your image*

## => Project and Docker :

- ~ *Introduction to node project for docker*
- ~ *Introduction to node project for docker part 2*
- ~ *Containerize a node application*
- ~ *Performance upgrade in node project container*

## => Multi container setup :

- ~ *Introduction to multi docker container*
- ~ *A mini mongo connector project*
- ~ *Put your node code in a container*
- ~ *Introduction to docker compose*
- ~ *Connect 2 compose images in docker*
- ~ *Access the compose container app with browser*

## => Ngnix - production grade deployment :

- ~ *Ngnix A production grade docker*
- ~ *Attaching volumes in Docker*
- ~ *Types of docker files*
- ~ *Dev test and production stages*
- ~ *Understand react project for docker deployment*
- ~ *Docker for development*
- ~ *Docker for testing*
- ~ *Docker for production*

## => Docker AWS and Travis CI :

- ~ *Docker CI and AWS*
- ~ *What is CI CD Jenkins vs Travis CI*
- ~ *Moving to AWS Elastic Beanstalk*
- ~ *Moving project to github repo*
- ~ *Reading Travis CI documentation*
- ~ *Writing our 1st travis CI config file*
- ~ *AWS IAM user generation*
- ~ *Elastic Beanstalk and S3 bucket*
- ~ *Finally hosting app on AWS with CI integrated with docker*
- ~ *TURN OFF those AWS apps*

## => What is Kubernetes? :

- ~ *What is Kubernetes?*
- ~ *Introduction to Kubernetes*
- ~ *Kubernetes History*
- ~ *Kubernetes Architecture*
- ~ *Kubernetes Architecture - In-depth*

## => Provisioning Infrastructure :

- ~ *Provisioning Kubernetes Infrastructure on AWS*
- ~ *Provisioning Kubernetes Infrastructure on GCP*
- ~ *Installing Kubernetes using kubeadm*
- ~ *Setting up K8 using kubeadm*

## => Installing kubectl and minikube :

- ~ *What is minikube?*
- ~ *What is kubectl?*
- ~ *Install minikube and kubectl*

=> Installing Kubernetes Using microk8s :

- ~ *Setting up K8 using microk8's*

=> Installing Kubernetes Using K3s :

- ~ *Setting up K8's using K3's*

=> Kubernetes Components :

- ~ *Node & Pod*
- ~ *Service & Ingress*
- ~ *ConfigMap & Secret*
- ~ *Volumes*
- ~ *Deployment & StatefulSet*

=> Create and start a minikube cluster in the local environment Kubernetes CLI :

- ~ *Commands with Example (kubectl)*
- ~ *Create a pod/deployment*
- ~ *Change the pod/deployment configuration*
- ~ *Debugging pods*
- ~ *Delete pod/deployment*
- ~ *Kubernetes YAML Configuration*
- ~ *Different attributes of a Kubernetes config file*
- ~ *Creating config files*

=> Kubernetes Namespace :

- ~ *What is a Namespace?*
- ~ *4 Default Namespaces*
- ~ *Create a Namespace and resources*
- ~ *Why use Namespaces?*

=> Kubernetes Healthchecks :

- ~ *What is Ingress?*
- ~ *Creating YAML Config Files for Ingress*
- ~ *How to configure Ingress in your cluster?*
- ~ *What is Ingress Controller?*
- ~ *Demo: Configure Ingress in Minikube*
- ~ *Ingress Config based on Paths*
- ~ *Ingress Config based on Domain and Subdomain*

=> Statefulset in Kubernetes :

- ~ *What is StatefulSet?*
- ~ *Deployment of Stateful and Stateless Application*
- ~ *Deployment vs StatefulSet*
- ~ *Pod Identity*
- ~ *Scaling database applications: Master and Worker Pods*

=> Kubernetes Services :

- ~ *What is a Service?*
- ~ *ClusterIP Services*
- ~ *Headless Services*
- ~ *NodePort Services*
- ~ *LoadBalancer Services*

=> Volumes in Kubernetes :

- ~ *Persistent Volume (PV)*
- ~ *Persistent Volume Claim (PVC)*
- ~ *Storage Class (SC)*

=> Deploying Microservices App to Kubernetes Cluster :

- ~ *Microservice Overview*
- ~ *Adding Dockerfile and Dockerfile Plugins*
- ~ *Adding configurations for Service Registry*
- ~ *Creating Kubernetes Config files (YAML)*
- ~ *Implementing API Gateway*
- ~ *Deploying applications to Kubernetes Cluster*
- ~ *Scaling Application*
- ~ *Kubernetes Dashboard*
- ~ *Deleting resources from Kubernetes Cluster*

=> Ansible :

- ~ *Getting started with Ansible*
- ~ *PlayBook Run and Lab Configurations*
- ~ *Ansible Modules Yaml Syntax*
- ~ *Variables*
- ~ *Playbook Flow*
- ~ *Include And roles*
- ~ *Conditionals and Loops*

=> Terraform :

- ~ *Getting started with Terraform*
- ~ *Understand Infrastructure as Code (IaC) concepts*
- ~ *Terraform Provider Basics*
- ~ *Variables, Resource Attributes and Dependencies*
- ~ *Terraform State*
- ~ *Use the Terraform CLI*
- ~ *Read, generate, and modify configuration*
- ~ *Terraform Modules*
- ~ *Terraform Cloud*
- ~ *Intro to alternatives*

=> Pulumi :

- ~ *Getting started with pulmi*
- ~ *Syntax understanding*



~ *laac with python aws*

=> CI-CD :

~ *Github Actions*  
~ *Jenkins*  
~ *Argo CD*

=> Github workflows and Actions :

~ *Getting started with Github*  
~ *Events*  
~ *Schedulers*  
~ *External Triggers*  
~ *Environment Variables*  
~ *Encrypting & Decrypting Files*  
~ *Using Functions in Expressions*  
~ *Strategy*  
~ *Matrix*  
~ *Docker containers on github actions*  
~ *Ci Cd Workflows to automate testing and deployment*

=> Prometheus :

~ *Getting started with Prometheus*  
~ *Architecture of Prometheus server*  
~ *Installation*  
~ *Exporters*  
~ *PromQL*  
~ *Client Libraries*  
~ *Quantification of Instruments*  
~ *Recording Rules*  
~ *Alerting*  
~ *Create Routing Tree for alerts*  
~ *PagerDuty - Slack Alerts*  
~ *BlackBox Exporters*  
~ *Pushgateway*  
~ *Service Gateway*  
~ *Aws With Prometheus*  
~ *Prometheus Http API*

=> Grafana :

~ *Introduction , Setup and Configuration*  
~ *Grafana UI Tour*  
~ *Integration with different data sources*  
~ *Grafana Templates*  
~ *Grafana Dashboards Introduction*  
~ *Application Dashboards*  
~ *Managing Dashboards*  
~ *Dynamic DashBoards*  
~ *Security and Administration of Grafana*

# DOMO BI

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/DOMO-BI>

## Course Description :-

Upskill your analytics skills through the power of DOMO BI Platform. Digital metrics, such as lead analytics, web traffic, Web Analytics, etc would be very much useful for a particular business and that can be achieved by learning this analytical platform. You will be able to enhance your skill by exploring this DOMO BI Analytical platform.

## Course Features :-

- => Live-Class Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => DOMO BI Features
- => DOMO BI Pricing
- => DOMO BI & Analytics
- => DOMO BI connectors
- => Dashboard
- => creating report
- => creating Dashboard

## Requirements :-

- => No prior knowledge in Analytics
- => System with Internet Connection
- => Interest to learn
- => Basic knowledge of BI
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> DOMO BI Introduction :

- ~ Overview Preview
- ~ Features of DomoBI Preview
- ~ DomoBI Analytics
- ~ DomoBI Pricing
- ~ DomoBI Connectors
- ~ Dashboard Overview

=> City Bike Project Analysis :

- ~ Project Overview
- ~ Loading Data
- ~ Start station
- ~ Trip duration
- ~ Gender
- ~ Sharing card
- ~ Dashboards
- ~ Alerts

# Recommender System

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING PROJECT

Course link : <https://ineuron.ai/course/Recommender-System>

## Course Description :-

Recommender System

## Course Features :-

- => Roadmap
- => Interview Questions and their approach discussions
- => Learn solving Scenario-based questions
- => Improve your skills and knowledge by solving different types of questions
- => Assignments
- => Quizzes
- => Challenges
- => Completion certificate

## What you will learn :-

- => Basic understanding of digital marketing tools
- => Search engine optimization techniques
- => How to work on Google ads
- => Social media marketing on various platforms
- => Email marketing using Mailchimp
- => Content creations like written, graphics and video

## Requirements :-

- => Understanding of basic marketing terminologies
- => A system with internet connection
- => Dedication

## Instructors :-

=> Boktiar Ahmed Bappy :

~ This is Bappy. I aim for simplicity in Data Science. Real Creativity won't make things more complex. Instead, I will simplify them, Interested in a Data Science Career and so develop myself accordingly. Data Scientist and lecturer with working experience in Machine Learning, Deep Learning, Microcontrollers and Electronics systems. Hands-on experience in classification, regression, clustering, computer vision, natural language processing and transfer learning models to solve challenging business problems. I have a huge interest in Robotics. I have innovated a lot of innovations, ideas, projects & robots and got a lot of achievements.

## Curriculum details :-

=> Book Recommender System :

- ~ Introduction & Demo Preview
- ~ What is recommender system & types Preview
- ~ Project Architecture
- ~ Data Collection
- ~ Data Loading
- ~ Analyzing data Part1
- ~ Analyzing data Part 2
- ~ Converting to pivot table
- ~ Model Building
- ~ Creating VENV
- ~ Making setup
- ~ Implementing web app
- ~ Deployment Preview

=> Movie Recommender System :

- ~ Introduction & Demo Preview
- ~ What is recommendation system & types?
- ~ Project Architecture
- ~ Data Collection
- ~ Data Loading
- ~ Analyzing data-Part1
- ~ Analyzing data-Part2
- ~ Generating embeddings
- ~ Creating VENV
- ~ Making setup
- ~ Implementing web app
- ~ Deployment Preview

# DART

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : DART

Course link : <https://ineuron.ai/course/DART>

## Course Description :-

Dart is a programming language designed for client development, such as for the web and mobile apps. It is developed by Google and can also be used to build server and desktop applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax.

## Course Features :-

- => Be able to program in Dart professionally
- => Master the Dart programming language by learning every concepts
- => Be able to build fully fledged apps with flutter(using dart)
- => Learn to use modern frameworks like Flutter in future

## What you will learn :-

- => Learn Dart from scratch to Classes & Objects
- => A step towards build mobile apps
- => Learn how to write control flow statements
- => Learn how to compile and debug the code
- => Learn all the basics without stopping after then: Dive deeply into Flutter & Dart and become an advanced developer

## Requirements :-

- => No prior experience in anything required.

## Instructors :-

- => Syed Ashraf :
  - ~ Full Stack Engineer at TensorGo Technologies

## Curriculum details :-

- => Introduction & Starting it up :

- ~ Introduction Preview
- ~ Installation
- ~ Setting it up
- ~ Comments
- ~ Programming Flow

- => Data Types :

- ~ Variables Preview
- ~ String Basics
- ~ Operators
- ~ Var Data Type

- => Collections :

- ~ Lists
- ~ Sets
- ~ Maps

- => Control Flow Statements :

- ~ If-Else
- ~ Loops
- ~ Other Loops
- ~ Break & Continue
- ~ Switch Case

- => Objects :

- ~ Functions
- ~ Classes & Objects

- => Extra :

- ~ Error Handling
- ~ Packages

# SQL Projects

---

Topic Name : DATA ANALYTICS

Sub-topic Name : SQL

Course link : <https://ineuron.ai/course/SQL-Projects>

## Course Description :-

SQL is utilised for a wide range of things, including banking, music, social media, data analysis, and so on. The majority of firms rely on huge, relational databases and are continually on the lookout for SQL experts. This course covers real world scenario based projects to gain hands-on knowledge and implement in real time to build business solutions.

## Course Features :-

- => Project source codes
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Start with Entity Relationship Model(ERM) logic.
- => Build stored procedure based on the business use case.
- => Database schema design

## Requirements :-

- => Prior knowledge of SQL.
- => A system with internet connection.
- => Your dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Python database connectivity (MYSQL) :
  - ~ Installing mysql Preview
  - ~ Database connectivity part 1
  - ~ Database connectivity part 2
  - ~ Database connectivity part 3
  - ~ Database connectivity part 4
- => Bank management system :
  - ~ Bank management system flow chart Preview
  - ~ Bank management system part 2
  - ~ Bank management system part 3
  - ~ Bank management system part 4

# Pro Database Management Systems

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Database-Management-Systems>

## Course Description :-

This course is designed mostly for computer science subject Database Management Systems test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> DBMS Theoretical Test

=> DBMS Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Database Management Systems Test :

- ~ DBMS Test 1
- ~ DBMS Test 2
- ~ DBMS Test 3
- ~ DBMS Test 4
- ~ DBMS Test 5
- ~ DBMS Test 6
- ~ DBMS Test 7
- ~ DBMS Test 8
- ~ DBMS Test 9
- ~ DBMS Test 10
- ~ DBMS Test 11
- ~ DBMS Test 12
- ~ DBMS Test 13
- ~ DBMS Test 14

# Backend Development with Django-8 Projects

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : DJANGO

Course link : <https://ineuron.ai/course/Backend-Development-with-Django-8-Projects>

## Course Description :-

The most widely used Python web development framework is Django. Django is a Python framework that covers all elements of web development, from handling requests and answers to creating dynamic HTML pages using templates and making database access and maintenance simple. This course has it all baked in, and it's all covered in excellent depth. Django is taught from the ground up in this course. We'll start from the beginning and work our way up, learning how to construct Django projects, execute them, and add functionality step by step.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Webpages in Django
- => Admin control
- => Sliders
- => Search feilds
- => Navbars
- => Components
- => User authentication
- => Facebook and Google authentication
- => Django message frameworks
- => Django contact forms

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Getting tools ready for Django :

- ~ Tools that we need
- ~ Your first HelloWorld project in Django
- ~ Understanding file structure

- => Project 1 - Getting basics done :

- ~ our very first Django App
- ~ Configuring new app

- => Project 2 - All about templating basics :

- ~ Project 2 and templating basics
- ~ Rendering form HTML Page
- ~ Adding an about us page
- ~ Adding contact Us page
- ~ Extending pre built templates

- => Project 3 - Interaction with Database :

- ~ Setting up command project

- ~ Creating our first model for sqlite3
- ~ Registering models to admin
- ~ Create read update and delete from database

=> Project 4 - Blog with static pages and Unique URL :

- ~ Articles app creation
- ~ ForeignKey and many to one relation
- ~ Adding articles in database
- ~ Configuring URLs
- ~ Setting up home page
- ~ Adding static files for css
- ~ Unique URL for articles

=> Project 5 - Handling forms and taking input from user :

- ~ Handling user input via forms - setup
- ~ Preparing models and admin
- ~ fixing views and other issues
- ~ Making templates look great with static files
- ~ fetching data from database
- ~ Interaction with Django forms
- ~ Taking input from user and storing it in database

=> Project 6- A CRUD Blog from user input :

- ~ Jump start of CRUD project
- ~ Reusing admin fields in web page
- ~ Update from database
- ~ Delete from database and reverse lazy urls

=> Project 7 - A TODO list with sqlite3 :

- ~ How we will take down this TODO

=> Project 7 - A TODO list with sqlite4 :

- ~ Create models for database

=> Project 7 - A TODO list with sqlite5 :

- ~ Setting up templates and static files

=> Project 7 - A TODO list with sqlite6 :

- ~ Adding a todo and decorators

=> Project 7 - A TODO list with sqlite7 :

- ~ Finishing up Todo

=> Project 8 - Login, Logout and SignUp :

- ~ Setup URL for login
- ~ Having a login View
- ~ A working Login and logout
- ~ A complete signup app



# Complete Bootstrap4 - Build 5 Projects

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : BOOTSTRAP

Course link : <https://ineuron.ai/course/Complete-Bootstrap4---Build-5-Projects>

## Course Description :-

This course will take you from having no prior knowledge of Bootstrap to mastering all of the utilities, components, widgets, and grids, as well as designing real-world themes and websites. This project oriented course does not need prior knowledge of Bootstrap .Upon successful completion of this course, you will be able to build responsive and interactive websites and beautiful static pages using the bootstrap framework. So hurry up and enroll now to start a successful career as a front-end web developer.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Bootstrap Integration and typography
- => Buttons, breakpoints and utilities
- => Team pages
- => Navbars
- => Flexboxes
- => Forms
- => Modals
- => Custom cards

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with bootstrap :

- ~ Introduction to Bootstrap4
- ~ Tools to be used in this course
- ~ File structure for learning
- ~ Emmet quick start part 1
- ~ Emmet quick start part 2

=> Bootstrap integration and typography :

- ~ Bootstrap integration
- ~ Bootstrap typography basics
- ~ Bootstrap typography for testimonials
- ~ Embed responsive youtube videos

=> Vedio Landing Page :

- ~ Getting assets and preparing html
- ~ Beautiful landing page
- ~ Customized fonts

=> Buttons Breakpoints and utilities :

- ~ Get started with bootstrap buttons
- ~ Button size and backgrounds
- ~ Border utilities in Bootstrap
- ~ Grid system basics in Bootstrap

- ~ Mobile first concept of bootstrap
- ~ Breakpoints in grid

=> Project team-page :

- ~ Getting assets and basic setup of project
- ~ Logo and display utilities
- ~ Heading section
- ~ Team person one content
- ~ Custom styling for team section
- ~ Some fix and assignments

=> Navbar, flexbox, forms and modals :

- ~ Get started with navs
- ~ Flexbox utilities
- ~ Nav panels and assignment
- ~ Basics of navbars
- ~ Toggles and colors in navbars
- ~ Forms in bootstrap
- ~ input groups in Bootstrap
- ~ Modals in bootstrap

=> Project- App launch website :

- ~ Device mockups
- ~ Getting resources
- ~ Navbar part 1
- ~ center menu of navbar
- ~ Customized navbars
- ~ Login Modal
- ~ Feature section with custom font
- ~ Background svg image
- ~ Device mockups usage
- ~ Subscription form customization
- ~ App store icons
- ~ app store CSS
- ~ Building feature section
- ~ feature column section
- ~ Customized CSS for features
- ~ fixing bugs and gradients
- ~ Just fun - unplanned video

=> Project - Build 4 Custom Cards :

- ~ Introduction to cards
- ~ Introduction to cards part 2
- ~ Downloading project 4 files
- ~ preparing HTML for Card 1
- ~ Card 1 custom CSS part 1
- ~ Card 1 custom CSS part 2 and assignment
- ~ preparing HTML for Card 2
- ~ Card 2 custom CSS
- ~ preparing HTML for Card 3
- ~ Card 3 custom CSS part 1
- ~ Custom CSS for card 3 - part 2
- ~ Custom CSS for card 3 - part 3
- ~ preparing HTML for Card 4
- ~ Custom CSS for card 4

=> Bonus sign-up page :

- ~ Download project 5 files
- ~ Preparing our HTML
- ~ CSS for background image
- ~ Purple Styling of buttons
- ~ Adding colors to buttons
- ~ Fixing custom forms
- ~ Fixing errors and media queries

# Class 10 Physics

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10-Physics>

## Course Description :-

Physics is a combination of theoretical and practical knowledge. The Physics syllabus includes laws, formulas, equations, theories and series of experiments which a student must grab. The subject leads you to the real world of imagination, the planetary motions and the universe which sounds an interesting one, isn't it? NCERT covers all the conceptual points properly but they are very summarized in the book. iNeuron allows you to explore all the content from NCERT physics in a proper manner and flow.

## Course Features :-

- => Self paced video session
- => Covered entire class 10th Physics syllabus
- => Solved questions chapter wise
- => Notes
- => Previous year solved questions

## What you will learn :-

- => Entire NCERT Class 10th Physics Syllabus
- => Chapter wise solution with detailed explanation

## Requirements :-

- => Computer with Internet Connectivity

## Instructors :-

=> Jwala Prakash :

~ I have 4+ years of experience in teaching mathematics and physics for grade 9 and 10.

I am also an experienced teacher for mathematics aptitude. I have qualified mains exam twice of the most reputed central government exam, staff selection commission(SSC)

## Curriculum details :-

=> Electricity :

- ~ Electric current and circuits
- ~ Electric potential and potential difference
- ~ Circuit diagram
- ~ Ohms law
- ~ Factors on which the resistance of conductor depends
- ~ Resistance of a system of resistors
- ~ Heating effect of electric current
- ~ Electric power

=> Magnetic effect of electric current :

- ~ Magnetic field and field lines
- ~ Magnetic field due to current carrying conductors
- ~ Force on current carrying conductor in electric field
- ~ Electric motor
- ~ Electromagnetic induction
- ~ Domestic electric circuits

=> Light - reflection and refraction :

- ~ Reflection of light
- ~ Spherical mirrors: Formation of image
- ~ Uses of spherical mirrors
- ~ Sources of Energy
- ~ Mirror formula and magnification
- ~ Refraction of light
- ~ Refraction by spherical lenses
- ~ Lens formula and magnification Power of lens

=> Human eye and colorful world :

- ~ The human eye
- ~ Power of accommodation
- ~ Defects of vision and its correction: myopia
- ~ Defects of vision and its correction: Hypermetropia
- ~ Presbyopia
- ~ Astigmatism
- ~ Refraction through prism
- ~ Dispersion of white light through glass prism
- ~ Atmospheric refraction and its effect
- ~ Scattering of light: Tyndall effect

=> Sources of energy :

- ~ *Different forms of energy*
- ~ *conventional and non-conventional sources of energy*
- ~ *Fossil fuels, solar energy; biogas; wind, water and tidal energy; Nuclear energy*
- ~ *Renewable versus non-renewable sources of Energy*

# Pro Aptitude - C++

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Aptitude---C++>

## Course Description :-

This course is designed mostly for C++ test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> C++ Theoretical Test

=> C++ Practical Test

=> C++ Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> C++ Coding Test :

~ C++ Test 1

~ C++ Test 2

~ C++ Test 3

~ C++ Test 4

# APIGEE

---

Topic Name : PROGRAMMING

Sub-topic Name : API

Course link : <https://ineuron.ai/course/APIGEE>

## Course Description :-

This course will help you to learn the fundamentals of APIGEE.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Apigee Introduction
- => Apigee Fundamentals
- => Working with APIGEE

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Apigee :
  - ~ Overview
  - ~ Apigee & API
  - ~ Apigee components
  - ~ Apigee product
  - ~ Apigee architecture
  - ~ Apigee setup
  - ~ Apigee create\_proxy

# Pro Aptitude - Java

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Pro-Aptitude---Java>

## Course Description :-

This course is designed mostly for Java test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Java Theoretical Test

=> Java Practical Test

=> Java Aptitude test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Java Test :

~ *Java Test 1*

~ *Java Test 2*

~ *Java Test 3*

~ *Java Test 4*

# Deep Learning for Kids

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Deep-Learning-for-Kids>

## Course Description :-

Learners will master the fundamentals of deep learning as well as how to tackle a challenging real-world problem that is difficult to handle with standard programming in this course. This course will teach you the fundamentals of AI, allowing you to create incredible AI applications. Students will receive hands-on practical experience in designing AI-based projects after successfully completing the course. Learners might begin applying for freelancing employment in order to make a fortune.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with tech
- => Real-time project
- => Live class recording
- => Doubt clearing
- => Assignment in all the module
- => Quiz in every module
- => Career Counselling
- => Completion certificate

## What you will learn :-

- => Introduction Artificial Intelligence
- => Introduction to Deep Learning
- => Supervised learning
- => Unsupervised learning
- => Python basics
- => NumPy basics
- => Pandas basics
- => TensorFlow
- => Kera's
- => Artificial neural network
- => Convolution neural network
- => Projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Course Introduction :
  - ~ Welcome to machine learning course
  - ~ What you will learn from this course
  - ~ Course pre-requisites
  - ~ What is deep learning?
  - ~ Who is this course for?
  - ~ What you will get from this course?
  - ~ How to get access to course materials?
  - ~ What career path you can follow after completion of this course?
- => Introduction to AI :
  - ~ What is Artificial intelligence?
  - ~ History of AI
  - ~ Applications of AI
  - ~ Advantage of AI
  - ~ Practical use of AI
- => Introduction to learning :
  - ~ What do you mean by learning?



- ~ Why deep learning?
- ~ How babies learn: An Analogy
- ~ Different types of learning
- ~ What is Supervised learning?
- ~ Supervised learning example: Importance of Teacher feedback
- ~ What is Unsupervised learning?
- ~ Unsupervised learning example: Categorizing students based on hobbies (Annual Function)
- ~ What is Reinforcement learning?
- ~ How a self-driving car works: An Analogy
- ~ Discussion: Sofia robot
- ~ Uses of Deep learning

#### => Assignment1 :

- ~ Give 3 examples of AI used in the education sector.

#### => Preparing your system :

- ~ Why python?
- ~ Colab overview

#### => Working with important libraries :

- ~ Python basics
- ~ Numpy basics
- ~ Pandas basics
- ~ Tensorflow basics
- ~ Keras basics

#### => Assignment2 :

- ~ Create a function to add 2 numbers
- ~ Create a function that will take name and address from user and print the output
- ~ Print multiplication table of 1 to 10 using for loop
- ~ Print multiplication table of 1 to 10 using for loop
- ~ Using TensorFlow add two numbers and print the output
- ~ Using TensorFlow add two multiply to matrices and print the output

#### => Neural network basics :

- ~ What is neuron?
- ~ Neural network vs Human brain network
- ~ What is perceptron?
- ~ What is ANN?
- ~ Practical: Perceptron
- ~ Tensorboard overview
- ~ Logging the activity of training using Tensorboard
- ~ Analysis: How to classify orange and apple with features
- ~ Practical: Predicting the price of premium phones for the year 2023 using Neural network
- ~ Explain Logistic Regression
- ~ Practical: Classifying male and female based on height and weight of a person
- ~ What do you mean by Activation function?

#### => Assignment3 :

- ~ Create a neural network and predict the price of mobile network recharge for next 1 year

#### => Convolution neural networks :

- ~ Introduction
- ~ What are images?
- ~ Image data vs numerical data
- ~ Practical: Deep neural network
- ~ Using Netron to visualize neural network
- ~ What is CNN?
- ~ Why use CNN instead of N-layer neural network?
- ~ Visualizing different layers of CNN using web app: <https://blog.terencebroad.com/archive/convnetvis/vis.html>
- ~ Practical: Basic CNN using keras
- ~ Practical: Create a CNN and identify day, evening and night
- ~ Discussion: Use cases of CNN (Detection, tracking)

#### => Assignment4 :

- ~ Create a CNN to classify whether the room is empty or not

#### => Projects :

- ~ Classifying apple vs orange
- ~ Fruit classification using CNN

#### => Summary :

- ~ Course Outro
- ~ Future Scope of Deep learning

# SQL Foundations

---

Topic Name : DATA ANALYTICS

Sub-topic Name : SQL

Course link : <https://ineuron.ai/course/SQL-Foundations>

## Course Description :-

Data practitioners must master SQL since it is the most essential query language you can learn. Many prominent relational database management systems such as MySQL employ it. However, data analysis and big data frameworks and tools such as Apache Spark also utilise it. As a result, learning MySQL offers up a plethora of prospects and occupations - whether you want to work with relational databases or become a data scientist, knowing Mysql is essential. Even if you have no previous experience of MySQL, this practical course will build the groundwork for SQL and structured database querying.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Basic Concepts of Advantages of DBMS.
- => Exploring Relational DBMS
- => E-R Modeling and Diagram
- => Normalization
- => Introduction to SQL
- => DDL and DML Statements
- => Working with Queries (DQL)
- => Aggregate Functions
- => Joins and Set Operations
- => Implementation of Data integrity
- => Working with Constraints
- => Implementing Views

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Introduction to Basic Database Concepts :
  - ~ What is Data, Field, Record and database?
  - ~ Limitations of File Management System.
  - ~ Basic Concepts of Advantages of DBMS.
  - ~ Exploring Relational DBMS
  - ~ Understanding Client and Server
- => E-R Modeling and Diagram :
  - ~ Analyzing the Requirement
  - ~ Identify Entities and their Relationships
  - ~ Drawing E-R Diagram
  - ~ Conversion of E.R. Diagrams into Tables
- => Normalization :
  - ~ First Normal Form
  - ~ Second Normal Form

~ *Third Normal Form Practically Normalizing Tables*

=> Introduction to SQL Server :

- ~ *What is SQL Server Version history and different editions*
- ~ *Installation of SQL in Windows*
- ~ *Basic Features Components and Tools*
- ~ *Starting and Stopping SQL Server Instances / Services*
- ~ *Introduction to Management Studio*
- ~ *Types of System Databases in SQL*

=> Introduction to SQL :

- ~ *Basics of SQL Types of SQL Statements*
- ~ *DDL, DML, DQL, DCL and TCL*
- ~ *Create Database using Management Studio*
- ~ *Datatypes in SQL Server*
- ~ *Exploring DDL Statements on Table using Management Studio*
- ~ *Why write statements in Frontends?*

=> DDL and DML Statements :

- ~ *Create, Alter and Drop Table Insert,*
- ~ *Update and Delete Statement Truncate Statement*

=> Working with Queries (DQL) :

- ~ *Understanding Select Statement*
- ~ *Usage of Top, Distinct, Null etc...keywords*
- ~ *Using String and Arithmetic Expressions*
- ~ *Exploring Where Clause with Operators*
- ~ *Using Advanced Operators*
- ~ *Sorting data using Order By clause*
- ~ *Working with basic of Sub Queries*

=> Aggregate Functions :

- ~ *Using functions in Queries*
- ~ *Count, Sum, Min, Max, Avg Group By and Having Clause*
- ~ *Using Group By with Rollup and Cube*

=> Joins and Set Operations :

- ~ *Introduction to Joins Cross Joins*
- ~ *Inner Join*
- ~ *Outer Join*
- ~ *Self Join*
- ~ *Co-related Sub Queries*
- ~ *Set Operations using Unions, Intersect and Except*

=> Implementation of Data integrity :

- ~ *Entity integrity*
- ~ *Domain integrity*
- ~ *Referential integrity*
- ~ *Types of constraints*

=> Working with Constraints :

- ~ *Unique*
- ~ *Not NULL*
- ~ *Primary Key*
- ~ *Default Check Foreign Key*

=> Implementing Views :

- ~ *Introduction & Advantages of Views*
- ~ *Creating, Altering, Dropping Views*
- ~ *Advance Options while Creating a View*
- ~ *SQL Server Catalogue Views*

=> Data Control language (DCL) :

- ~ *Creating Users & Roles*
- ~ *Granting & Revoking of Roles & privileges*
- ~ *Managing using Management Studio*

=> Working with Indexes :

- ~ *Introduction Clustered and Non Clustered Index*
- ~ *Creating and Dropping Indexes*

=> Writing Transact-SQL (T-SQL) :

- ~ *What is T-SQL?*
- ~ *Scripts and Batches Declaring Variables*
- ~ *Using Statements*
- ~ *Working with Temp tables*
- ~ *Error Handling*
- ~ *Using System Functions / Global Variables Using Dynamic SQL*

=> Working with Stored Procedures and Functions :

- ~ *Introduction to stored procedures*
- ~ *Benefits of Stored Procedures*
- ~ *Creating, Executing Modifying, Dropping*
- ~ *InputOutput and Optional Parameters*
- ~ *System defined SPs and Functions.*
- ~ *User defined Functions*

=> Implementing Triggers :

- ~ *Introduction to triggers*
- ~ *Constraints vs Triggers*
- ~ *Creating, Altering, Dropping triggers*
- ~ *for/after/instead of triggers*
- ~ *Using Rollback Tran*

## => Working with Cursors :

- ~ *Creating Cursors*
- ~ *Cursors vs. Select*
- ~ *Types of cursors*
- ~ *Locks on cursors*
- ~ *Advantages of cursors*

## => Transaction Control Language (TCL) :

- ~ *Introduction Transactions process*
- ~ *Types of transactions (Implicit, explicit)*
- ~ *Working with Locks, Types of locks*

## => Backup and Restore :

- ~ *Generating SQL Script*
- ~ *Executing SQL Script*
- ~ *Generating Change Script*
- ~ *Taking database Backup*
- ~ *Restoring database using backup*
- ~ *Attaching and Detaching of database*

## => Interview Questions for Beginners :

- ~ *What is SQL?*
- ~ *How to create a table in SQL?*
- ~ *How to delete a table in SQL?*
- ~ *How to change a table name in SQL?*
- ~ *How to delete a row in SQL?*
- ~ *How to create a database in SQL?*
- ~ *What is Normalization in SQL?*
- ~ *What is join in SQL?*
- ~ *What is SQL Server ?*
- ~ *How to insert date in SQL?*
- ~ *What is Primary Key in SQL?*
- ~ *How do I view tables in SQL ?*
- ~ *What is PL/SQL*
- ~ *What is MYSQL?*
- ~ *How can see all tables in SQL ?*
- ~ *What is ETL in SQL?*
- ~ *How to install SQL?*
- ~ *What is the Update Command in SQL ?*
- ~ *How to rename column name in SQL server ?*
- ~ *What are the types of SQL Queries?*
- ~ *Write a Query to display the number of employees working in each region ?*
- ~ *What are nested Triggers ?*
- ~ *Write a Query the employee names having a salary greater than 20000 or equal to or less than 10000*
- ~ *Given a table Employee having columns empName and empid, what will be the result of the SQL query below? Select empName from Employee order by 2 asc;*
- ~ *What is OLTP?*
- ~ *What is Data Integrity?*
- ~ *What is OLAP?*
- ~ *Find the constraints information from the table*
- ~ *Can you get the list of employees with same salary ?*
- ~ *What is an alternative for TOP clause in SQL*
- ~ *Will following statement give error or 0 as output ? SELECT AVG (NULL)*
- ~ *What is cartesian product of the Table ?*
- ~ *What is a schema in SQL?*
- ~ *What is the where clause in SQL ?*
- ~ *How to delete a column in SQL?*
- ~ *What is a unique key in SQL?*
- ~ *How to Implement multiple conditions using WHERE Clause ?*

## => Intermediate Level Questions :

- ~ *What is SQL Injection ?*
- ~ *What is a Trigger in SQL ?*
- ~ *How to insert multiple rows in SQL ?*
- ~ *How to find the nth Highest salary in SQL ?*
- ~ *How to Copy table in SQL ?*
- ~ *How to add a new column in SQL ?*
- ~ *How to use LIKE in SQL ?*
- ~ *If we drop a table, does it also drops related objects like constraints, indexes, columns, default, views and sorted procedures ?*
- ~ *Can we disable trigger? If yes, How ?*
- ~ *What is a Live Lock ?*
- ~ *How to fetch alternate records from a table ?*
- ~ *Define COMMIT and give an example ?*
- ~ *Can you join table by itself ?*
- ~ *Example Equi join with example ?*
- ~ *How do we avoid getting duplicate entries in a query ?*
- ~ *How can you create an empty table from an existing table ?*
- ~ *Write a query to display odd records from student table ?*
- ~ *Explain Non Equi join with example ?*
- ~ *How can you delete duplicate records in a table with no primary key ?*
- ~ *Difference between NVL and NVL2 functions ?*
- ~ *What is the difference between clustered and non-clustered indexes ?*
- ~ *What does this query says ? GRANT privilege\_name ON object\_name TO {user\_name|PUBLIC |role\_name } [WITH GRANT OPTION];*

## => Resume Discussion :

- ~ *Resume Discussion*

# Numerical Computation with Python

---

Topic Name : K12

Sub-topic Name : CLASS9

Course link : <https://ineuron.ai/course/Numerical-Computation-with-Python>

## Course Description :-

In this course, students will learn the principles of numerical analysis and computation in Python and use the power of a powerful Python library called NumPy. In the world of Artificial Intelligence and Data Science, NumPy is one of the most extensively used Python packages. NumPy provides high-performance functions and data analysis tools that are very efficient and simple to use.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to NumPy
- => Applications of NumPy
- => NumPy fundamentals
- => Linear algebra using NumPy
- => Generating random number using NumPy

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shivan Kumar :

~ Associate Data Scientist, Mentor, and Kaggle Master with 2 Years of Experience. Experience in building models that translate data points into business insights. Highly accurate at collecting, analyzing, and interpreting large datasets, developing machine learning, deep learning, Chatbots models, and deploying the solutions on the cloud. I also love to participate in Hackathons. In my free time, I like to mentor students to learn Data Science and achieve their goals.

## Curriculum details :-

=> Course Introduction :

- ~ Dashboard Overview
- ~ Course Overview
- ~ Who is this course for?
- ~ What is Data Science?
- ~ Why should you learn Data Science?
- ~ History of Data Science
- ~ What is NumPy?
- ~ Why should you learn NumPy?
- ~ Applications of NumPy

=> Introduction to Colab and Python programming :

- ~ Getting started with Google Colab
- ~ Saving and loading Jupyter Notebooks

=> Introduction to Numpy :

- ~ What is an array?
- ~ Understanding array data structure
- ~ Creating a 1-Dimensional array using NumPy
- ~ 1-Dimensional array operations
- ~ Searching an element in 1-Dimensional array

=> Assignment 1 :

- ~ Write a Program to find first occurrence of an element in a 1-Dimensional array

=> NumPy fundamentals :

- ~ *What are multi-dimensional arrays?*
- ~ *Why should we learn multi-dimensional arrays?*
- ~ *Creating a 2-Dimensional array*
- ~ *Creating a 3-Dimensional array*
- ~ *Data Types for ndarray*
- ~ *Arithmetic operations with NumPy arrays*
- ~ *Basic indexing and slicing*
- ~ *How to do array indexing with slices?*
- ~ *Boolean Indexing in NumPy arrays*
- ~ *Transposing arrays in NumPy*
- ~ *Swapping axes in Numpy*

=> Assignment 2 :

- ~ *Building a calculator to perform arithmetic operations on two 1-Dimensional arrays*

=> Linear Algebra in NumPy :

- ~ *Return the diagonal elements of a square matrix*
- ~ *Matrix multiplication*
- ~ *Compute the sum of the diagonal elements*
- ~ *Compute the matrix determinant*
- ~ *Eigenvalues and eigenvectors*
- ~ *Inverse of a square matrix*

=> Assignment 3 :

- ~ *Program to compare elements of two NumPy arrays*

=> Assignment 4 :

- ~ *Program to calculate the sum of the diagonal elements of a NumPy array*

=> Random Number Generation in Numpy :

- ~ *Random number generator*
- ~ *Generating uniform distribution*
- ~ *Generating random integers*
- ~ *Normal distribution*
- ~ *Standard normal distribution*
- ~ *Binomial distribution*

=> Assignment 5 :

- ~ *Program to generate random numbers from the uniform distribution using NumPy*

=> Assignment 6 :

- ~ *Program to return a Matrix of random values from a standard normal distribution*

=> Course Summary :

- ~ *Course Outro*
- ~ *Future Scope and References*

# OpenCV Job Preparation

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION INTERVIEW

Course link : <https://ineuron.ai/course/OpenCV-Job-Preparation>

## Course Description :-

OpenCV provides a real-time optimized Computer Vision library, tools, and hardware. It also supports model execution for Machine Learning (ML) and computer vision. The goal here is to make you completely ready with OpenCV along with various interview-based questions and a complete detailed roadmap.

## Course Features :-

- => Roadmap
- => Challenges
- => Interview questions
- => Resume preparation
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Various practical questions
- => Visual computing questions
- => Resume discussion
- => Working with vision projects
- => Image manipulation
- => Video manipulation

## Requirements :-

- => Prior understanding in OpenCV
- => Knowledge in Python programming
- => A system with a decent internet connection
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Interview questions :
  - ~ Explain what OpenCV is? Preview
  - ~ What are Erosion and Dilation in OpenCV? Preview
  - ~ Which method of OpenCV is used to save the image and show the image?
  - ~ What is the use of Sobel operation in OpenCV?
  - ~ Enlist different types of filters available in OpenCV?
  - ~ Which function is used to draw a line in OpenCV?
  - ~ How to connect GPU with OpenCV?
  - ~ What is computer vision? Enlist a few applications?
  - ~ What is Haarcascade?
  - ~ Advantages of OpenCV
  - ~ Disadvantages of OpenCV

# C language for Absolute Beginners

---

Topic Name : PROGRAMMING

Sub-topic Name : C

Course link : <https://ineuron.ai/course/C-language-for-Absolute-Beginners>

## Course Description :-

This course is designed mostly for novice programmers who may not have any prior programming language knowledge. From the most fundamental to the most sophisticated subjects, there is something for everyone. Step by step, from a simple to a sophisticated programme. This course should be taken if one want to pursue a career as a programmer.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to Programming
- => How to develop a software using C Language?
- => Constants
- => Variables
- => Keywords
- => Program to print ASCII code of a given character
- => Unary Operators
- => Arithmetic Operators
- => Bitwise Operators
- => Relational Operators
- => Logical Operators
- => if
- => If else
- => Conditional operator
- => Nested if else
- => If else ladder
- => Practice Programs on decision control

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Saurabh Shukla :

~ Saurabh Shukla has been educating children with this credo, and he does so for free. MySirG.com, the educator's YouTube channel, features video lessons on programming languages. Saurabh has successfully reached thousands of students around the country by making it his aim to produce subject-related videos on a daily basis.

## Curriculum details :-

=> Day1 :

- ~ Introduction to Programming
- ~ How to develop a software using C Language?
- ~ Setup Environment for C Programming

=> Day2 :

- ~ History of C Language
- ~ Develop first C Program
- ~ Formal Beginning of Learning C Language



=> Day3 :

- ~ Tokens
- ~ Constants
- ~ Variables
- ~ Keywords

=> Day4 :

- ~ Data Types
- ~ Declaring Variables
- ~ Garbage Value
- ~ float vs double
- ~ ASCII codes

=> Day5 :

- ~ Output text on monitor using printf()
- ~ Escape Sequences
- ~ Printing value of a variable and expression
- ~ Format Specifiers

=> Day6 :

- ~ Taking input from keyboard using scanf()
- ~ Program to calculate sum of two numbers
- ~ Program to calculate area of a circle
- ~ Program to print ASCII code of a given character

=> Day7 :

- ~ Unary Operators
- ~ Arithmetic Operators
- ~ Bitwise Operators
- ~ Relational Operators
- ~ Logical Operators
- ~ Assignment Operators

=> Day8 :

- ~ Practice Programs on Operators
- ~ Doubt Handling Session

=> Day9 :

- ~ Decision Control Instruction
- ~ if
- ~ If else
- ~ Conditional operator
- ~ Nested if else
- ~ If else ladder

=> Day10 :

- ~ Practice Programs on decision control
- ~ Doubt Handling Session

=> Day11 :

- ~ Iterative Control Instruction
- ~ While loop
- ~ Practice Programs

=> Day12 :

- ~ Do while
- ~ For loop
- ~ Practice Programs

=> Day13 :

- ~ Use of break keyword in loop
- ~ Practice Programs
- ~ Doubt Handling Session

# AIOps

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : <https://ineuron.ai/course/AIOps>

## Course Description :-

These days the most in-demand technical expertise is Artificial Intelligence Operations (AIOps). It aids in the application of DevOps principles to AI product development. This course will cover multiple ways to implement AIOps methodology in ML and DL projects, including implementation on various clouds such as AWS, Azure, GCP, and DigitalOcean. Researchers from diverse disciplines widely use it to estimate and analyze the result.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable Resources

## What you will learn :-

- => AIOps/MLOps
- => Linux, Git foundation
- => Docker
- => Kubernetes

## Requirements :-

- => Minimum system requirements: Intel Core i3 processor and 4GB RAM or higher.
- => A system with a decent internet connection

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> AIOps introduction theory :

- ~ Introduction 1
- ~ Introduction 2
- ~ Introduction 3
- ~ Challenges
- ~ AIML generic steps
- ~ Level 0 workflow
- ~ Level 0 characteristics and observations
- ~ Level 1 workflow
- ~ Level 1 aim
- ~ Level 1 characteristics
- ~ Frequently used terms
- ~ Data validation
- ~ Model validation - Offline
- ~ Model validation - Online
- ~ Feature store
- ~ Metadata storage
- ~ Pipeline trigger
- ~ Final summary
- ~ Level 2 aim
- ~ Level 2 CI CD workflow detail discussion part 1
- ~ Level 2 CI CD workflow detail discussion part 2
- ~ Level 2 more on CI
- ~ Level 2 more on CD
- ~ Level 2 deployment types
- ~ Level 2 summary final

=> Linux introduction :

- ~ Introduction to Linux
- ~ What is Linux
- ~ Important pieces in Linux
- ~ Features of Linux

- ~ Evolution of Linux
- ~ Differences between Windows and Linux

#### => Setting up our Linux space :

- ~ Downloading necessary tools
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting up passwordless SSH

#### => Linux concepts :

- ~ What is Kernel
- ~ Types of Kernel
- ~ What is Shell
- ~ Types of Shell
- ~ Distro in Linux
- ~ Linux boot process
- ~ File system
- ~ Run levels in Linux
- ~ File types of Linux

#### => Package management :

- ~ Package management
- ~ Package managers & DPKG
- ~ Working with APT & APT GET
- ~ Apt-get advanced part - 1
- ~ Apt-get advanced part - 2

#### => Linux commands :

- ~ Linux commands part - 1
- ~ Linux commands part - 2
- ~ Linux commands part - 3
- ~ Linux commands part - 4
- ~ Cat command usages

#### => Working with terminal :

- ~ File archival
- ~ File compression
- ~ Files and patterns search
- ~ Input output redirection
- ~ Working with VI editor
- ~ Advanced VI editor part - 1
- ~ Advanced VI editor part - 2

#### => Permissions & security :

- ~ Types of account in Linux
- ~ User management
- ~ Group management
- ~ Files access controls
- ~ Linux file permissions
- ~ Modifying file ownership
- ~ Sudoers in Linux
- ~ Cronjobs
- ~ SCP
- ~ Special permissions
- ~ System management
- ~ System tools
- ~ Hard link and Soft link
- ~ Aliasing in Linux
- ~ Creating users in multiple ways

#### => Linux in AWS cloud deploy an app in EC2 :

- ~ Launching an Ubuntu VM and SSH setup
- ~ Package installation in VM
- ~ Running our calculator app
- ~ Unicorn & Nginx setup
- ~ Creating a Unicorn service
- ~ Attaching an Elastic IP
- ~ Attaching OpenSSL certificates for https

#### => GitHub introduction :

- ~ Git introduction
- ~ What is version control?
- ~ Types of version control
- ~ What is git?
- ~ Why git?
- ~ Git installation on Windows
- ~ Git installation on Linux
- ~ Git setup
- ~ Git terminologies
- ~ Repositories in GIT
- ~ Creating repository
- ~ Checking repository history
- ~ Doing commits
- ~ Git diff
- ~ Git restore
- ~ Gitignore
- ~ Tagging
- ~ Branching

- ~ Branching practicals
- ~ Merging
- ~ Merge conflicts
- ~ Remote repository
- ~ Cloning repository
- ~ Working with remote repository
- ~ Pushing to remote failed in GitHub
- ~ Personal access token setup in Windows
- ~ Personal access token setup in Linux
- ~ Pull request
- ~ Git fetch & pull
- ~ Fork
- ~ Rebasing
- ~ Interactive rebasing
- ~ Git rewrite history
- ~ Git rewrite history continued
- ~ Cherry picking
- ~ Modify recent commits
- ~ Git revert
- ~ Git checkout
- ~ Git reset
- ~ Git stash
- ~ Git reflog
- ~ Course outro

=> DVC introduction :

- ~ What is DVC?
- ~ Installation

=> Automate ML pipelines with DVC :

- ~ Workflow
- ~ Basic setup
- ~ Stage 01 implementation
- ~ Stage 01 added to dvc.yaml
- ~ Stage 02 implementation
- ~ Stage 02 added to dvc.yaml
- ~ Stage 03 implementation
- ~ Stage 03 added to dvc.yaml
- ~ Stage 04 implementation
- ~ Stage 04 added to dvc.yaml
- ~ Final update

=> Getting started with DVC :

- ~ Data versioning 01
- ~ Data versioning 02
- ~ Data versioning 03

=> Automate DL pipelines with DVC (dl-tensorflow) :

- ~ Workflow description
- ~ Creation of project skeleton
- ~ Stage 01 implementation
- ~ Stage 01 added to dvc.yaml
- ~ Stage 01 final update
- ~ Stage 02 main file creation
- ~ Stage 02 base model creation
- ~ Stage 02 python scripting
- ~ Stage 02 logging model summary in the logs
- ~ Stage 02 added to the dvc.yaml file
- ~ Stage 03 preparing directory creation for callbacks
- ~ Stage 03 adding callback utility
- ~ Stage 03 adding to dvc.yaml
- ~ Stage 04 loading binary file of callbacks
- ~ Stage 04 load untrained model and start training
- ~ Stage 04 training valid generator
- ~ Stage 04 model training added
- ~ Stage 04 save model

=> Docker :

- ~ Docker introduction
- ~ What is Docker?
- ~ Why Docker?
- ~ Benefits of Docker
- ~ What is container?
- ~ Containers vs VM
- ~ Containers vs image
- ~ Docker editions
- ~ What docker is not?
- ~ Important terminologies
- ~ Docker setup in Windows
- ~ Docker setup in Linux
- ~ Docker setup in Mac

=> Docker basic usage :

- ~ Docker basic commands part 1
- ~ Docker basic commands part 2

=> Docker run :

- ~ Docker run part 1
- ~ Docker run part 2

=> Docker images :

- ~ Docker images
- ~ Creating a new image
- ~ Environment variables
- ~ Commands & entrypoints

#### => Docker compose :

- ~ Docker compose
- ~ Voting application understanding
- ~ Docker compose versions
- ~ Docker compose networks
- ~ Voting application with docker run
- ~ Voting application with docker compose

#### => Docker concepts :

- ~ Docker engine
- ~ Docker storage
- ~ Docker networking
- ~ Docker registry

#### => Kubernetes :

- ~ Course introduction
- ~ What is Kubernetes?
- ~ Why Kubernetes?
- ~ Containers
- ~ Containers orchestration

#### => Kubernetes setup :

- ~ Kubernetes setup on Windows
- ~ Kubernetes setup in Linux
- ~ Kubernetes setup in Mac
- ~ Minikube
- ~ Kubeadm
- ~ Kubernetes architecture

#### => Kubernetes concepts :

- ~ Pods
- ~ Node architecture
- ~ Replication controller
- ~ Deployments
- ~ Services
- ~ Labels
- ~ Healthchecks
- ~ Readiness probe
- ~ Pod state
- ~ Pod lifecycle
- ~ Secrets
- ~ Webui

#### => Services :

- ~ Nodeport
- ~ Clusterip
- ~ Loadbalancer

#### => Advanced :

- ~ Service discovery
- ~ ConfigMap
- ~ Ingress controller
- ~ External DNS
- ~ Volumes
- ~ Volumes autoprovisioning
- ~ Pod presets
- ~ Statefulsets
- ~ Daemonsets
- ~ Resource usage monitoring
- ~ Autoscaling

#### => Deploying apps :

- ~ Microservices architecture
- ~ Deploying in Kubernetes
- ~ Deploying in kubernetes with deployments

#### => Packaging & deployment :

- ~ Introduction to Helm
- ~ Creating your own Helm charts
- ~ Setting up a Helm repository on S3
- ~ Building and deploying Helm charts with Jenkins

#### => MLFlow introduction :

- ~ What is MLFlow?
- ~ Installation
- ~ Where runs are recorded
- ~ How runs and artifacts are recorded
- ~ Scenario 1: MLFlow on localhost
- ~ Scenario 2: MLFlow on localhost with sqlite
- ~ Scenario 3: MLFlow on localhost with tracking server
- ~ Scenario 4: MLFlow with remote tracking server, backend and artifact stores
- ~ Logging data to runs
- ~ Logging functions
- ~ Launching multiple runs in one program
- ~ Performance tracking with metrics
- ~ Visualizing metrics

=> Automatic logging :

- ~ *Scikit-Learn*
- ~ *Tensorflow and Keras*
- ~ *Gluon*
- ~ *Xgboost*
- ~ *Pytorch*

=> MLFlow tracker :

- ~ *Organizing runs in experiments*
- ~ *Managing experiments and runs with the tracking service API*
- ~ *Tracking UI*
- ~ *Querying runs programmatically*
- ~ *MLFlow tracking servers*
- ~ *Storage*
- ~ *Networking*
- ~ *Logging to a tracking server*

=> MLFlow projects :

- ~ *Overview*
- ~ *Specifying projects*
- ~ *Running projects*
- ~ *Iterating quickly*
- ~ *Building multistep workflows*

=> MLFlow models :

- ~ *Storage format*
- ~ *Model signature and input example*
- ~ *Model API*
- ~ *Built-In model flavors*
- ~ *Model customization*
- ~ *Built-In deployment tools*
- ~ *Deployment to custom targets*

=> Model registry :

- ~ *Model registry workflows*
- ~ *UI workflow*
- ~ *Registering a model*
- ~ *Using the model registry*
- ~ *API workflow*
- ~ *Adding an MLFlow model to the model registry*
- ~ *Fetching an MLFlow model from the model registry*
- ~ *Serving an MLFlow model from model registry*
- ~ *Adding or updating an MLFlow model descriptions*
- ~ *Renaming an MLFlow model*
- ~ *Transitioning an MLFlow models stage*
- ~ *Listing and searching MLFlow models*
- ~ *Archiving an MLFlow model*
- ~ *Deleting mlflow models*

=> Mlflow integration with project :

- ~ *Mlflow integration with project*

=> Kubeflow foundation :

- ~ *What is Kubeflow?*
- ~ *Core Kubeflow components*
- ~ *How to set up Kubeflow on Kubernetes*
- ~ *How to develop basic ML models in Kubeflow notebooks*
- ~ *How to train and deploy models in kubeflow*
- ~ *How to use Kubeflow pipelines*
- ~ *How to use kfserving to deploy models*
- ~ *How to manage logs with Kubeflow metadata component*
- ~ *Katib hyper parameter tuning*
- ~ *Kubeflow pipelines to kfserving*

=> AWS MLOps :

- ~ *Amazon sagemaker*
- ~ *Amazon s3*
- ~ *AWS codebuild*
- ~ *AWS codecommit*
- ~ *Sagemaker training job*
- ~ *Sage maker endpoint*
- ~ *Amazon api gateway*
- ~ *Sagemake model monitoring*
- ~ *Cloudwatch synthetics*
- ~ *Cloudwatch alarm*

=> Azure MLOps :

- ~ *Create an Azure machine learning workspace*
- ~ *Setup a new project in Azure DevOps*
- ~ *Import existing YAML pipeline to Azure DevOps*
- ~ *Declare variables for CI/CD pipeline*
- ~ *Create training compute*
- ~ *Train ML model*
- ~ *Register model*
- ~ *Deploy model in AKS*

=> GCP MLOps :

- ~ *Creating Flask application using Python*
- ~ *Best practices building Flask app*
- ~ *Understanding docker files and dependencies*
- ~ *Creating container image*

- ~ Walkthrough of different deployment options
- ~ Serverless deep dive
- ~ Deploying on GCP app engine
- ~ Deploying on serverless framework
- ~ Hosted Kubeflow pipelines
- ~ Start hosted pipelines
- ~ Cluster permissions
- ~ Development environment
- ~ Launch AI platform notebook
- ~ CI/CD production environment
- ~ Set up continuous integration (CI)
- ~ Verify CD

=> Digital ocean :

- ~ Droplets
- ~ File transfers
- ~ Gitops
- ~ Jenkins
- ~ Creating jobs
- ~ Creating pipelines in jenkins
- ~ Docker images
- ~ Kubernetes flow
- ~ Creating clusters
- ~ Load testing

# Enterprise Java with Spring Boot Tech Neuron

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Enterprise-Java-with-Spring-Boot-Tech-Neuron>

## Course Description :-

Java is one of the most widely used programming languages, owing to its versatility and compatibility. Java can be used for a variety of purposes, including software development, mobile application development, and large-scale system development. This Java course will teach you all you need to know to get started with Java.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Java Basics
- => Loops
- => OOP
- => Java Projects
- => Exception Handling
- => Blockchain Project
- => MultiThreading
- => Collection Framework
- => Junit
- => MySQL
- => NoSQL
- => JDBC
- => Hibernate

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Welcome to the Course :
  - ~ *Course Introduction*
- => Java Basics :
  - ~ *Introduction to Java*
  - ~ *Setup*
  - ~ *Getting Started*
  - ~ *How Java Works*
  - ~ *Variables*
  - ~ *Data Types*
  - ~ *Naming Conventions*
  - ~ *Operators*
  - ~ *Conditional Statements*
  - ~ *If Else*
  - ~ *Ternary*
  - ~ *Switch*
- => Loops :
  - ~ *While*
  - ~ *For*
  - ~ *Nested Loops*
  - ~ *Break and Continue*
- => OOP :



- ~ *Introduction to Object Oriented Programming*
- ~ *Class and Object*
- ~ *Constructor*
- ~ *This Keyword*
- ~ *Method and Constructor Overloading*
- ~ *Static Keyword*
- ~ *Inner Class*
- ~ *Arrays*
- ~ *Enhanced for loop*
- ~ *Varargs*
- ~ *Inheritance*
- ~ *Super Method*
- ~ *Multiple Inheritance*
- ~ *Method Overriding*
- ~ *Super Keyword*
- ~ *Multiple Inheritance*
- ~ *Encapsulation*
- ~ *Wrapper class | Autoboxing*
- ~ *Abstract Keyword*
- ~ *Final Keyword*
- ~ *Interface*
- ~ *Anonymous Inner class*
- ~ *Functional Interface*
- ~ *Lambda Expression*
- ~ *Package*
- ~ *Access Modifier*

#### => Java Projects :

- ~ *Number Guessing Game*
- ~ *Java Object to JSON Convertor*

#### => Exception Handling :

- ~ *Exceptions*
- ~ *Try Catch*
- ~ *Finally*
- ~ *Multiple Catch Blocks*
- ~ *Checked Exceptions*
- ~ *User Defined Exception*

#### => Blockchain Project :

- ~ *Building Blockchain using Java*

#### => Multithreading :

- ~ *Introduction*
- ~ *Thread Class*
- ~ *Runnable*
- ~ *Lambda Expression*
- ~ *Thread Priority*
- ~ *Synchronized*

#### => Project :

- ~ *Chatting Application*

#### => Collection Framework :

- ~ *Introduction*
- ~ *Iterator and List Interface*
- ~ *Set*
- ~ *Map*
- ~ *Comparator, Comparable*
- ~ *Generics*

#### => Projects :

- ~ *Online Voting System*
- ~ *Work Scheduler*

#### => Junit :

- ~ *Introduction*
- ~ *Unit Testing*
- ~ *Test Exceptions*
- ~ *Multiple Assertions*
- ~ *Annotations*

#### => Junit Project :

- ~ *Test Cases for Online Voting System*

#### => MySQL :

- ~ *Introduction to SQL*
- ~ *Setup*
- ~ *What is Database*
- ~ *Creating, Dropping Databases*
- ~ *Introduction to Tables*
- ~ *Creating, Dropping, Altering the Tables*
- ~ *CRUD*
- ~ *Select, Insert, Update, delete Queries*
- ~ *Where, groupby, having*
- ~ *Aggregate Functions*
- ~ *One to many, many to one, many to many relationship*

#### => Sql with Java Projects :

- ~ *Indexing for Blockchain Project*
- ~ *Online Course Management*

=> NoSQL :

- ~ *Introduction to NoSQL*
- ~ *Categories*
- ~ *NoSQL vs RDBMS*
- ~ *Couch DB*
- ~ *Mongo DB*
- ~ *Cassandra*
- ~ *Redis*

=> NoSql with Java Projects :

- ~ *IPL Stats*

=> JDBC :

- ~ *Introduction*
- ~ *CRUD Operations*
- ~ *ResultSet*
- ~ *Connection Pooling*

=> JDBC Project :

- ~ *Book My Calendar*

=> Hibernate :

- ~ *Introduction to Hibernate*
- ~ *Setup*
- ~ *Configuration File*
- ~ *SQL Property*
- ~ *Annotation*
- ~ *CRUD*
- ~ *Embeddable Object*
- ~ *Mapping Relations*
- ~ *EAGER LAZY*
- ~ *Caching*
- ~ *HQL*
- ~ *Object States Persistence Life Cycle*
- ~ *Get vs Load*
- ~ *JPA*

=> Hibernate Project :

- ~ *Hall of Fame*

=> Servlets and JSP :

- ~ *Introduction*
- ~ *Setup*
- ~ *Creating First project*
- ~ *Creating Servlet and XML*
- ~ *Get vs Post*
- ~ *RequestDispatcher*
- ~ *HttpServletRequest and HttpServletResponse*
- ~ *RequestDispatcher and sendRedirect*
- ~ *HttpSession and Cookie*
- ~ *ServletConfig and ServletContext*
- ~ *Servlet Annotation Configuration*
- ~ *JSP*
- ~ *JSP to Servlet*
- ~ *Tags, Scriptlet, Declaration, Directive, Expression*
- ~ *Implicit Objects*
- ~ *Exception handling in JSP*
- ~ *JDBC in JSP*
- ~ *Servlet Filters*

=> JSP Project :

- ~ *Stock Broker*

=> Spring :

- ~ *Introduction*
- ~ *Dependency Injection*
- ~ *BeanFactory*
- ~ *ApplicationContext*
- ~ *Spring Container*
- ~ *Singleton vs Prototype*
- ~ *Setter Injection*
- ~ *Ref Attribute*
- ~ *Constructor Injection*
- ~ *Autowire*
- ~ *Primary Bean*

=> Spring Boot :

- ~ *Introduction*

=> Spring MVC :

- ~ *Introduction*
- ~ *Creating Controller*
- ~ *Accepting User Input*
- ~ *@RequestParam*
- ~ *@ModelAttribute*
- ~ *Prefix and Suffix*
- ~ *Model and ModelAndView*
- ~ *ModelAttribute*
- ~ *GetMapping and PostMapping*

=> Spring Project :

- ~ Quiz Application

## => Spring ORM :

- ~ Introduction
- ~ Spring Hibernate Config
- ~ MySQL and DAO
- ~ DAO Creation
- ~ Add and Fetch

## => Spring Data JPA :

- ~ Spring Data JPA Configuration
- ~ JPARepository
- ~ JPARepository Add and Fetch
- ~ Query DSL
- ~ Query Annotation

## => Spring REST :

- ~ Introduction
- ~ REST GetMapping
- ~ Jackson
- ~ PathVariable
- ~ RestController
- ~ PostMapping
- ~ Jackson XML
- ~ Produces Attribute
- ~ RequestBody and Consumes Attribute

## => Spring AOP :

- ~ Why AOP
- ~ AOP Terms
- ~ Aspect and Before Annotation
- ~ Logger
- ~ After Finally
- ~ AfterReturning and Throwing

## => Spring Security :

- ~ Introduction
- ~ Implementation
- ~ Managing Users
- ~ Passwords
- ~ Authentication
- ~ Authorization
- ~ CSRF and CORS
- ~ OAuth2
- ~ JWT

## => Spring Mega Project :

- ~ Secure Stock Broker App

## => Agile and Scrum (Optional) :

- ~ Agile Values
- ~ 12 Agile Principles
- ~ Scrum Overview
- ~ Scrum Values
- ~ Concept of Sprints
- ~ Scrum Roles
- ~ Role of Scrum Master
- ~ Role of Product Owner
- ~ Role of Development Team
- ~ Daily Stand-up
- ~ Sprint Planning
- ~ Sprint Review
- ~ Sprint Retrospective
- ~ Backlog Refinement
- ~ User Stories
- ~ Product Backlog and Sprint Backlog
- ~ Working Agreements
- ~ Definition of Ready, Done
- ~ Team Velocity
- ~ Burndown Chart

## => Docker Installation Basics :

- ~ What is Docker?
- ~ How to install Docker and Hello World
- ~ What is container in Docker
- ~ Docker vs Virtual Machine
- ~ First interaction with busy box image

## => Fundamentals of docker :

- ~ Docker lifecycle and PS
- ~ Start and delete a container
- ~ Getting a mongodb container for fun
- ~ Exploring exec command
- ~ Multiple ways to get inside a container

## => Custom Docker images :

- ~ Analogy for custom docker image
- ~ Our first base image and custom image
- ~ Behind the scene for custom image
- ~ Creating a custom mongodb image
- ~ Concept of caching in docker
- ~ Provide a custom name for your image

=> Project and Docker :

- ~ *Introduction to node project for docker*
- ~ *Introduction to node project for docker part 2*
- ~ *Containerize a node application*
- ~ *Performance upgrade in node project container*

=> Multi container setup :

- ~ *Introduction to multi docker container*
- ~ *A mini mongo connector project*
- ~ *Put your node code in a container*
- ~ *Introduction to docker compose*
- ~ *Connect 2 compose images in docker*
- ~ *Access the compose container app with browser*

=> Ngnix - production grade deployment :

- ~ *Ngnix A production grade docker*
- ~ *Attaching volumes in Docker*
- ~ *Types of docker files*
- ~ *Dev test and production stages*
- ~ *Understand react project for docker deployment*
- ~ *Docker for development*
- ~ *Docker for testing*
- ~ *Docker for production*

=> Docker AWS and Travis CI :

- ~ *Docker CI and AWS*
- ~ *What is CI CD Jenkins vs Travis CI*
- ~ *Moving to AWS Elastic Beanstalk*
- ~ *Moving project to github repo*
- ~ *Reading Travis CI documentation*
- ~ *Writing our 1st travis CI config file*
- ~ *AWS IAM user generation*
- ~ *Elastic Beanstalk and S3 bucket*
- ~ *Finally hosting app on AWS with CI integrated with docker*
- ~ *TURN OFF those AWS apps*

# The Complete Front End Web Developer Bootcamp

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/The-Complete-Front-End-Web-Developer-Bootcamp>

## Course Description :-

This course will help you to grab the fundamentals of Front End technologies used in Web Developement and implement them using various projects.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction, getting the dev tools and basics of HTML
- => Div, tables and Forms with Challenge to create FB page
- => HTML 5 semantics and adding audio, video and YouTube to web
- => CSS-Box model, color selection, Google Fonts
- => Web development projects - GYM and Sushi Restro Templates
- => CSS - classes, ID's, parallax and project to edit template
- => CSS - box sizing, gradients and TODO list project
- => Getting started with Bootstrap - Tour and creating landing page
- => Project - Pokemon Corporate site, yahoo selling page and Adm
- => Javascript projects to practice
- => Moving on to learn JQuery - Selectors and event
- => Actions in JQuery - fading, animations and callback function
- => Projects in JQuery and using JS plugins

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction, getting the dev tools and basics of HTML :
  - ~ Course introduction
  - ~ A personal Note
  - ~ Projects that we will build
  - ~ Collecting and installing developers tool
  - ~ Structuring the files and creating first file
  - ~ Text tags
  - ~ List items
  - ~ Ending with Challenge and solution
- => Div, tables and Forms with Challenge to create FB page :
  - ~ Divisions and Spans
  - ~ Images and links
  - ~ Challenge for links on images and solution
  - ~ Tables in HTML
  - ~ More about forms in HTML
  - ~ Challenge to create facebook page and solution
- => HTML 5 semantics and adding audio, video and YouTube to web :
  - ~ Introducing HTML 5

- ~ *Comparing HTML 4 semantics with HTML 5*
- ~ *Adding video 2C audio and youtube videos*

=> CSS-Box model, color selection, Google Fonts :

- ~ *Introduction to css and where to write it*
- ~ *Solving the color selection problem*
- ~ *Comming soon template and backgrounds*
- ~ *Box model and centering text*
- ~ *Google fonts and font awesome*

=> Web development projects - GYM and Sushi Restro Templates :

- ~ *Project - GYM comming soon part 1*
- ~ *Project - GYM comming soon part 2*
- ~ *Project - Restro comming soon with video part 1*
- ~ *Project - Sushi Restro Comming soon part 2*

# RASA

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/RASA>

## Course Description :-

Data Scientists are in high demand right now since they can create a chatbot to handle user communications. Chatbots are predicted to be used by approximately 80% of businesses in the near future. Want to be the one who puts it into action? From creating the first file to the deployment on platforms like Facebook and Telegram, the course is meant to teach you how to create a chatbot that can assist consumers with suggestions of computers and phones from Amazon. The course places a strong emphasis on putting what students learn into practice. You will be guided through the entire process of launching your first chatbot using a combination of document and code walkthroughs.

## Course Features :-

- => Source code
- => Challenges
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Intents and Entities
- => Create Conversational Chatbots
- => Build chatbots using RASA
- => Conversational interfaces and agents in RASA

## Requirements :-

- => Minimal knowledge of Python programming
- => No prior experience required of chatbot
- => Any IDE for Python
- => A system with a decent internet connection
- => Your dedication

## Instructors :-

=> Khushali Shah :

*~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.*

## Curriculum details :-

=> Introduction :

- ~ What is Chatbot? Preview
- ~ Why Chatbot?
- ~ What is Rasa? Preview
- ~ Why Rasa? Preview

=> Create environment and installation :

- ~ Create a virtual environment using conda
- ~ Installation of Rasa in Windows

=> RASA NLU (Natural Language Understanding) :

- ~ Introduction to Rasa NLU - Intents and Entities
- ~ Creating Intents & Entities Examples: Training Data
- ~ Rasa NLU file structure
- ~ Defining NLU pipeline in config file
- ~ Install RASA-x
- ~ Train our first Rasa NLU model
- ~ Rasa NLU entity synonyms & lookup tables

=> Custom components - RASA NLU :

- ~ Introduction to custom components in RASA NLU
- ~ Introduction of transfer learning and pre-trained word embeddings
- ~ Custom gensim embeddings in RASA

=> Introduction to RASA core :

- ~ RASA core-stories Preview
- ~ Custom action defined

- ~ RASA core-stories
- ~ Introduction of dialogue policies
- ~ Memoization & mapping policy
- ~ Machine learning policy
- ~ Priority policies
- ~ Config file to defining policies

=> Conversation with our first bot :

- ~ Introduction use case mental health
- ~ Create intent
- ~ Add intent domain.yml
- ~ Update response
- ~ Add stories.md
- ~ Train model Preview
- ~ Telegram integration
- ~ Facebook integration
- ~ Twilio account
- ~ Whatsapp integration
- ~ Whatsapp integration URL

=> Course summary :

- ~ Course summary



# Data Structure and Algorithm Interview Preparation

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA INTERVIEW

Course link : <https://ineuron.ai/course/Data-Structure-and-Algorithm-Interview-Preparation>

## Course Description :-

This course is designed mostly for Data structure and Algorithms test takers.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => DSA Theoretical Test
- => DSA Practical Test
- => DSA Aptitude Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

=> Data structure and Algorithms Test :

- ~ DSA Test 1
- ~ DSA Test 2
- ~ DSA Test 3
- ~ DSA Test 4
- ~ DSA Test 5
- ~ DSA Test 6
- ~ DSA Test 7
- ~ DSA Test 8
- ~ DSA Test 9
- ~ DSA Test 10
- ~ DSA Test 11
- ~ DSA Test 12
- ~ DSA Test 13
- ~ DSA Test 14
- ~ DSA Test 15
- ~ DSA Test 16
- ~ DSA Test 17
- ~ DSA Test 18
- ~ DSA Test 19
- ~ DSA Test 20

# Game Development using Unity

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Game-Development-using-Unity>

## Course Description :-

This course provides an overview of the core principles of gaming using Unity. You will learn to define a game, as well as the mechanics and rules that govern a variety of games. After finishing this course, you will have a thorough understanding of the entire game development process.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction using Unity
- => Introduction to game engine
- => Introduction to C# programming
- => 2D physics concepts
- => Introduction to animation
- => 3D game development
- => Publishing games to various platforms

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ What is Unity?
  - ~ What do you learn in Unity?
  - ~ Why Unity?
  - ~ Downloading Unity and installation
  - ~ Unity Objects
  - ~ Unity Components
  - ~ Creating your own components
- => Assignment no.1 :
  - ~ Write down the Unity Competitors and alternatives?
  - ~ Are there free gaming softwares available? if yes please name them?
- => Game Engine :
  - ~ Engine concepts
  - ~ Development tools
  - ~ User interface text
  - ~ Countdown timer
  - ~ Digital clock
  - ~ Sprites
- => C# programming :
  - ~ What is scripting?
  - ~ What is c# language?
  - ~ Introduction to C# language
  - ~ C# coding fundamentals
  - ~ Loops and functions
  - ~ If - Else statements

=> 2D Physics concepts :

- ~ Rigidbody components
- ~ Unity colliders
- ~ Physics materials
- ~ scripting collision events
- ~ Importing asset
- ~ Making character move
- ~ Giving jumping abilities

=> Animation :

- ~ Simple Unity animation
- ~ Animator states
- ~ Scripting animations
- ~ Animations and colliders

=> Assignment no.2 :

- ~ Create your own Avatar

=> Sound effects :

- ~ Sound files
- ~ Adding sounds to game objects
- ~ Scripting sounds

=> Assignment no.3 :

- ~ Create a cloud object and give a thunder cloud sound effect

=> 3D game development :

- ~ Creating multiple scenes
- ~ Creating lighting effects
- ~ Creating Controlling character script
- ~ Controlling camera movements
- ~ Implementation of Occlusion culling

=> Assignment no.4 :

~ Create your own game where a rabbit is walking in the farm, if he digs and finds a carrot, give a point and if he digs and finds nothing, decrease the point.

=> Artificial Intelligence :

- ~ What is Artificial intelligence?
- ~ Artificial intelligence concepts
- ~ Flowcharts and algorithms
- ~ Scripting AI

=> Publishing Games :

- ~ Publishing games to PC
- ~ Publishing games to MAC
- ~ Publishing games to LINUX
- ~ Publishing games to Smartphones
- ~ Publishing games to Game Consoles

=> Project :

- ~ Create a game where a boy is riding a cycle on road

# Django

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : DJANGO

Course link : <https://ineuron.ai/course/Django>

## Course Description :-

Django is a Python-based web framework. You'll be developing and learning about Django one step at a time in this course. We'll go through all you need to know about using Python, Django and other web technologies to create a website.

## Course Features :-

- => Quizzes
- => Assignments
- => Hands-on practicals
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Django overview
- => Project implementation
- => Project deployment

## Requirements :-

- => Basic knowledge of Python programming
- => A system with stable internet connection
- => Your dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Introduction :

~ Overview Preview

=> Django core :

- ~ Django features Preview
- ~ Creating a project
- ~ The development server
- ~ Creating the Polls app
- ~ Write your first view
- ~ `path()` argument: route
- ~ `path()` argument: view
- ~ `path()` argument: name

# Tableau Foundation

---

Topic Name : DATA ANALYTICS

Sub-topic Name : TABLEAU

Course link : <https://ineuron.ai/course/Tableau-Foundation>

## Course Description :-

Tableau is a powerful and fastest growing data visualization tool used in the Business Intelligence Industry. It helps in simplifying raw data into the very easily understandable format. Tableau software is one of the fastest growing data visualization tools which is currently being used in the BI industry. It is the best way to change or transform the raw set of data into an easily understandable format with zero technical skills and coding knowledge.

## Course Features :-

=> Lifetime Dashboard

=> Free Course

=> Assignment

=> Quiz

## What you will learn :-

=> Tableau tools

=> Practical implementation

=> Tableau complete understanding

## Requirements :-

=> Computer with Internet Connectivity

=> Basic programing understanding

## Instructors :-

=> Amit Bose :

~

## Curriculum details :-

=> Tableau Working with Dimension & Measures :

~ *Introduction Preview*

=> Working with Dimension & Measures in Tableau

=> 4 0 Tableau Adv Visuals Day3

=> Data Management Filters in Tableau

=> Tableau Sets Parameters

=> Filters in Detail Tableau

=> Advance Visuals in Tableau

=> Tableau - LOD | Data Joining & Blending

=> Tableau - Parameters | Calculated Columns

# Big Data Masters

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA MASTERS

Course link : <https://ineuron.ai/course/Big-Data-Masters>

## Course Description :-

Data is an essential part of any organization. Every organization generates a massive amount of real-time or batch data. This is where Big data plays a vital role irrespective of domain and industry. This complete course is designed to fulfill such requirements so that we will be able to work with a humongous amount of data. You will be able to create your Big Data Engine in your organization by implementing various big data stacks used across the industry.

## Course Features :-

- => Online Instructor-led learning
- => Proper Roadmap
- => Resume Building
- => Lifetime Dashboard access
- => Doubt clearing
- => Quiz in every module
- => Career Counselling
- => Assessments
- => Mock Interview
- => Certificate
- => Job Referrals
- => Get a chance to work with iNeuron Team

## What you will learn :-

- => 30+ Big Data Technologies
- => Big Data Engine Creation
- => Streaming and Batch Processing of Data
- => Various SQL Databases
- => Various NOSQL Databases
- => Real-Time Implementation
- => Spark
- => Hive
- => Talend
- => Informatica
- => Hadoop Distributions
- => Deployment
- => DataBricks Implementation

## Requirements :-

- => Minimum system requirement i3 or higher
- => Dedication

## Curriculum details :-

=> Introduction to Distributed Systems - Hadoop and MapReduce :

- ~ *Why Is Data So Important?*
- ~ *Pre-Requisite Data Scale*
- ~ *What Is Big Data?*
- ~ *Big Bank: Big Challenge*
- ~ *Common Problems*
- ~ *3 Vs Of Big Data*
- ~ *Defining Big Data*
- ~ *Sources Of Data Flood*
- ~ *Exploding Data Problem*
- ~ *Redefining The Challenges Of Big Data*
- ~ *Possible Solutions: Scaling Up Vs. Scaling Out*
- ~ *Challenges Of Scaling Out*
- ~ *Solution For Data Explosion-Hadoop*
- ~ *Hadoop: Introduction*

- ~ Hadoop In Layman's Term
- ~ Hadoop Ecosystem
- ~ Evolutionary Features Of Hadoop
- ~ Hadoop Timeline
- ~ Why Learn Big Data Technologies?
- ~ Who Is Using Big Data?
- ~ HDFS: Introduction
- ~ Design Of HDFS
- ~ Why Hadoop Cluster?
- ~ HDFS Blocks
- ~ Components Of Hadoop 3
- ~ NameNode And Hadoop Cluster
- ~ Arrangement Of Racks
- ~ Arrangement Of Machines And Racks
- ~ Local FS And HDFS
- ~ NameNode
- ~ Checkpointing
- ~ Replica Placement
- ~ Benefits-Replica Placement And Rack Awareness
- ~ URI
- ~ URL And URN
- ~ HDFS Commands
- ~ Problems With HDFS In Hadoop 1.X
- ~ HDFS Federation
- ~ High Availability
- ~ Anatomy Of File Read From HDFS
- ~ Data Read Steps
- ~ Important Java Classes To Write Data To HDFS
- ~ Anatomy Of File Write To HDFS
- ~ Writing File To HDFS: Steps
- ~ Building Principles
- ~ InputSplit
- ~ InputSplit And Data Blocks Difference
- ~ Why Is The Block Size 128 MB?
- ~ RecordReader
- ~ InputFormat
- ~ Default Inputformat : TextInputFormat
- ~ OutputFormat
- ~ Using A Different OutputFormat
- ~ Important Points
- ~ Partitioner
- ~ Using Partitioner
- ~ Map Only Job
- ~ Flow Of Operations In MapReduce
- ~ Serialization In MapReduce
- ~ Schedulers In YARN
- ~ FIFO Scheduler
- ~ Capacity Scheduler
- ~ Fair Scheduler
- ~ Differences Between Hadoop 1.X And Hadoop 2.X and hadoop 3.X

=> Hive :

- ~ Introduction
- ~ Hive DDL
- ~ Demo: Databases.Ddl
- ~ Demo: Tables.Ddl
- ~ Hive Views
- ~ Demo: Views.Ddl
- ~ Architecture
- ~ Primary Data Types
- ~ Data Load
- ~ Demo: ImportExport.Dml
- ~ Demo: HiveQueries.Dml
- ~ Demo: Explain.Hql Table Types
- ~ Demo: ExternalTable.Ddl
- ~ Complex Data Types
- ~ Demo: Working With Complex Datatypes
- ~ Hive Variables
- ~ Demo: Working With Hive Variables
- ~ Hive Variables And Execution Customisation

=> Advanced Hive :

- ~ Working With Arrays
- ~ Sort By And Order By
- ~ Distribute By And Cluster By
- ~ Partitioning
- ~ Static And Dynamic Partitioning
- ~ Bucketing Vs Partitioning
- ~ Joins And Types
- ~ Bucket-Map Join
- ~ Sort-Merge-Bucket-Map Join
- ~ Left Semi Join
- ~ Demo: Join Optimisations
- ~ Input Formats In Hive
- ~ Sequence Files In Hive
- ~ RC File In Hive
- ~ File Formats In Hive
- ~ ORC Files In Hive
- ~ Inline Index In ORC Files

- ~ ORC File Configurations In Hive
- ~ SerDe In Hive
- ~ Demo: CSVSerDe
- ~ JSONSerDe
- ~ RegexSerDe
- ~ Analytic And Windowing In Hive
- ~ Demo: Analytics.Hql
- ~ Hcatalog In Hive
- ~ Demo: Using\_HCatalog
- ~ Accessing Hive With JDBC
- ~ Demo: HiveQueries.Java
- ~ HiveServer2 And Beeline
- ~ Demo: Beeline
- ~ UDF In Hive
- ~ Demo: ToUpper.Java And Working\_with\_UDF
- ~ Optimizations In Hive
- ~ Demo: Optimizations

=> NoSQL and Hbase :

- ~ Challenges With Traditional RDBMS
- ~ Features Of NoSQL Databases
- ~ NoSQL Database Types
- ~ CAP Theorem
- ~ What Is HBase Regions
- ~ HBase HMaster ZooKeeper
- ~ HBase First Read
- ~ HBase Meta Table
- ~ Region Split
- ~ Apache HBase Architecture Benefits
- ~ HBase Vs. RDBMS
- ~ Shell Commands

=> Sqoop :

- ~ Sqoop Architecture
- ~ Sqoop Features
- ~ Sqoop Hands On

=> Python :

- ~ Python Core
- ~ Introduction of python and comparison with other
- ~ Programming language
- ~ Installation of Anaconda Distribution and other python
- ~ IDE Python Objects, Number & Booleans, Strings
- ~ Container objects, Mutability of objects
- ~ Operators Arithmetic, Bitwise, Comparison and Assignment operators, Operators Precedence and associativity
- ~ Conditions(if else,if elif else) Loops(While ,for)
- ~ Break and Continue statement and Range Function.
- ~ String Objects And Collections
- ~ String object basics
- ~ String methods
- ~ Splitting and Joining Strings
- ~ String format functions
- ~ List object basics
- ~ List as stack and Queues
- ~ List comprehensions
- ~ Tuples,Set ,Dictionaries Functions
- ~ Tuples,Sets Dictionary Object basics, Dictionary Object methods, Dictionary View Objects.
- ~ Functions basics, Parameter passing, Iterators Generator functions
- ~ Lambda functions
- ~ Map , Reduce, Filter functions
- ~ OOPS Concepts Working With Files
- ~ OOPS basic concepts
- ~ Creating classes and Objects Inheritance
- ~ Multiple Inheritance
- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other File methods
- ~ Exception Handling Database Programming
- ~ Using Standard Module
- ~ Creating new modules
- ~ Exceptions Handling with Try except
- ~ Creating ,inserting and retrieving Table
- ~ Updating and deleting the data

=> SQL :

- ~ Installing and configuring MySQL
- ~ Install and Configure MySQL Client
- ~ DDL- Create database/table, Drop, Alter, etc
- ~ DML - INSERT, DELETE, UPDATE, MERGE etc
- ~ DML - INSERT, DELETE, UPDATE, MERGE etc
- ~ DQL - SELECT,etc
- ~ JOINS - One Many, Many Many
- ~ DISTINCT
- ~ ORDER BY
- ~ LIMIT
- ~ WILD CARDS
- ~ LOGICAL OPERATORS - LIKE, EQUAL, AND, OR etc
- ~ STRING Functions
- ~ DATE Functions



- ~ MATH Functions
- ~ COUNT, MIN and MAX
- ~ SUM
- ~ AVG
- ~ LAG and LEAD function Examples
- ~ Top N Analysis
- ~ ROW\_NUMBER
- ~ RANK AND DENSE\_RANK
- ~ CASE WHEN
- ~ PIVOT
- ~ LISTAGG
- ~ UNION
- ~ Sub-Queries
- ~ EXISTS
- ~ NOT EXISTS
- ~ WITH CLAUSE
- ~ Recursive WITH & CTE
- ~ Regular Expressions in SQL

=> Cassandra :

- ~ Cassandra Introduction
- ~ Cassandra Installation in local system
- ~ DATASTAX Cassandra setup
- ~ Cassandra ArchitectureCassandra Queries

=> MongoDB :

- ~ MongoDB Introduction
- ~ MongoDB Compass Setup
- ~ MongoDB Atlas Setup
- ~ MongoDB Architecture
- ~ MongoDB Queries

=> Spark :

- ~ Introduction To Apache Spark
- ~ Map Reduce Limitations
- ~ RDD's
- ~ Spark Context - SQLContext And HiveContext
- ~ Programming With RDD's
- ~ Creating RDD's From Text-Files
- ~ Transformations And Actions
- ~ How Does Spark Execution Work
- ~ RDD API's - Filter
- ~ FlatMap
- ~ Fold
- ~ Foreach
- ~ Glom
- ~ GroupBy
- ~ Map
- ~ ReduceByKey
- ~ Zip
- ~ Persist
- ~ Unpersist
- ~ Read/Write From Storage
- ~ RDD Examples
- ~ RDD API's - Aggregate
- ~ Cartesian
- ~ Checkpoint
- ~ Coalesce
- ~ Repartition
- ~ Cogroup
- ~ CollectAsMap
- ~ CombineByKey
- ~ Count And CountApprox Functions
- ~ More RDD Examples
- ~ Schema - StructType
- ~ StructFields
- ~ DataType
- ~ DataFrame API's And Examples
- ~ Create Temporary Tables
- ~ SparkSQL
- ~ Spark Dataset
- ~ Parquet Vs Avro
- ~ Examples And Problem Solving On Real Data Using RDD And Converting The Same To Dataframe
- ~ Create A Spark Project
- ~ SBT / Maven
- ~ How Do Maven Repo Work
- ~ Accumulators
- ~ BroadCast Variables
- ~ Query Execution Plan
- ~ Internal Of Spark Workings

=> DATABRICKS :

- ~ Databricks Introduction
- ~ Databricks Setup
- ~ Databricks Integration with cloud
- ~ Databricks OPS Pipeline
- ~ Databricks in Production

=> Kafka :

- ~ Introduction To Kafka

- ~ Kafka Architecture
- ~ Kafka Key Concepts/Fundamentals
- ~ Overview Of Zookeeper And Its Role In Kafka Cluster
- ~ Cluster, Nodes, Brokers, Topics Consumer, Producers, Logs, Partitions Concept Of Consumer Groups
- ~ Leader & Follower Partition
- ~ Installing One Node Kafka Cluster On Local Installing Multinode Kafka Cluster On Local Command Line Producer And Consumer Replication Concept For Fault Tolerance How Data Is Stored In Brokers
- ~ Log Segments, Message Offsets, Message Index
- ~ Isr List / Minimum Isr
- ~ Committed Vs Uncommitted Messages Writing A Kafka Producer In Java Writing A Kafka Consumer In Java Scaling Up The Kafka Cluster Achieving Exactly Once Semantics
- ~ Integrating Kafka With Spark Structured Streaming.

=> Apache Airflow - Workflow Management Platform :

- ~ Introduction To Airflow And Its Usage What Is Workflow
- ~ Cron-Job Creation Example Airflow Additional Features
- ~ Airflow Architecture And Components Airflow Installation Demo
- ~ Dags-Creating A Simple Helloworld Dag Introduction To Tasks And Operators
- ~ Viewing The DAG In Ui-Graph View, Tree View, Logs Viewing
- ~ Example Showcasing Bash Operators Usage Setting Precedence Among Various Tasks Lifecycle Of ATask-Understanding Various Stages About Trigger\_rules & Understanding With Example Airflow Artifact - More On Operators
- ~ Writing Our Own Custom Operators Walkthrough Of Airflow UI
- ~ Connections To Various Datastores & Variables
- ~ Working With Connections, Understanding Sensors Demo
- ~ Building an end-to-end customer-360 pipeline using Airflow involving data collection from various sources, processing in spark, loading the processed data in hive and uploading the same to HBase and generating a notification about success of the pipeline to the downstream applications.

=> Spark Streaming :

- ~ Kind of Processing
- ~ What is Real-time Processing
- ~ The Importance of Real-time Processing
- ~ Batch processing vs Real-time Stream Processing Spark Streaming Data
- ~ Spark discretized stream or DStream Batch & Batch Interval
- ~ Do Spark is a real-time streaming engine Stream Processing in Spark Transformed DStream
- ~ Understanding Producer & Consumer Practical on Realtime Processing Stream Transformations
- ~ Stateless Transformations Stateful Transformations Window Operations
- ~ Batch Interval Window Size Sliding Interval
- ~ Practical on Stateless Transformation Practical on Stateful Transformation reduceByKey vs updateStateByKey Working With Sliding Window reduceByKeyAndWindow Transformation reduceByWindow Transformation countByWindow Transformation
- ~ What Is Structured Streaming Requirement Of Structure Streaming Limitations Of Spark Streaming Benefits Of Spark Structure Streaming
- ~ Practical Wordcount Example On Structured Streaming
- ~ Dynamically Setting The Shuffle Partitions Data Stream Writer Output Modes
- ~ Datastream Output Modes - append, update & complete
- ~ Spark Streaming Graceful Shutdown
- ~ How Does Spark Streaming Code Executes Internally How a Job Converted to Micro batches
- ~ Trigger Point For Micro Batches
- ~ Types of Triggers unspecified, time interval, one time, continuous
- ~ Types of Data Sources Socket Source, Rate Source, File Source, Kafka Source
- ~ Limitations of socket source Practical on File Data Source
- ~ Types of Spark Streaming Output Data Options Fault Tolerance and Exactly Once Guarantee Understanding Checkpoint Location
- ~ Stateful vs Stateless Transformations
- ~ Managed Stateful Operations vs UnManaged Stateful Operations
- ~ Types of Aggregations - Continuous Aggregations vs Time Bound Aggregations
- ~ Window Transformations
- ~ UpdateStateByKey, reduceByKeyAndWindow, reduceByWindow, countByWindow
- ~ Types of windows - Tumbling Time Window, Sliding Time Window
- ~ Dealing With Late Coming Records Using Watermark
- ~ State Store Cleanup
- ~ Calculating the Watermark Boundary Streaming Joins
- ~ Streaming Dataframe to static dataframe
- ~ Streaming Dataframe With Another Streaming Dataframes

=> Big Data on Cloud :

- ~ AWS EMR (Elastic MapReduce):
- ~ What is a VM (Virtual Machine) On-Premise vs Cloud Setup
- ~ Major Vendors of Hadoop Distribution Why Cloud & Big Data on Cloud Major Cloud Providers of Bigdata What is EMR
- ~ Hdfs vs S3 What Is S3
- ~ Important Instances in AWS Kinds of Nodes in Cluster
- ~ Transient vs Long Running Cluster Running Spark Code on Emr
- ~ How to Track Your Job
- ~ Copy File From S3 to Local Zeppelin Notebook
- ~ Types of EC2 Instances How to Create a VM What is a Keypair Elastic IP
- ~ AWS Storage, Networking & CLI Instance Store
- ~ S3 & EBS
- ~ Public ip Vs Private Ip Network Switches Security Group
- ~ Aws Command Line Interface
- ~ Launch A Emr Cluster Using Advanced Options
- ~ AWS Athena
- ~ What is Athena?
- ~ When do we require Athena What problem Athena Solve How Athena Works
- ~ Athena Pricing
- ~ Athena Practical Demonstration
- ~ How to create a normal table manually on csv data residing in s3
- ~ How to minimize data scanning in Athena How to create partition table on Parquet file
- ~ Inferring Schema automatically using AWS Glue
- ~ AWS Glue
- ~ What is AWS Glue? Introduction To Glue Features of Glue AWS Glue Benefits
- ~ AWS Glue Terminology
- ~ Pointing to Specific Data Stores and Endpoints Glue Data Catalogue

- ~ Crawlers
- ~ Connecting to Your Data Store Using Crawlers for Catalogue Tables
- ~ Overview and Working of Glue Jobs Adding New Jobs in Glue
- ~ Triggering Jobs and Their Scheduling
- ~ AWS Redshift
- ~ Database vs Data Warehouse vs Data Lake Introduction to Amazon Redshift
- ~ Benefits of Amazon Redshift Use Cases of Amazon Redshift
- ~ Redshift Master Slave Architecture Types of Nodes
- ~ Redshift Spectrum Redshift Fault Tolerance Redshift Sort Keys
- ~ Redshift Distribution Styles Practical Demonstration

#### => Spark ML :

- ~ Basic statistics
- ~ Data sources
- ~ Pipelines
- ~ Extracting, transforming and selecting features
- ~ Classification and Regression
- ~ Clustering
- ~ Collaborative filtering
- ~ Frequent Pattern Mining
- ~ Model selection and tuning
- ~ Advanced topics

#### => Enterprise Big Data ETL Tools :

- ~ Introduction to ETL from Talend Studio- Integration with HDFS, Hive, Sqoop, Spark etc
- ~ Introduction to ETL from Informatica BDM- Integration with HDFS, Hive, Sqoop, Spark etc

#### => PROJECT AND INTERVIEW PREPARATION :

- ~ End-to-end Big Data Pipeline Engine PROJECT
- ~ Involving all Major components like
- ~ Sqoop, Hdfs, Hive, Hbase, Spark... etc.
- ~ Interview Preparation Tips
- ~ Sample Resume
- ~ 300+ Mock Interview Recordings
- ~ Mock Interview QA
- ~ Interview Questions
- ~ How to Handle Various Interview Round Qs
- ~ Career Guidance
- ~ One to One Resume Discussion
- ~ Certification

# Web Automation Using Selenium Community Class

---

Topic Name : TESTING

Sub-topic Name : AUTOMATION TESTING

Course link : <https://ineuron.ai/course/Web-Automation-Using-Selenium-Community-Class>

## Course Description :-

In this course you will learn automation testing using Selenium. Selenium is one of the testing suite which has different components Selenium WebDriver, Selenium IDE, Selenium Grid. During this course, you will learn how to automate web application using Selenium 4. You can automate Smoke test, Regression test and end to end test cases.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Getting Started With Automation Testing - Orientation Program
- => Roadmap to learn Automation Testing
- => Different tools for automation in each category
- => Web Automation using Selenium
- => Interacting with Web Elements
- => Automating Web Application End-to-End Scenarios Using Selenium

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Mukesh Otwani :

~ Myself Mukesh Otwani having 10 years of experience in Automation Testing and worked with Dell International and SAP Labs India. I am also passionate teacher, mentor, have been into teaching for 8 years now and running a YouTube channel with 137000 subscribers and 480+ videos on different tools and libraries.

## Curriculum details :-

- => Day 1 :
  - ~ Getting Started With Automation Testing - Roadmap to learn Automation Testing
- => Day 2 :
  - ~ Getting Started With Selenium - Downloading and Installation - WebDriver and WebElement Commands
- => Day 3 :
  - ~ Interacting With WebElements
- => Day 4 :
  - ~ Automating Web Application Using Selenium 4

# Aptitude Live Class

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Aptitude-Live-Class>

## Course Description :-

Quantitative aptitude is a test that assesses a person's numerical and problem-solving abilities. This is a common section seen on most competitive examinations. This Aptitude course has been designed to help students get started and succeed in tests and interviews.

## Course Features :-

- => Online live classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Numbers & Algebra
- => Percentage
- => Average
- => Time & Work
- => Distance
- => Time & Speed
- => Ratio, Proportion & Mixture

## Requirements :-

- => No Prior knowledge
- => A system with internet connection.
- => Your dedication

## Instructors :-

=> Prerna Singh :

*~ I have guided and mentored children for 8+ Years. I teach mathematics to children across grades 9-12. Also having helped children for more than 8 years & being a university topper, I know how to guide children best when it is about performing under stress and managing time in the best possible way. I have experience in both taking live classes and delivering offline sessions to children. Being a passionate math educator & enthusiast, helps me deliver the best of my capabilities. It also helps deliver interactive sessions.*

## Curriculum details :-

=> Number Systems :

- ~ Numbers & their types
- ~ Prime numbers
- ~ Divisibility
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> HCF & LCM :

- ~ Definition
- ~ Factors & Multiples
- ~ Methods to find HCF, LCM
- ~ HCF & LCM of fractions
- ~ Tricks & Tips
- ~ Formulae
- ~ Previous year questions

=> Simplification :

- ~ BODMAS rule
- ~ Modulus of real number
- ~ Vinculum
- ~ Questions
- ~ Formulae
- ~ Tricks & tips

~ *Previous year questions*

=> Surds and Indices :

~ *Definition*

~ *Types*

~ *Laws*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Problems on Ages :

~ *Introduction*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Time Speed Distance :

~ *Introduction*

~ *Average speed*

~ *Relative Speed*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Time and Work :

~ *Basics & Concepts*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Boats and Streams :

~ *Basics & Concepts*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Pipes and Cisterns :

~ *Introduction*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Progressions :

~ *AP, GP, HP Basics*

~ *Sequence & Series Difference*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Averages :

~ *Introduction*

~ *Definition & types*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Alligations and Mixtures :

~ *Introduction*

~ *Types*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Percentages :

~ *Basics & Concepts*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Profit Loss :

~ *Introduction*

~ *Basics & Concepts*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> SI & CI :

~ *Introduction*

~ *Questions*

~ *Formulae*

~ *Tricks & Tips*

~ *Previous year questions*

=> Ratio and Proportions :

- ~ *Introduction*
- ~ *Concepts & Definitions*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Probability :

- ~ *Introduction & examples*
- ~ *Experiment*
- ~ *Sample space*
- ~ *Event & its probability*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Permutation & Combination :

- ~ *Introduction*
- ~ *Permutations*
- ~ *Combinations*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Heights and Distances :

- ~ *Basics of trigonometry*
- ~ *Trigonometric identities*
- ~ *T-ratios*
- ~ *Angel of elevation & depression*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Problems on Trains :

- ~ *Introduction*
- ~ *Various types of problems on trains*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Perimeter, Volume & Area :

- ~ *Introduction*
- ~ *Results on some polygons*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Partnership :

- ~ *Introduction*
- ~ *Working & sleeping partners*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Quadratic Equations :

- ~ *Introduction*
- ~ *Methods of finding roots*
- ~ *Nature of roots*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Coordinate Geometry :

- ~ *Introduction*
- ~ *Cartesian system*
- ~ *Quadrants*
- ~ *Location of points*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Logarithms :

- ~ *Definition*
- ~ *Types*
- ~ *Properties*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Set Theory :

- ~ *Definition*
- ~ *Types*
- ~ *Operations*
- ~ *Questions*
- ~ *Formulae*

- ~ *Tricks & tips*
- ~ *Previous year questions*

=> **Geometry :**

- ~ *Introduction*
- ~ *Plane geometry*
- ~ *Solid geometry*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> **Work & Wages :**

- ~ *Introduction*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> **Square & Cube Root :**

- ~ *Square root*
- ~ *Cube root*
- ~ *Questions*
- ~ *Race*
- ~ *Introduction*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> **Stocks & Shares :**

- ~ *Introduction*
- ~ *Some facts*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> **Chain Rule :**

- ~ *Direct proportion*
- ~ *Indirect Proportion*

=> **Algebra :**

- ~ *Introduction & theory*
- ~ *Questions*
- ~ *Formulae & summary*
- ~ *Tricks & tips*
- ~ *Previous year questions*



# Deep Learning ANN

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/Deep-Learning-ANN>

## Course Description :-

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. It is a function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Learn Deep Learning, Transfer Learning and Neural Networks using the latest frameworks. Become a Deep Learning Expert !!

## Course Features :-

- => Source code
- => Roadmap
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Neural Networks
- => Perceptron
- => Evolution of Neural Networks
- => Maths behind concepts of Neural Networks
- => Back propagation
- => Problems faced while training Neural Networks and its solution

## Requirements :-

- => Basic programming knowledge
- => A system with a decent internet connection
- => Your dedication

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

=> Introduction :

- ~ Introduction to Deep Learning Preview
- ~ Importance of Deep Learning
- ~ Why you should study Deep Learning? (Motivation)
- ~ ANN vs BNN
- ~ The first Artificial Neuron

=> Perceptron :

- ~ Overview of Perceptron Preview
- ~ More about Perceptron
- ~ Perceptron implementation using python - 1
- ~ Perceptron implementation using python - 2
- ~ Perceptron implementation using python - 3
- ~ Perceptron implementation using python - 4
- ~ Perceptron implementation using python - 5
- ~ Perceptron implementation using python - 6
- ~ Perceptron implementation using python - 7
- ~ Python scripting & modular coding for Perceptron
- ~ Python logging basics and docstrings
- ~ Python packaging, Github actions, and PyPI

=> ANN -1 :

- ~ Multilayer Perceptron
- ~ Forward propagation
- ~ Why we need Activation function?
- ~ ANN implementation using tf.keras - 1
- ~ ANN implementation using tf.keras - 2
- ~ ANN implementation using tf.keras - 3
- ~ ANN implementation using tf.keras - 4

~ ANN with Callbacks | Tensorboard | Early Stopping | Model Checkpointing

=> ANN - 2 :

- ~ Vector
- ~ Differentiation
- ~ Partial differentiation
- ~ Maxima and minima concept
- ~ Gradient descent basics
- ~ In-depth understanding of Gradient descent with mathematical proof

=> ANN - 3 :

- ~ Chain rule
- ~ Back propagation

=> ANN - 4 :

- ~ General problems in training Neural Networks
- ~ Vanishing and Exploding gradients
- ~ Activation function basics
- ~ Weight initialization
- ~ Activation functions - 1
- ~ Activation functions - 2
- ~ Activation functions - 3
- ~ Transfer learning
- ~ Batch normalization -1
- ~ Batch normalization -2
- ~ Batch normalization -3

=> ANN - 5 :

- ~ Introduction to fast optimizers
- ~ Momentum optimization
- ~ NAG
- ~ Elongated bowl problem | AdaGrad
- ~ RMSProp
- ~ Adam
- ~ Loss functions
- ~ Regularization
- ~ Dropout

# Airflow

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Airflow>

## Course Description :-

Airflow is a tool developed by Apache for automating and scheduling tasks, data pipelines and workflows. It makes the management of data pipelines easy to manage and provides great functionalities and user interface to create pipelines.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Airflow in detail
- => Course completion certificate

## What you will learn :-

- => What is Airflow?
- => Why Airflow?
- => DAGs, Schedulers, Workflows
- => HDFS
- => Capstone project

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn
- => Python
- => Docker
- => HDFS

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Introduction :
  - ~ Welcome to course Preview
  - ~ Introduction to Apache Airflow Preview
  - ~ Conventional Scheduling Approaches
  - ~ Why move to airflow
  - ~ Basic Terminologies in Airflow DAG
  - ~ Operators
  - ~ scheduling the dags
  - ~ Executors
  - ~ Tasks and Workflow
- => Architecture of Apache Airflow :
  - ~ Architecture of Airflow
  - ~ Life Cycle of Task
- => Installation :
  - ~ Docker installation
  - ~ Airflow Installation Part 1 Preview
  - ~ Airflow Installation Part 2
- => Understanding Directories in Air :
  - ~ understanding compose files
  - ~ understanding other directories
- => Airflow UI Tour :
  - ~ First look of Airflow UI

- ~ Running Default DAG in UI
- ~ Views in UI
- ~ Understanding DAG Definition file
- ~ DAG File execution

=> What are Operators :

- ~ What are Operators

=> Project :

- ~ Project Requirements
- ~ writing project compose file part 1
- ~ writing project compose file part 2
- ~ App Password generation
- ~ writing projects dag file part 1
- ~ writing projects dag file part 2
- ~ writing projects dag file part 3
- ~ writing projects dag file part 4
- ~ creating connections in UI part 1
- ~ creating connections in UI part 2
- ~ full project explanation
- ~ mysql view table

=> Airflow CLI Basics :

- ~ Running project Dag in airflow CLI part 1
- ~ Running project Dag in airflow CLI part 2
- ~ Running project Dag in airflow CLI part 3

=> Executors in airflow :

- ~ What are executors
- ~ Sequential Executor
- ~ Local Executor
- ~ Celery Executor

# Salesforce Administrator

---

Topic Name : SALESFORCE

Sub-topic Name : SALESFORCE ADMINISTRATOR

Course link : <https://ineuron.ai/course/Salesforce-Administrator>

## Course Description :-

This hands-on Salesforce Administration course is for IT professionals and students interested in learning the fundamentals of Salesforce Administration duties. The goal of this course is to provide a foundation in Salesforce Administration activities. You will learn how to use CRM products, gain a foundational understanding of Sales Cloud, Service Cloud, and Customer Service, manage users, apply validations and formulas, provide object level security and field level security, manage workflows and data, build processes, create communities, and administer emails after completing this course.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Cloud Computing And Types
- => Salesforce And Its Products
- => Sales Cloud-Generic Business Process
- => Relationships In Salesforce

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Definition Of Cloud Computing And Types :
  - ~ Definition of cloud computing
  - ~ On-demand advantages of Cloud computing
  - ~ Services of Cloud computing
  - ~ SaaS(Software as a Service)
  - ~ PaaS(Platform as a Service)
  - ~ IaaS(Infrastructure as a Service)
  - ~ Types of Clouds
  - ~ Public Cloud
  - ~ Private Cloud
  - ~ Hybrid Cloud
  - ~ Community Cloud
- => Definition Of Salesforce And Its Products :
  - ~ Importance of Salesforce
  - ~ Importance of CRM and Solutions
  - ~ Growth of Revenue using CRM
  - ~ comparison of Salesforce CRM with other CRM Products
  - ~ Salesforce Products
  - ~ Salesforce Features and Edition Limits
- => Sales Cloud-Generic Business Process :
  - ~ Features of Sales Cloud
  - ~ Products
  - ~ Campaign
  - ~ Lead
  - ~ Account
  - ~ Opportunity
  - ~ Contact
  - ~ Contract
  - ~ QuoteManage
  - ~ Order
- => Service Cloud and Customer Service :
  - ~ Features of Service cloud
  - ~ case

- ~ *Solution*
- ~ *Public Knowledge Base*
- ~ *Web-to-case*
- ~ *Self-Service Portal*
- ~ *Escalation rules*

#### => **Company Information :**

- ~ *Creating Company Profile*
- ~ *Setting Fiscal Year, Business Hours*
- ~ *Setting Holidays and Language*
- ~ *Identifying Edition*

#### => **Salesforce-Force.Com Platform :**

- ~ *Creating User Login Credentials*
- ~ *Setup-Personal Setup*
- ~ *About Administration Setup*
- ~ *Standard Applications, Tabs and Objects*
- ~ *Creating Custom Application(Design), Custom Objects and Custom Tabs*
- ~ *Creating Custom Fields Using DataTypes and Picklist and Dependencies*

#### => **Manage Users :**

- ~ *Creating users and Roles*
- ~ *Creating Custom Profiles*
- ~ *Discussion on Standard Profiles*
- ~ *Creating Permission Sets and Public Groups and Queues*
- ~ *Login History*

#### => **Relationships In Salesforce :**

- ~ *Lookup Relationship*
- ~ *Master-Detail Relationship*
- ~ *Many to many Relationships*
- ~ *Junction Object*
- ~ *Rollup Summary Fields*
- ~ *RealTime Scenarios*

#### => **Applying Validations And Formulas :**

- ~ *Overview of validation*
- ~ *Creating Validation Rule*
- ~ *Apply Formula in Validation*
- ~ *PageLayout on Objects*
- ~ *Mini PageLayout*
- ~ *Search Layout*
- ~ *Record Type*
- ~ *Field Level Security on Layout*

#### => **Object Level Security Model (Table) :**

- ~ *Profile Level*
- ~ *Permission Set Level*

#### => **Field Level Security Model (Column) :**

- ~ *Profile Level*
- ~ *Page Layout Security*
- ~ *Permission Set Security*

#### => **Record Level Security Model-Sharing Settings(Row) :**

- ~ *Overview of Record Level security*
- ~ *Organization-Wide-Default*
- ~ *Sharing Rules*
- ~ *Grant Access Using Hierarchies*
- ~ *Internal Access*
- ~ *External Access*
- ~ *Manual Sharing*
- ~ *Owner Based Sharing Rule*
- ~ *Both Usage of Profile and OWD*
- ~ *RealTime Scenarios*

#### => **Workflows And Approvals :**

- ~ *Overview of Workflows*
- ~ *Email Alerts*
- ~ *Tasks*
- ~ *Field Updates*
- ~ *Outbound Messages*
- ~ *Time Dependent Workflow Actions*
- ~ *Real Time Scenarios*

#### => **Data Management With SFDC :**

- ~ *Overview Import wizard*
- ~ *Limitations on Import Wizard*
- ~ *Data Export*
- ~ *Import Objects*
- ~ *.CSV File usage in salesforce*
- ~ *Import Data into Salesforce*
- ~ *Data Loader*
- ~ *Mass Transfer Records and Delete of Records*

#### => **Process Builder :**

- ~ *Overview of Process Builder*
- ~ *Working with Process Builder*
- ~ *Workflows With Process Builder*

#### => **Community Creation :**

- ~ *Customer community*

- ~ Partner Community
- ~ Creating User On Community
- ~ Self Registration

=> Security Settings :

- ~ Single-Sign-on settings
- ~ Session Settings
- ~ Password Policies
- ~ Identity Provides
- ~ Login Access
- ~ Identity Connect
- ~ Email Admin Setup

=> Email Administration :

- ~ Deliverability
- ~ Test Deliverability
- ~ Organization-Wide Addresses
- ~ Setting Email Footers
- ~ Compliance BCC Email

=> Reports And Dashboards :

- ~ Importance of Reports
- ~ Discussion On Standard Report
- ~ Creating Custom Report, Tabular Report, Summary report, Matrix Report
- ~ Creating Joined Report and Report Types
- ~ Applying Filters on Report
- ~ Run Report
- ~ Export Report
- ~ Create Dashboard
- ~ Fetching Dashboard into Visualforce

# Interactive Visualization using Seaborn

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Interactive-Visualization-using-Seaborn>

## Course Description :-

This course will help learners to understand the fundamentals of data visualization with seaborn. In this course, You will learn how to generate line plots, scatterplots, histograms, distribution plots, pair plots, bar plots, count plots, and many more. Upon successful completion, you can create beautiful visualizations and extract insights out of them.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction of Seaborn
- => Environment Setup
- => Importing Dataset and Libraries
- => Different Types of Plot in Seaborn
- => Statistical Estimation
- => Project

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ Course Introduction
  - ~ Who is this course for?
  - ~ Course Overview & Course outcome
  - ~ Course Pre-requisite
  - ~ What are graphs?
  - ~ What can we depict from graphs?
  - ~ What are the different types of graphs?
  - ~ What is Seaborn?
  - ~ Why is Seaborn used?
- => Assignment 1 :
  - ~ Which is better Seaborn or Matplotlib and what is the advantage of using one over another.
- => Installation of Seaborn :
  - ~ Introduction to Google Colab
  - ~ The convenience of using Google Colab
  - ~ Setting up Google Colab
- => Different Types of Plot :
  - ~ What is a scatter plot?
  - ~ What can we understand from the scatter plot?
  - ~ Applying scatter plot on air pollution dataset
  - ~ What is a Line plot?
  - ~ What can we understand from the Line plot?
  - ~ Applying Line plot on air pollution dataset
  - ~ What is a Bar plot?
  - ~ What can we understand from the Bar plot?
  - ~ Applying Bar plot on air pollution dataset
  - ~ What is a Count plot?
  - ~ What can we understand from the Count plot?



- ~ Applying Count plot on air pollution dataset
- ~ What is a Box plot?
- ~ What can we understand from the Box plot?
- ~ Applying Box plot on air pollution dataset
- ~ What is a Violin plot?
- ~ What can we understand from the Violin plot?
- ~ Applying Violin plot on air pollution dataset
- ~ What is a Strip plot?
- ~ What can we understand from the Strip plot?
- ~ Applying Strip plot on air pollution dataset
- ~ What is a Swarm plot?
- ~ What can we understand from the Swarm plot?
- ~ Applying Swarm plot on air pollution dataset
- ~ What is a Factor plot?
- ~ What can we understand from the Factor plot?
- ~ Applying Factor plot on air pollution dataset
- ~ What is a Histogram?
- ~ What can we understand from the Histogram?
- ~ Applying Histogram on air pollution dataset
- ~ What is a KDE plot?
- ~ What can we understand from the KDE plot?
- ~ Applying KDE plot on air pollution dataset
- ~ What is a Heat map?
- ~ What can we understand from the Heat map?
- ~ Applying Heat map on air pollution dataset
- ~ What is a catplot?
- ~ What can we understand from the catplot?
- ~ Applying catplot on air pollution dataset

=> Assignment 2 :

- ~ Apply a scatter plot on your own dataset and write down your observations from them.

=> Assignment 3 :

- ~ Apply Line plot on your own datasets and write down your observations from it.

=> Assignment 4 :

- ~ Apply Bar plot on your own dataset and write down your observations from it.

=> Assignment 5 :

- ~ Apply Count plot on your own dataset and write down your observations from it.

=> Assignment 6 :

- ~ Apply Box plot on your own dataset and write down your observations from it.

=> Assignment 7 :

- ~ Apply Violin plot on your own dataset and write down your observations from it.

=> Assignment 8 :

- ~ Apply Strip plot on your own dataset and write down your observations from it.

=> Assignment 9 :

- ~ Apply Swarm plot on your own dataset and write down your observations from it.

=> Assignment 10 :

- ~ Apply Factor plot on your own dataset and write down your observations from it

=> Assignment 11 :

- ~ Apply Histogram on your own dataset and write down your observations from it.

=> Assignment 12 :

- ~ Apply KDE plot on your own dataset and write down your observations from it.

=> Assignment 13 :

- ~ Apply a Heat map on your own dataset and write down your observations from it.

=> Assignment 14 :

- ~ Apply catplot on your own dataset and write down your observations from it.

=> Course Summary :

- ~ Course outro
- ~ Future learning Path

=> Project :

- ~ Using seaborn analyze geographical datasets

# C++ Job Preparation

---

Topic Name : PROGRAMMING

Sub-topic Name : C++

Course link : <https://ineuron.ai/course/C++-Job-Preparation>

## Course Description :-

C++ Interview-ready course has been created specifically to familiarize you with the types of questions you may encounter during the interview. This course is for the people who want to prepare for their interview after learning all the initial concepts & theories with Project Building. After completing this course, you will be confident enough to crack any discussion related to C++.

## Course Features :-

- => Challenges
- => Downloadable resources
- => Quizzes
- => Assignments in each module
- => Completion certificate

## What you will learn :-

- => Tackle difficult interview questions
- => Present projects in interview
- => Create application
- => Distribution
- => Packaging
- => Architecture design
- => Bug testing
- => Modular coding
- => Object-oriented programming STL (Structured Template Library) for competitive programming.

## Requirements :-

- => Prior knowledge of C++ language
- => A system with internet connection.
- => Dedication

## Instructors :-

=> Umang Pincha :

~ Data Warehouse/Business Intelligence Developer @ Amdocs & a Competitive Programmer. He is also having a good knowledge in C , C++ , Data Structure , Python , ML I can easily make websites on it and have done PG in Machine Learning and Artificial Learning from NIT, WARANGAL

## Curriculum details :-

=> Essentials and Fundamentals :

- ~ What is the difference between new/delete and malloc/free? Preview
- ~ What is the difference between new and malloc? Preview
- ~ What is the difference between delete and delete[]?
- ~ What is difference between malloc()/free() and new/delete?
- ~ What is the difference between "new" and "operator new"?
- ~ What is Memory alignment?
- ~ Is there a way to force new to allocate memory from a specific memory area?
- ~ How does free know the size of memory to be deleted.?
- ~ int \*i = (int \*)malloc(12); followed by free(i); how did free function call know how much of memory to delete?
- ~ How do I allocate multidimensional arrays using new ones?
- ~ Can I free() pointers allocated with new? Can I delete pointers allocated with malloc()?
- ~ Why should I use new instead of trustworthy old malloc()?
- ~ Can I use realloc() on pointers allocated via new?
- ~ Do I need to check for NULL after p = new Fred()?
- ~ How can I convince my (older) compiler to automatically check new to see if it returns NULL?
- ~ Do I need to check for NULL before deleting p?
- ~ What are the two steps that happen when I say delete p?
- ~ In p = new Fred(), does the Fred memory "leak" if the Fred constructor throws an exception?
- ~ How do I allocate/deallocate an array of things?
- ~ What if I forget the [] when deleting the array allocated via new T[n]?
- ~ Can I drop the [] when deleting an array of some built-in type (char, int, etc.)?
- ~ After p = new Fred[n], how does the compiler know there are n objects to be destructed during delete[] p?
- ~ Is it legal (and moral) for a member function to say delete this?
- ~ C++ Basics & Conditional Statements
- ~ How do you link a C++ program to C functions?
- ~ Is there anything you can do in C++ that you cannot do in C?

- ~ What are the differences between a struct in C and in C++?
- ~ What does `extern "C" int func(int *, Foo)` accomplish?
- ~ What are the access privileges in C++? What is the default access level?
- ~ How does C++ help with the tradeoff of safety vs usability?
- ~ What are the benefits of operator overloading?
- ~ What are some examples of operator overloading?
- ~ But operator overloading makes my class look ugly; isn't it supposed to make my code clearer?
- ~ Can I overload operator`==`, so it lets me compare two `char[]` using a string comparison?
- ~ Can I create an operator`**` for "to-the-power-of" operations?
- ~ Okay, that tells me the operators I can override; which operators should I override?
- ~ What are some guidelines / "rules of thumb" for overloading operators?
- ~ Base Class Pointer & Derived Class Object
- ~ What is a dangling pointer?
- ~ What is Memory Leak?
- ~ What is an auto pointer?
- ~ What issue do `auto_ptr` objects address?
- ~ What is a smart pointer?
- ~ Is there any problem with the following : `char*a=NULL; char& p = *a;`
- ~ What is the difference between a pointer and a reference?
- ~ What is the difference between `const char *myPointer` and `char *const myPointer`?
- ~ When should I use references, and when should I use pointers?

=> String :

- ~ How to convert an integer to a string?
- ~ Programs related to string.

=> OOPS :

- ~ Is it possible to have a Virtual Constructor? If yes, how? If not, why not possible? Is it possible to have Virtual Destructor? If yes, how? If not, Why not possible?
- ~ What is a constructor or ctor?
- ~ What is the difference between a copy constructor and an overloaded assignment operator?
- ~ Can a constructor throw an exception? How to handle the error when the constructor fails?
- ~ What is a default constructor?
- ~ What is a copy constructor?
- ~ When are copy constructors called?
- ~ Can a copy constructor accept an object of the same class as a parameter instead of an object reference?
- ~ What is a conversion constructor?
- ~ What is a conversion operator?
- ~ How can I handle a constructor that fails?
- ~ How can I handle a destructor that fails?
- ~ Can a copy constructor accept an object of the same class as parameter, instead of reference of the object?
- ~ What's the order that local objects are destructed?
- ~ What's the order that objects in an array are destructed?
- ~ Can I overload the destructor for my class?
- ~ Should I explicitly call a destructor on a local variable?
- ~ What if I want a local to "die" before the close `}` of the scope in which it was created? Can I call a destructor on a local if I really want to?
- ~ OK, OK already; I won't explicitly call the destructor of a local, but how do I handle the above situation?
- ~ What if I can't wrap the local in an artificial block?
- ~ But can I explicitly call a destructor if I've allocated my object with `new`?
- ~ What is "placement new", and why would I use it?
- ~ When I write a destructor, do I need to explicitly call the destructors for my member objects?
- ~ When I write a derived class's destructor, do I need to explicitly call the destructor for my base class?
- ~ Is there any difference between `List x;` and `List x();`?
- ~ Can one constructor of a class call another constructor of the same class to initialize this object?
- ~ Is the default constructor for Fred always `Fred::Fred()`?
- ~ Which constructor gets called when I create an array of Fred objects?
- ~ Should my constructors use "initialization lists" or "assignment"?
- ~ Should you use this pointer in the constructor?
- ~ What is the "Named Constructor Idiom"?

=> Polymorphism & Virtual Function :

- ~ What is Polymorphism?
- ~ What is the problem with Runtime type identification?
- ~ What is virtual function?
- ~ What is a "pure virtual" member function?
- ~ How are virtual functions implemented in C++?
- ~ What is pure virtual function? or what is an abstract class?
- ~ How Virtual functions call up is maintained?
- ~ What is a virtual destructor?

=> Inheritance :

- ~ What is inheritance?
- ~ When should you use multiple inheritance?
- ~ Explain the ISA and HASA class relationships. How would you implement each in a class design?
- ~ When is a template a better solution than a base class?
- ~ What is multiple inheritance (virtual inheritance)? What are its advantages and disadvantages?
- ~ What a derived class inherits or doesn't inherit?

# Data Structures and Algorithms Live Class

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/Data-Structures-and-Algorithms-Live-Class>

## Course Description :-

The Data Structure and Algorithm program focused on learning algorithmic strategies for addressing a myriad of challenges while having complete control of memory and time. Develop a thorough understanding of how data structures work and how to create efficient algorithms.

## Course Features :-

- => Online live classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Problem solving
- => Analytical skill
- => Design Solution
- => Architecture design
- => Answer Confidently in interview.
- => Upscale your skill as a Developer.

## Requirements :-

- => Understanding of python programming language
- => A system with a decent internet connection
- => Your dedication

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

=> INTRODUCTION :

~ Course Overview

=> ANALYSIS IN ALGORITHMS :

- ~ Introduction to Algorithms
- ~ Analyzing Algorithms
- ~ Asymptotic Notations - Big O, Theta and Omega Notations

=> RECURRENCE RELATION :

- ~ Introduction to Recurrence Relation Solving
- ~ Substitution Method - Problem 1
- ~ Substitution Method - Problem 2
- ~ Substitution Method - Problem 3
- ~ Substitution Method - Problem 4
- ~ Recursive Tree Method - Problem 1
- ~ Recursive Tree Method - Problem 2
- ~ Recursive Tree Method - Problem 3
- ~ Master's Theorem - Case 1
- ~ Master's Theorem - Case 2
- ~ Master's Theorem - Case 3

=> ARRAY DATA STRUCTURE :

- ~ Introduction to Arrays
- ~ One Dimensional Array - How to find the address of an Element
- ~ Two Dimensional Array - Row major order and column major order
- ~ Searching Algorithm - Linear search in an Array
- ~ Comparison Sort in an Array - Selection sort
- ~ Comparison Sort in an Array - Bubble sort

- ~ Comparison Sort in an Array - Insertion sort
- ~ Non-Comparison Sort in an Array - Count sort
- ~ Non-Comparison Sort in an Array - Radix sort
- ~ Non-Comparison Sort in an Array - Bucket sort
- ~ Interview-Based Problem Statement - Missing Number in an array
- ~ Solution Discussed - Missing Number in an array
- ~ Interview-Based Problem Statement and Brute Force Approach - Divide two Integer without division operator
- ~ Solution Discussed - Optimised Approach with Complexity Analysis

#### => DIVIDE AND CONQUER :

- ~ Introduction to Divide and Conquer

#### => DISCUSSIONS OF APPLICATIONS AND CONQUER :

- ~ Binary Search in an Array
- ~ Coding Implementation of Binary Search
- ~ Finding of Power of an Element
- ~ Coding Implementation of Power Of an Element
- ~ Inplace and Outplace Sorting Algorithm
- ~ Merge Sort Recursive Tree
- ~ Merge Sort Recursive Tree Continue
- ~ Max and Min Comparison in Merge Procedure
- ~ Code Implementation of Merge Sort
- ~ Finding of Maxima and Minima
- ~ Strassen's Matrix Multiplication
- ~ Finding Of Number of Inversions

#### => QUICKSORT :

- ~ Introduction To QuickSort
- ~ Partition Algorithm in QuickSort
- ~ Psuedo Implementation of QuickSort
- ~ Recurrence Relation Analysis with time complexity finding
- ~ Coding Implementation of QuickSort Algorithm
- ~ Problem 1 Based on QuickSort Algorithm
- ~ Solution 1 Based on QuickSort Algorithm
- ~ Problem 2 Based on QuickSort Algorithm
- ~ Solution 2 Based on QuickSort Algorithm
- ~ Randomized QuickSort Algorithm

#### => SELECTION PROCEDURE :

- ~ Introduction To Selection Procedure and Pseudocode
- ~ Recurrence Relation Analysis with time complexity finding Of Selection Procedure
- ~ Code Implementation of Selection Procedure

#### => LINKED LIST :

- ~ Introduction to Linked List
- ~ Insertion of Node(Beginning and End Position) in Linked List
- ~ Insertion of Node(Any Position) in Linked List
- ~ Deletion of Node in Linked List
- ~ Searching of Node from Linked List
- ~ Reversal of Nodes in Linked List
- ~ Floyd's Cycle Detection Algorithm
- ~ Doubly Linked List
- ~ Circular Linked List

#### => SKIP LIST(ADVANCE DSA) :

- ~ Introduction to Skip List
- ~ Build-in Skip List
- ~ Search in Skip List
- ~ Insertion in Skip List
- ~ Deletion in Skip List
- ~ Complexity Analysis

#### => STACK AND QUEUE DATA STRUCTURE :

- ~ Introduction to Stack and Queue Data Structure
- ~ Implementation of Stack and Queue using Array in Python
- ~ Implementation of Stack and Queue using Collection.deque in Python
- ~ Interview-Based Coding question

#### => HASHING DATA STRUCTURE :

- ~ Introduction to Hashing Data Structure
- ~ Hash Function and its types

#### => COLLISION RESOLUTION TECHNIQUES :

- ~ Chaining

#### => OPEN ADDRESSING :

- ~ Linear Probing
- ~ Quadratic Probing
- ~ Double Hashing
- ~ Perfect Hashing
- ~ Consistent Hashing
- ~ Interview-Based Coding Question - Two Sum Problem
- ~ Bloom Filters

#### => TREE DATA STRUCTURE :

- ~ Introduction to Binary Tree
- ~ Complete Binary Tree and Almost Complete Binary Tree
- ~ Full Binary Tree and Representation using Array and Linked List
- ~ Interview-Based Coding Question - Symmetric Tree Or Not

#### => BINARY SEARCH TREE :

- ~ Introduction

- ~ Insertion
- ~ Inorder Traversal in BST Gives Sorted Array Concept
- ~ Searching
- ~ Coding Implementation of Searching Operation
- ~ Deletion
- ~ Deletion Implementation
- ~ Standard Formula to count the number of possible BSTs given number of nodes
- ~ Interview-Based Coding question - Catalan Number Concept to find number of BST

#### => HEIGHT BALANCED TREE: AVL TREE :

- ~ Introduction to AVL Tree
- ~ Insertion
- ~ Insertion Demonstration and Searching in AVL Tree
- ~ Deletion

#### => HEIGHT BALANCED TREE: RED BLACK TREE :

- ~ Introduction: Why Red Black Tree?
- ~ Properties Of Red Black Tree
- ~ Insertion Rules in Red Black Tree
- ~ Example Demonstration 1 of Insertion in Red Black Tree
- ~ Example Demonstration 2 of Insertion in Red Black Tree Part 1
- ~ Example Demonstration 2 of Insertion in Red Black Tree Part 2
- ~ Example Demonstration 2 of Insertion in Red Black Tree Part 3
- ~ Searching
- ~ Deletion Rules in Red Black Tree
- ~ Example Demonstration of Deletion in Red Black Tree

#### => B AND B+ TREE: USAGE IN DATABASES :

- ~ Insertion
- ~ Searching
- ~ Deletion

#### => GRAPH TRAVERSAL ALGORITHMS :

- ~ Introduction to Graph Traversal Algorithms
- ~ Introduction to Depth First Search Algorithm
- ~ Depth First Search with Example illustration
- ~ DFS Pseudocode and illustration using Example
- ~ DFS Coding Implementation and Complexity Analysis
- ~ Breadth-First Search with Example illustration
- ~ Level Order Traversal Using BFS
- ~ BFS Pseudocode and coding implementation with complexity analysis
- ~ Interview-Based Coding Question - Binary Tree Zigzag Level Order Traversal

#### => HEAP DATA STRUCTURE :

- ~ Introduction to Heap Data Structure
- ~ Maxheap and Minheap Overview
- ~ Insertion in Minheap
- ~ Example Demonstration of Insertion in Minheap
- ~ Deletion in Minheap
- ~ Creation in Minheap Part1
- ~ Creation in Minheap Part2
- ~ Mathematical Derivation to analyse the complexity of creation in minheap
- ~ Interview-Based Coding Question - Maximum Product of three numbers in an array
- ~ Interview-Based Coding Question - Finding of K closest Points from an origin
- ~ HeapSort Algorithm with Time complexity analysis
- ~ Pseudocode of HeapSort and Why HeapSort is not stable algorithm?

#### => TREE TRAVERSAL :

- ~ Introduction to Tree Traversal
- ~ Inorder Traversal
- ~ Preorder Traversal
- ~ Postorder Traversal
- ~ Questions Based on Above Traversal Algorithms

#### => GREEDY ALGORITHMS :

- ~ Introduction to Greedy Algorithms

#### => DISCUSSION AND APPLICATIONS OF GREEDY :

- ~ Fractional Knapsack Problem
- ~ Pseudocode Of Fractional Knapsack Problem
- ~ Implementation Of Fractional Knapsack Problem

#### => MINIMUM SPANNING TREE :

- ~ Introduction to Basics Of Graphs
- ~ Null Graph and Complete Graph
- ~ Introduction To Spanning Tree Algorithm
- ~ Concept Of Minimum Spanning Tree
- ~ Kruskal Algorithm
- ~ Time Complexity Of Kruskal Algorithm
- ~ Prim's Algorithm
- ~ Decrease Key Operation in MinHeap
- ~ Time Complexity Of Prim's Algorithm

#### => SINGLE SOURCE SHORTEST PATH :

- ~ Introduction to Single Source Shortest Path
- ~ Dijkstra's Algorithms
- ~ Time Complexity Of Dijkstra's Algorithm
- ~ Implementation of Dijkstra's Algorithm
- ~ Introduction to Huffman Coding
- ~ Algorithm and Time Complexity Of Huffman Coding
- ~ Optimal Merge Pattern Algorithm and Time Complexity Analysis
- ~ Job Sequencing with Deadline

- ~ *Implementation Of Job Sequencing with Deadline*
- ~ *Bellman-Ford Algorithm*

=> DYNAMIC PROGRAMMING :

- ~ *Introduction to Dynamic Programming*
- ~ *Overlapping Subproblem in Dynamic Programming*
- ~ *Tabulation in Dynamic Programming*
- ~ *Memoization in Dynamic Programming*

=> DISCUSSION AND APPLICATIONS OF DYNAMIC PROGRAMMING :

- ~ *Fibonacci Series*
- ~ *Longest Common Subsequence*
- ~ *0/1 Knapsack Problem*
- ~ *Sum of Subset*

=> ALL PAIR SHORTEST PATH :

- ~ *Floyd Warshall Algorithm and Complexity Analysis*

=> STRING MATCHING ALGORITHMS :

- ~ *Introduction*
- ~ *Naive String Matching Algorithms*
- ~ *Rabin Karp Algorithm*
- ~ *KnuthMorrisPratt (KMP) Pattern Matching*

=> NP-HARD AND NP-COMPLETE PROBLEM :

- ~ *NP-Hard*
- ~ *NP-Complete Problem*

# Data Manipulation using Pandas

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Data-Manipulation-using-Pandas>

## Course Description :-

This course will teach students the fundamentals of data analysis using the highly popular Pandas library in Python programming. The course will cover data manipulation and cleaning techniques using the popular Python pandas data science library, as well as the abstraction of Series and DataFrame as central data structures for data analysis, as well as tutorials on how to effectively use functions like groupby, merge, and pivot tables. Students will be able to take tabular data, clean it, alter it, and execute basic inferential statistical analyses at the conclusion of this course.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to Pandas
- => Basic Data structure of Pandas
- => Pandas Series
- => Pandas DataFrame
- => Pandas Operation
- => Pandas groupby
- => Data Operation
- => Plotting in Pandas
- => Advance Pandas
- => Project

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ *Pandas Intro*
  - ~ *Course prerequisite*
  - ~ *What is Pandas?*
  - ~ *Why do we use Pandas?*
  - ~ *What can Pandas do?*
  - ~ *Advantages of Pandas*
  - ~ *Installation of Pandas*
- => Assignment 1 :
  - ~ *Write a command to check the version of Pandas?*
- => Pandas Series :
  - ~ *What is Pandas Series?*
  - ~ *Index*
  - ~ *Labels*
  - ~ *Creating own Labels*
- => Assignment 2 :
  - ~ *Create a simple Pandas Series from a list.*



=> Pandas Read CSV :

- ~ What is a CSV file?
- ~ What is `pandas.read_csv` function?
- ~ What is `pandas.head` function?
- ~ What is `pandas.tail` function?

=> Pandas read JSON :

- ~ What is a JSON file?
- ~ What is `pandas.read_json` function?

=> Pandas DataFrame :

- ~ What is Pandas DataFrame?
- ~ How to create Pandas DataFrame?
- ~ How to create Pandas DataFrame from the list?
- ~ How to create Pandas DataFrame from the dictionary?
- ~ Selecting multiple columns
- ~ Selecting Columns with methods
- ~ DataFrame Attributes and methods
- ~ Meaningful Indexing
- ~ Renaming row and column names
- ~ Locate row
- ~ Creating and deleting columns

=> Assignment 3 :

- ~ Read any CSV data file and show the first five results of the file.

=> Assignment 4 :

- ~ Create pandas DataFrame from Dictionary

=> Selecting Subset of DataFrame :

- ~ Selecting subset of data with `[]`
- ~ Selecting subset of data with `.loc`
- ~ Selecting subset of data with `.iloc`

=> Boolean Indexing :

- ~ What is Boolean values?
- ~ What is Boolean indexing?
- ~ What is a comparison operator?
- ~ Boolean selection with criteria
- ~ Condition expressions
- ~ `.isin` function
- ~ `.isnull` function

=> Assignment 5 :

- ~ Read any CSV file and check whether there are null values or not in file.

=> Assigning Subset of data :

- ~ Assigning a new column
- ~ Assigning a new column with a list or array
- ~ What is a datatype?
- ~ Changing the datatype with `astype` function
- ~ Assigning a subset with `.iloc`
- ~ Assigning an entire column with `.loc` and `.iloc`

=> Pandas Groupby :

- ~ What is Pandas Groupby?
- ~ Why do we use the Groupby function?
- ~ What are different aggregation functions?
- ~ How to use Groupby?

=> Advance Pandas :

- ~ Sorting `nlargest`, `nsmallest`, `sort_values`
- ~ Replacing values in DataFrame/Series
- ~ Renaming columns and indexes in DataFrame/Series
- ~ `fillna` function
- ~ `dropna` function
- ~ Descriptive Statistics
- ~ Combining DataFrames
- ~ Merge and Join function

=> Plotting in Pandas :

- ~ What is Visualization?
- ~ What are the different visual segments?
- ~ What are the different plots that are available in Pandas?

=> Assignment 6 :

- ~ Create any dummy data and plot a graph using that data.

=> Project :

- ~ Use a dataset and do data analysis using pandas and find insights from the data.

# Big Data Masters Tech Neuron

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA MASTERS

Course link : <https://ineuron.ai/course/Big-Data-Masters-Tech-Neuron>

## Course Description :-

This unique industry program will help to learn the entire stack of Big Data and be ready to crack jobs in leading organizations.

## Course Features :-

- => Full stack Data Science masters certification
- => One year of internship Anytime
- => Online Instructor-led learning: Live teaching by instructors
- => 20 + hands-on industry real-time projects.
- => 200 hours live interactive classes.
- => Lifetime Dashboard access
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Interview Preparation Anytime
- => Regular assessment

## What you will learn :-

- => Big Data
- => Hadoop
- => HDFS
- => YARN
- => Linux
- => AWS EC2
- => AWS IAM
- => AWS S3
- => AWS SNS
- => AWS DMS
- => AWS RDS
- => AWS Redshift
- => Hbase
- => Sqoop
- => Confluent
- => Atlas
- => Ambari
- => Databricks

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Shashank Mishra :

~ Experienced Data Engineer with a demonstrated history of working in service and product companies. Solved data mysteries for different domains like Aviation, Pharmaceutical, FinTech, Telecom and Employee Services. Have designed scalable & optimized data pipelines to handle PetaBytes of data, with Batch & Real Time frequency. Got good exposure on different BigData frameworks (Hadoop, Spark, Hive, Sqoop, Flume, Flink, Kafka, Docker), Databases (MySQL, HBase, Cassandra, Redshift, Elastic Search), AWS Services (S3, Lambda, EMR, Glue, Cloudwatch, Redshift, SNS, SQS, Athena, Appflow), Dashboarding Tools (Grafana, Kibana, QuickSight, DataDog, Data Studio), Monitoring Tools (Airflow, Azkaban), Web Development (HTML, CSS, Scala Play, Django, Rest API, JavaScript, Ajax, JQuery), Good command over programming languages (Python, Java, Scala, Shell Scripting) and strong Data Structures & Algorithm fundamentals.

## Curriculum details :-

=> Big Data Introduction: Introduction :

- ~ What is Big Data?
- ~ Evolution of Big Data
- ~ Why to learn Big Data technologies?
- ~ Examples of Big Data
- ~ Who is using Big Data?
- ~ Why is Data so important?
- ~ Characteristics of Big Data
- ~ Challenges of Big Data
- ~ Data scale
- ~ Manage, store and process Big Data
- ~ 5 Vs of Big Data
- ~ Sources of Data flood
- ~ Exploding data problem
- ~ OLTP and OLAP
- ~ Operational vs Analytical Big Data
- ~ Possible solutions: scaling up vs. scaling out
- ~ Challenges of scaling up and scaling out

=> Hadoop fundamentals :

- ~ What is Hadoop?
- ~ Hadoop in layman's term
- ~ History and timeline of Hadoop
- ~ Evolutionary features of Hadoop
- ~ Why hadoop in demand?
- ~ Components of Hadoop ecosystem
- ~ Hadoop architecture
- ~ How hadoop solve data explosion problem?
- ~ Differences between Hadoop 1.X and Hadoop 2.X and Hadoop 3.X
- ~ Hadoop 1.x 2.x 3.x architecture, components and working of those Components

=> HDFS :

- ~ Design of HDFS
- ~ HDFS architecture
- ~ HDFS features
- ~ Name node and data node
- ~ Secondary name node
- ~ Job tracker
- ~ Task tracker
- ~ Client nodes
- ~ Explain master-slaves
- ~ Pseudo-distributed
- ~ Fully-distributed
- ~ Data replication
- ~ How does a file read and write work?
- ~ Local file system and HDFS
- ~ Rack awareness
- ~ Arrangement of racks
- ~ Arrangement of machines and racks
- ~ Checkpointing in Hadoop
- ~ Benefits of replica placement and rack awareness
- ~ URL And URN
- ~ HDFS commands
- ~ HDFS web interface
- ~ Fault tolerance
- ~ Name node failure management
- ~ Anatomy of file read and write from HDFS
- ~ Important java classes to write data to HDFS
- ~ Inputsplit and data blocks difference
- ~ Why Is the block size 128 MB?
- ~ Recordreader
- ~ Inputformat
- ~ Default Inputformat:TextInputformat
- ~ Outputformat
- ~ What is partitioner?
- ~ Using partitioner
- ~ Map only job
- ~ Flow of operations in MapReduce
- ~ Serialization in MapReduce

=> HDFS Operations :

- ~ Start HDFS
- ~ Listing files in HDFS
- ~ Writing a file into HDFS
- ~ Reading data from HDFS
- ~ Shutting down HDFS
- ~ Listing contents of directory
- ~ Displaying and printing disk usage
- ~ Moving files & directories
- ~ Copying files and directories
- ~ Displaying file contents

=> YARN :

- ~ What is Yarn?
- ~ Why Yarn?
- ~ Classic MapReduce v/s Yarn
- ~ Yarn architecture

- ~ Resource Manager
- ~ Node manager
- ~ Application master
- ~ Node manager containers
- ~ Resource manager components
- ~ Advantages & disadvantages of Yarn
- ~ Yarn applications
- ~ Scheduling in Yarn
- ~ Fair Scheduler
- ~ Fault Tolerance
- ~ Schedulers in Yarn
- ~ FIFO scheduler
- ~ Capacity scheduler
- ~ Fair scheduler

=> Setting up Our Linux Space :

- ~ Downloading necessary tools
- ~ Installing Ubuntu in Windows
- ~ What is SSH?
- ~ Install SSH Clients
- ~ Setting up SSH in Ubuntu VM
- ~ How to do SSH to your Ubuntu VM?
- ~ Setting Up Passwordless SSH

=> AWS EC2 :

- ~ Launch a Basic EC2 Instance
- ~ Different Types of instances - Reserved, On-Demand, Spot, Dedicated
- ~ Different configurations of EC2 machines
- ~ Attaching detaching of EBS Volume in EC2
- ~ Practising few commands on EC2

=> AWS IAM :

- ~ The Mechanics behind IAM
- ~ Managing IAM Users
- ~ IAM Administration (Guide) (Listing, Deleting Users & Accounts)
- ~ Managing Permissions for IAM Users
- ~ Changing IAM User Permissions
- ~ Creating and Administering IAM Groups
- ~ Creating and Administering IAM Group Policies
- ~ Assigning Preset and Custom Group Policies

=> AWS S3 :

- ~ Buckets
- ~ Objects
- ~ Upload, Delete Files
- ~ Data Encryptions
- ~ Pricing & Data Limitation on S3
- ~ S3 Versioning
- ~ Version ID
- ~ Bucket policy
- ~ Notifications from S3
- ~ Work with S3 using AWS CLI
- ~ AWS Lambda
- ~ What is AWS Lambda and Why it is needed?
- ~ Features & Limitations of Lambda
- ~ Hello world program using Lambda
- ~ Auto trigger Lambda Function based on S3 file upload notification
- ~ Access other services from Lambda
- ~ AWS Secret Manager
- ~ Create and Maintain secrets
- ~ Accessing credentials from Secret Manager using Boto3

=> AWS EMR :

- ~ Setting up EMR Cluster
- ~ Install Spark, Hive, Hadoop
- ~ Resource types in EMR cluster
- ~ Data Processing on EMR Cluster
- ~ AWS Glue
- ~ Setting up cluster in Glue
- ~ Properties of Glue
- ~ Creating Catalogs in Glue
- ~ Read partitioned Data
- ~ Bulk and Incremental data processing from S3 in Glue
- ~ Data Processing in Glue
- ~ Glue jobs and Triggers

=> AWS SNS :

- ~ What is SNS?
- ~ How SNS works?
- ~ Creating SNS Topics and subscribing
- ~ Different types of subscribers
- ~ Sending notifications via SNS
- ~ AWS SQS
- ~ What is SQS?
- ~ Different types of SQS?
- ~ At-Least once and Exactly once delivery via SQS
- ~ Ingesting data to SQS
- ~ Inflight messages
- ~ Consume data from SQS
- ~ Dead Letter Queue

=> AWS DMS :

- ~ What is DMS?
- ~ Capturing CDC event in DMS where Database as a source
- ~ Capture CDC events and sending it to downstream systems
- ~ AWS Kinesis
- ~ Creating Kinesis Streams
- ~ Ingesting real time data in Kafka Streams
- ~ Consume real time data from Kafka Streams

=> AWS RDS :

- ~ MySQL Database using AWS RDS
- ~ Scalability & Limitations of AWS RDS
- ~ Creating tables and loading data in AWS RDS
- ~ Querying data from RDS
- ~ AWS Athena
- ~ What is serverless database services
- ~ Athena vs RDS
- ~ Table metadata in Athena for the data residing in S3
- ~ Creating table for S3 data
- ~ Querying S3 data using Athena

=> AWS Redshift :

- ~ What is Data warehousing services?
- ~ Architecture of Redshift
- ~ Resources types in Redshift Cluster
- ~ Creating tables in Redshift
- ~ Internal & External tables
- ~ Partitioning, Sort Keys, Column compression
- ~ Querying data in Redshift
- ~ Views & Materialized views in Redshift

=> AWS Dynamo :

- ~ Architecture of DynamoDB
- ~ Creating tables and Ingesting data into DynamoDB table
- ~ Querying data from DynamoDB
- ~ AWS Cloudwatch
- ~ Cron based triggers
- ~ Event pattern based triggers
- ~ Monitoring & Alerting using Cloudwatch
- ~ AWS QuickSight
- ~ Creating business dashboards using Quick sight

=> Introduction :

- ~ What is Hive?
- ~ Hive Vs Map Reduce
- ~ Hive Vs Relational databases
- ~ Installation and setup of Hive
- ~ Introduction to CouchDB
- ~ Why CouchDB?
- ~ History of CouchDB
- ~ Features of CouchDB
- ~ Advantages of CouchDB
- ~ Disadvantages of CouchDB
- ~ What is Neo4j?
- ~ Why Neo4j?
- ~ Features of Neo4j
- ~ Advantages of Neo4j
- ~ Neo4j Architecture
- ~ Applications of Neo4j
- ~ Data model of Neo4j
- ~ Building Blocks of Neo4j

=> Hive Architecture :

- ~ Hive architecture
- ~ Different modes of Hive
- ~ Hive Functions: Built-in & UDF
- ~ Datatypes in Hive
- ~ Operators in Hive
- ~ How to create and drop databases?
- ~ Hive create table: internal table, external table , alter, drop

=> DDL and DML commands in Hive :

- ~ Hive DDL
- ~ Create
- ~ Show
- ~ Describe
- ~ Use
- ~ Drop
- ~ Alter
- ~ Truncate
- ~ Hive DML
- ~ Load
- ~ Select
- ~ Insert
- ~ Delete
- ~ Update
- ~ Export
- ~ Import
- ~ Hive view and index
- ~ What is Hive metastore?

- ~ How to install and configure Hive metastore?
- ~ What is Hive data modeling?

=> Hive partitioning and bucketing :

- ~ Partitioning in Hive
- ~ Static and dynamic partitioning
- ~ Bucketing in Hive
- ~ Bucketing vs Partitioning
- ~ What is Hive query language(HQL)?

=> HQL language :

- ~ HiveQL- Where
- ~ HiveQL- Order By
- ~ HiveQL- Group By
- ~ HiveQL- Joins and types
- ~ HiveQL- SubQuery
- ~ Hive ETL: loading JSON, XML, text data
- ~ Working with arrays
- ~ Sort by and order by
- ~ Distribute by and cluster by
- ~ Bucket-map join
- ~ Sort-Merge-Bucket-Map join
- ~ Left semi join

=> Different File formats in Hive :

- ~ File formats in Hive
- ~ Text files
- ~ Input formats in Hive
- ~ Sequence files in Hive
- ~ RC file in Hive
- ~ Sequencefile
- ~ ORC files in Hive
- ~ Avro files
- ~ Parquet file
- ~ Inline index in ORC files
- ~ ORC file configurations in Hive
- ~ SerDe in Hive
- ~ Demo: CSVSerDe
- ~ JSONSerDe
- ~ RegexSerDe
- ~ Analytic and windowing in Hive
- ~ Demo: analytics.hql
- ~ Hcatalog in Hive
- ~ Demo: using\_HCatalog
- ~ Accessing Hive with JDBC
- ~ Demo: HiveQueries.Java
- ~ HiveServer2 and beeline
- ~ Demo: beeline
- ~ Demo: ToUpper.Java and working\_with\_UDF
- ~ Optimizations in Hive
- ~ Demo: Optimizations

=> Introduction of HBase :

- ~ What is HBase?
- ~ HDFS and HBase
- ~ HBase vs RDBMS
- ~ HBase vs HIVE
- ~ HBase storage mechanism
- ~ Feature of HBase
- ~ Applications of HBase

=> HBase installation setup :

- ~ Apache HBase setup
- ~ Hardware recommendations
- ~ Software recommendations
- ~ Installation using cloudera manager
- ~ Basic static configuration

=> HBase architecture :

- ~ Architecture of HBase
- ~ Components of HBase architecture
- ~ Client library
- ~ Zookeeper
- ~ HMaster server
- ~ HBase regions servers

=> HBase commands :

- ~ General commands
- ~ status
- ~ table\_help
- ~ version
- ~ whoami
- ~ Data definition commands
- ~ alter
- ~ alter\_async
- ~ alter\_status
- ~ create
- ~ drop
- ~ drop\_all
- ~ enable
- ~ enable\_all

- ~ exists
- ~ get\_table
- ~ is\_disabled
- ~ is\_enabled
- ~ show\_filters
- ~ Data manipulation commands
- ~ append
- ~ count
- ~ delete
- ~ deleteall
- ~ get\_table
- ~ get\_counter
- ~ put
- ~ truncate
- ~ truncate\_preserve
- ~ Other HBase shell commands
- ~ Admin commands
- ~ Replication commands
- ~ Snapshot commands
- ~ Visibility labels commands
- ~ Security commands

=> CRUD operations using HBase shell :

- ~ What is HBase shell?
- ~ HBase shell usage
- ~ Starting HBase shell
- ~ Creating table
- ~ Inserting a row
- ~ Updating a row
- ~ Retrieving a row
- ~ Retrieving a range of rows
- ~ Deleting a row
- ~ Deleting a table
- ~ Retrieve rows within a time range
- ~ Filter by column value - SingleColumnValueFilter
- ~ Filter by Row id - RowFilter
- ~ Apply multiple conditions - Filterlist

=> Understanding the troubleshooting in HBase :

- ~ Understand the troubleshooting
- ~ Trouble shooting distributed clusters
- ~ Administration from the command line
- ~ How to use the HBase UI?
- ~ How to use the Metrics and the logs?

=> Basic Introduction :

- ~ Challenges with traditional RDBMS
- ~ What is Nosql database?
- ~ History behind the creation of Nosql databases
- ~ Features of Nosql database
- ~ Different types of Nosql databases
- ~ When Nosql should be used?
- ~ Advantages of Nosql
- ~ Disadvantages of Nosql
- ~ Why Nosql database?

=> Introduction and overview of cassandra :

- ~ What is Apache Cassandra?
- ~ History of Cassandra
- ~ Cassandra Database vs Relational Database
- ~ Apache Cassandra features
- ~ Cassandra use cases and applications
- ~ Advantages of Cassandra
- ~ Disadvantages of Cassandra

=> Setup, installation and configuration :

- ~ Cassandra configuration with datastax
- ~ Understanding different ways to communicate with cassandra
- ~ Using cqlsh

=> Cassandra Architecture :

- ~ Cassandra architecture
- ~ Cassandra data model
- ~ Cassandra as a distributed database
- ~ Node
- ~ Data center
- ~ Cluster
- ~ Commit log
- ~ Mem-table
- ~ SSTable
- ~ Data replication
- ~ Write operation
- ~ Read operation
- ~ Data compaction

=> Cassandra Data Modeling :

- ~ Data modeling basics
- ~ Cassandra data modeling
- ~ Cassandra column types
- ~ Cassandra keyspace

=> Cassandra cluster and node :

- ~ *Configure and managing a cluster*
- ~ *Cluster and nodes*
- ~ *Adding nodes to cluster*
- ~ *Monitoring a cluster*
- ~ *Repairing a nodes*
- ~ *Removing a node*

=> Cassandra - Shell Commands :

- ~ *Help*
- ~ *Capture*
- ~ *Consistency*
- ~ *Copy*
- ~ *Describe tabel*
- ~ *Describe keyspaces*
- ~ *Expand*
- ~ *Exit*
- ~ *Show*
- ~ *Source*

=> Cassandra Query Language(CQL) :

- ~ *CQL Data Definition Commands*
- ~ *Cassandra CQL Data Types*
- ~ *Creating Database*
- ~ *Creating Keyspace*
- ~ *Use Keyspace*
- ~ *Alter Keyspace*
- ~ *Drop Keyspace*
- ~ *Create Table*
- ~ *Alter table*
- ~ *Drop table*
- ~ *Truncate*
- ~ *Create Index*
- ~ *Drop Index*
- ~ *CQL Data Manipulation Commands*
- ~ *Insert*
- ~ *Update*
- ~ *Delete*
- ~ *Batch*
- ~ *CQL Clauses*
- ~ *Select*
- ~ *Cassandra Where Clause*
- ~ *Cassandra Order by Clause*

=> Advanced CQL :

- ~ *CQL Collections*
- ~ *CQL User-Defined Types*
- ~ *Defining a Primary key*
- ~ *Defining a Partition key*
- ~ *Introduction to User-defined types(UDT)*
- ~ *How to Create a UDT?*
- ~ *UDT literals*
- ~ *How to alter a UDT?*
- ~ *How to drop a UDT?*

=> Cassandra CRUD Operation :

- ~ *Create data*
- ~ *Update data*
- ~ *Read data*
- ~ *Delete data*
- ~ *Maps*
- ~ *Sets*
- ~ *Lists*
- ~ *Key and indexing*

=> Introduction to MongoDB :

- ~ *Introduction*
- ~ *key charcristic of MongoDB*
- ~ *Understanding MongoDB ecosystem*
- ~ *Advantages & disadvantages of using MongoDB*

=> MongoDB installtion and setup :

- ~ *MongoDB installation in local*
- ~ *Setup MongoDB server*
- ~ *Setup MongoDB compass*
- ~ *Exploring the MongoDB compass*
- ~ *MondoDB local server and compass setup*
- ~ *MongoDB atlas setup*

=> Architecture :

- ~ *Architecture of MongoDB*
- ~ *Understanding databases, collections & documents*
- ~ *Creating databases & collections*
- ~ *Understanding JSON Data*
- ~ *Comparing JSON & BSON*
- ~ *Storage engines*
- ~ *Read path*
- ~ *Write path*
- ~ *Working set*
- ~ *Capped collection*



- ~ Oplog collection
- ~ TTL index
- ~ Gridfs

#### => CRUD operations :

- ~ MongoDB data types
- ~ Finding, Inserting, Deleting & Updating elements
- ~ Querying the documents
- ~ Bulk insert operations
- ~ Updating multiple document
- ~ Limiting documents
- ~ Understanding insertOne vs insertMany()
- ~ Updateone() vs updateMany()
- ~ Understanding find() & fetchall()
- ~ Understanding "deleteOne()" & "deleteMany()"
- ~ Filtering documents

#### => Schema design and data modeling :

- ~ Why do we use Schemas?
- ~ What is data modeling?
- ~ RDBMS and MongoDB data modeling difference
- ~ Embedding document
- ~ Reference document
- ~ Structuring documents
- ~ Understanding relations
- ~ One To One
- ~ One To Many
- ~ Many To Many

#### => Database administration in MongoDB :

- ~ Database status
- ~ Troubleshooting issues
- ~ Current operations
- ~ Rotating log files
- ~ Users and roles
- ~ Copy and clone database
- ~ DB and collection stats
- ~ Explain plan
- ~ Profiling
- ~ Changing configuration files
- ~ Upgrading the database

#### => MongoDB: backup and security :

- ~ Concept of backups
- ~ Mongoexport/mongoimport
- ~ Mongodump/mongorestore
- ~ Oplog backups
- ~ LVM backups
- ~ Backups using MMS/Ops manager
- ~ Purpose of security
- ~ Authentication and authorization
- ~ Role based access control

#### => Working with python driver :

- ~ Splitting work between the Driver & the Shell
- ~ Preparing our project
- ~ Installing Visual Studio Code or Pycharm
- ~ Installing the Python
- ~ Connecting Python & the MongoDB cluster
- ~ Storing products in the database
- ~ Fetching data from the database
- ~ Getting a single product
- ~ Editing & deleting products
- ~ Implementing pagination
- ~ Adding an index
- ~ Adding an index to make the Email unique
- ~ Adding user sign-in

#### => Replication in MongoDB :

- ~ Concept of replication
- ~ Replicaset member roles
- ~ Voting and electing primary
- ~ Role of oplog in replication
- ~ Read and write concern
- ~ Arbiter, Hidden and Delayed replica node
- ~ Priority settings
- ~ Replicaset nodes health check
- ~ Concept of resyncing the nodes
- ~ Rollbacks during failover
- ~ Keyfile authentication

#### => MongoDB scalability :

- ~ Concept of scalability
- ~ Sharding concept
- ~ Shardkey and chunks
- ~ Choosing shardkey
- ~ Sharding components
- ~ Types of sharding
- ~ Balanced data distribution
- ~ Sharded and non-sharded collection
- ~ Sharded replicaset

- ~ Tag aware sharding

## => MongoDB Monitoring :

- ~ MMS manager
- ~ Ops manager
- ~ MongoDB utility commands
- ~ MongoDB developer tools
- ~ MongoDB client drivers

## => CouchDB Architecture :

- ~ CouchDB engine
- ~ HTTP request
- ~ Document
- ~ Replica database

## => Graph Database in Neo4j :

- ~ What is Graph Database?
- ~ Why Graph Database?
- ~ Graph DB Data Model
- ~ Graph DB vs RDBMS
- ~ The Property Graph Model

## => Setup :

- ~ Environment setup for Neo4j
- ~ Installation of Neo4j on Windows
- ~ Installation of Neo4j on Linux
- ~ Installation of Neo4j on Mac
- ~ Exploring Neo4j Bloom

## => Neo4j CQL :

- ~ Introduction to Neo4j CQL
- ~ Neo4j CQL clauses
- ~ Neo4j CQL Functions
- ~ Neo4j CQL Data Types
- ~ Neo4j CQL operators
- ~ Neo4j CQL Boolean operators
- ~ Neo4j CQL Comparison operators
- ~ Node Creation in Neo4j CQL
- ~ Relationship creation in Neo4j CQL

## => Neo4j CQL Operators :

- ~ Neo4j CQL Operators
- ~ Comparison Operators
- ~ Boolean Operators
- ~ String Operators
- ~ List Operators
- ~ Regular Expression
- ~ String matching

## => Neo4j clauses :

- ~ Match Clause
- ~ Optional Match Clause
- ~ Where Clause
- ~ Count Function
- ~ Return Clause
- ~ Order by Clause
- ~ Limit Clause
- ~ Skip Clause

## => Neo4j CQL Clauses :

- ~ Neo4j CQL General clauses
- ~ Neo4j CQL Write clauses
- ~ Neo4j CQL Readclauses

## => Introduction to Kafka :

- ~ Introduction to Apache Kafka
- ~ History of Apache Kafka
- ~ Why Apache Kafka?
- ~ What is messaging system?
- ~ Kafka message flow
- ~ Committed vs uncommitted messages
- ~ Kafka operations
- ~ Kafka communication
- ~ Advantages of Kafka
- ~ Kafka use-cases

## => Architecture of kafka :

- ~ Kafka architecture

## => Installation of kafka :

- ~ Installation of Kafka in local system
- ~ kafka setup on cloud
- ~ Kafka - Confluent
- ~ Kafka - Confluent platform

## => Kafka CLI :

- ~ Introduction to Kafka CLI
- ~ Creating Kafka topic
- ~ Listing topics in Kafka CLI
- ~ Deleting topics in Kafka CLI
- ~ Getting details of Kafka topic
- ~ Producing data to Kafka topic

- ~ Consuming data to Kafka topic
- ~ Purging a Kafka topic

=> Zookeeper in Kafka :

- ~ Why Zookeeper is used in Kafka?
- ~ Role of Zookeeper in Kafka

=> Kafka APIs :

- ~ Introduction to Kafka API
- ~ Different types of Kafka API
- ~ Producer API
- ~ Consumer API
- ~ Streams API
- ~ Connector API
- ~ Kafka integration with Spark

=> Introduction to NiFi :

- ~ What is Apache NiFi?
- ~ Architecture of Apache NiFi
- ~ Characteristics of Apache NiFi
- ~ Advantages of Apache NiFi

=> Installation of Apache NiFi :

- ~ Environment Setup
- ~ Setting up Windows Developer Environment
- ~ Setting up Linux Developer Environment
- ~ Setting up Mac Developer Environment

=> Apache NiFi Repository :

- ~ Flowfile Repository
- ~ Content Repository
- ~ Provenance Repository

=> Apache NiFi User Interface :

- ~ Introduction to Apache NiFi User Interface
- ~ NiFi Canvas
- ~ NiFi Processors
- ~ Process Groups and Templates
- ~ Apache NiFi components

=> Apache NiFi Processors :

- ~ Introduction to Apache NiFi Processors
- ~ GenerateFlowFile
- ~ LogAttribute
- ~ Functionality of NiFi Processors

=> Data Flows and Content :

- ~ NiFi Properties and Settings
- ~ Data Flow Monitoring in NiFi

=> Processing files in Apache NiFi :

- ~ Processing of CSV file
- ~ Processing of JSON file
- ~ Processing of Text file

=> Getting started with Spark :

- ~ What is Spark and what it is purpose?
- ~ Why Spark is faster than Hadoop?
- ~ What is in-memory computation?
- ~ Features of Spark
- ~ Explain unified architecture of Spark
- ~ Components of the Spark unified architecture
- ~ Downloading and installing Spark standalone
- ~ Scala and Python overview, launching and using Sparks Scala and Python shell
- ~ Spark execution context
- ~ Driver
- ~ Executor
- ~ Master
- ~ Worker

=> The Resilient Distributed Datasets (RDD) :

- ~ Overview of RDD's
- ~ Features of RDD
- ~ RDD operations
- ~ RDD and pair RDDs and RDD performance
- ~ Flat maps and filters
- ~ Data loading in RDD
- ~ RDD deep dive
- ~ Partitions
- ~ Dependencies
- ~ Transformation in RDD
- ~ Action in RDD
- ~ Map
- ~ Filter
- ~ Filter map
- ~ Group by
- ~ Group by key
- ~ Reduce by key
- ~ Map partitions
- ~ Union
- ~ Join
- ~ Distinct

- ~ Coalesce
- ~ Key by
- ~ Partition by
- ~ Zip
- ~ Collect
- ~ Reduce by key
- ~ Aggregate
- ~ RDD Lineage
- ~ DAG for RDD
- ~ Limitations of Spark RDD
- ~ RDD persistence
- ~ Shared variables and broadcast variables
- ~ Accumulators

=> Spark SQL, DataFrames and Datasets :

- ~ Introducing Spark SQL
- ~ Introducing datasets and DataFrame
- ~ Data sources
- ~ Distributed SQL engine
- ~ Creating DataFrame
- ~ DataFrame operations
- ~ DataFrame from csv
- ~ DataFrame from db tables
- ~ DataFrame from hive NoSQL tabel
- ~ DataFrame from json
- ~ DataFrame from RDD
- ~ Different operations on DataFrame
- ~ Filter
- ~ Join
- ~ Group
- ~ Aggregation
- ~ Having
- ~ Where
- ~ User define function(UDF)
- ~ Grouping aggregation
- ~ Multiple grouping
- ~ More aggregation
- ~ Hash aggregation
- ~ Spark SQL vs RDD
- ~ Executing SQL commands and SQL-style functions on a DataFrame
- ~ Using DataFrames instead of RDD's
- ~ Different operations with dataframes with DataFrames
- ~ Word Count with DataFrames
- ~ DataFrames vs RDDs
- ~ Operations on DFs
- ~ Parquet files with Spark Sql Read, Write, Partitioning, Merging schema
- ~ ORC files
- ~ JSON files

=> Spark streaming :

- ~ Basic concepts of Spark Streaming
- ~ Linking
- ~ Initializing Streaming Context
- ~ Discretized Streams (DStreams)
- ~ Input DStreams and Receivers
- ~ Transformations on DStreams
- ~ Output operations on DStreams
- ~ DataFrame and SQL operations
- ~ MLlib operations
- ~ Caching / Persistence
- ~ Checkpointing
- ~ Accumulators, Broadcast Variables, and Checkpoints
- ~ Deploying applications
- ~ Performance tuning
- ~ Reducing the batch processing times
- ~ Setting the right batch interval
- ~ Memory tuning
- ~ Sliding window operations
- ~ Overview Spark Streaming and Structure Streaming and kafka streaming with kafka
- ~ Developing Spark Streaming applications Integration with Hbase
- ~ Kafka Twitter data setup
- ~ Writing Producer in Python
- ~ Writing Consumer in Python
- ~ Kafka Integration with Spark Streaming
- ~ Fault-tolerance semantics
- ~ Spark Cassandra

=> Spark Structure streaming :

- ~ Handling Event-time and Late Data
- ~ API using Datasets and DataFrames
- ~ Creating streaming DataFrames and streaming Datasets
- ~ Input Sources
- ~ Schema inference and partition of streaming DataFrames/Datasets
- ~ Operations on streaming DataFrames/Datasets
- ~ Basic Operations - Selection, Projection, Aggregation
- ~ Window Operations on Event Time
- ~ Handling Late Data and Watermarking
- ~ Types of time windows
- ~ Join Operations

- ~ *Stream-static Joins*
- ~ *Stream-stream Joins*
- ~ *Inner Joins with optional Watermarking*
- ~ *Outer Joins with Watermarking*
- ~ *Semi Joins with Watermarking*
- ~ *Support matrix for joins in streaming queries*
- ~ *Streaming Deduplication*
- ~ *Policy for handling multiple watermarks*
- ~ *Arbitrary Stateful Operations*
- ~ *Unsupported Operations*
- ~ *Limitation of global watermark*
- ~ *State Store*
- ~ *HDFS state store provider*
- ~ *RocksDB state store implementation*
- ~ *State Store and task locality*
- ~ *Starting Streaming Queries*

=> **Launching on a clusters :**

- ~ *Spark standalone*
- ~ *Running Spark on Mesos*
- ~ *Running Spark on YARN*
- ~ *Running Spark on Kubernetes*
- ~ *The Spark Standalone Web UI*

=> **PySpark Installtion :**

- ~ *Installtion using PyPi*
- ~ *Pyspark setup in local*
- ~ *Pyspark setup with anaconda*
- ~ *Pyspark setup with pycharm*

=> **PySpark DataFrame :**

- ~ *DataFrame creation*
- ~ *Viewing data*
- ~ *Accessing data*
- ~ *Applying a function*
- ~ *Grouping data*
- ~ *Object creation*
- ~ *Missing data*
- ~ *Grouping*
- ~ *Plotting*

=> **Spark Mlib :**

- ~ *Overview of Mlib*
- ~ *What is Machine Learning?*
- ~ *Supervised learning*
- ~ *Unsupervised learning*
- ~ *Basic statistics*
- ~ *Classification algorithms*
- ~ *Regression algorithms*
- ~ *Clustering algorithms*
- ~ *Collaborative filtering*
- ~ *Frequent pattern mining*
- ~ *Featurization*
- ~ *Pipelines*
- ~ *Persistence*
- ~ *Spark ml for ml*
- ~ *Collect tranning data*
- ~ *Different proccessing technique*
- ~ *Supervised learning*
- ~ *Linear regression*
- ~ *Logistic regression*
- ~ *Decision tree navie bayes*
- ~ *Recommender system*
- ~ *End to end case study with real time dataset*

=> **GraphX :**

- ~ *Overview*
- ~ *Graph operations*
- ~ *Graph builders*
- ~ *Graph algorithms*

=> **Spark configuration, monitoring and tuning :**

- ~ *Understand components of spark cluster*
- ~ *configure spark to modify the spark properties, environmental variables, or logging properties*
- ~ *Monitor Spark using the web UIs, metrics, and external instrumentation*

=> **Installation of Drill :**

- ~ *Installing Apache Drill for Mac and Linux*
- ~ *Running Drill In non-embedded and cluster mode*
- ~ *Overview of Drill web console*

=> **Connecting to Data sources :**

- ~ *Connecting to local file system*
- ~ *Understanding storage plugins and workspaces*
- ~ *Connecting to MySQL*
- ~ *Connecting to Mongo*
- ~ *Connecting to Kafka*
- ~ *Connecting to Hive*
- ~ *Connecting to HBase*
- ~ *Querying across data sources*

## => Introduction to Sqoop :

- ~ *Sqoop introduction*
- ~ *How Sqoop works?*
- ~ *Why we use Sqoop?*
- ~ *Features of Sqoop*

## => Sqoop Tools :

- ~ *Sqoop architecture and working*
- ~ *Using command aliases*
- ~ *Controlling the Hadoop installation*
- ~ *Using generic and specific arguments*
- ~ *Using options files to pass arguments*

## => Sqoop import :

- ~ *Purpose of Sqoop import*
- ~ *Connecting to a database server*
- ~ *Selecting the data to import*
- ~ *Free-form query imports*
- ~ *Controlling the import process*
- ~ *Controlling transaction isolation*
- ~ *Controlling type mapping*
- ~ *Incremental imports*
- ~ *File formats*
- ~ *Large objects*
- ~ *Importing data into Hive*
- ~ *Importing data into HBase*
- ~ *Importing data into Accumulo*
- ~ *Connecting to a Mainframe*

## => Sqoop export :

- ~ *Purpose of Sqoop export*
- ~ *Inserts vs Updates*
- ~ *Exports and Transactions*

## => Sqoop - Job :

- ~ *Create Job*
- ~ *Verify Job*
- ~ *Inspect Job*
- ~ *Execute Job*

## => Validation in Sqoop :

- ~ *Introduction to the validation*
- ~ *Purpose of Validation*
- ~ *Limitations of Validations*

## => Setup of Airflow :

- ~ *Components of Airflow*
- ~ *Installing Airflow on mac*
- ~ *Installing Airflow on linux*
- ~ *Installing Airflow on windows*
- ~ *Run Airflow locally*
- ~ *Introduction to the Airflow UI*
- ~ *What you need to know about the UI*
- ~ *Introduction to the Airflow CLI*

## => Core concepts of Airflow :

- ~ *What is DAG?*
- ~ *DAG skeleton*
- ~ *Default arguments*
- ~ *Instantiate a DAG*
- ~ *Jinja templating with Airflow*
- ~ *What are tasks?*
- ~ *What are operators?*
- ~ *How to setup dependencies?*
- ~ *What are hooks*
- ~ *What are executors*

## => Loading data to Data Warehouse :

- ~ *Set up*
- ~ *Connections*
- ~ *Load data from storage*
- ~ *Run SQL query*
- ~ *Use a hook to list storage objects.*
- ~ *Cross-Task communication (XComs)*
- ~ *Variables*

## => Advanced concepts in Airflow :

- ~ *Adios repetitive patterns*
- ~ *Minimising DAG's with SubDAG's*
- ~ *Adios SubDAG's, welcome taskgroups!*
- ~ *Sharing data between tasks with xcoms*
- ~ *Choosing a specific path in your DAG*
- ~ *How Tasks get triggered?*

## => Creating Plugins with Elasticsearch and PostgreSQL :

- ~ *Installation of Elasticsearch*
- ~ *How the plugin system works?*
- ~ *Creating a Hook interacting with Elasticsearch*

## => Testing Airflow DAGS's :

- ~ *Load test DAG's*

- ~ Unit test DAG's and operators
- ~ Unit test custom operators

#### => Docker Image for Apache Airflow :

- ~ Introduction to Docker
- ~ Why custom image?
- ~ How to build your own image?
- ~ Extending vs. customizing the image
- ~ Executors
- ~ Configure celery executors
- ~ Running Airflow on docker with celery executor
- ~ Configure local executors
- ~ Running Airflow on docker with local executor
- ~ Service level agreement
- ~ Security: Authentication, Roles, Encryption

#### => Monitoring Airflow :

- ~ Airflow monitoring with StatsD
- ~ Airflow monitoring with Prometheus
- ~ Airflow monitoring with Graphana
- ~ Error tracking with Sentry

#### => Introduction to Zookeeper :

- ~ Introduction of Apache Zookeeper
- ~ Why we need Zookeeper?
- ~ What is Distributed system?

#### => Internal structure :

- ~ Zookeeper Background
- ~ Architecture Diagram
- ~ Important Components

#### => Data models and Znodes :

- ~ Data model and Znode structure
- ~ What is Apache Zookeeper Znodes?
- ~ Sessions and watches

#### => Installation of Zookeeper :

- ~ Installation of Apache zookeeper
- ~ Configuration of Apache zookeeper
- ~ Starting Apache zookeeper server
- ~ CLI operations

#### => Monitoring in Zookeeper - Kafka :

- ~ Operating system
- ~ JMX monitoring

#### => Installation and setup oozie :

- ~ Installation of oozie on your machine

#### => Introduction to Ambari :

- ~ What is Apache Ambari?
- ~ Overview of Apache Ambari
- ~ History of Apache Ambari
- ~ Goals of Apache Ambari
- ~ Features of Apache Ambari
- ~ Benefits of Apache Ambari
- ~ Why should you learn Apache Ambari?
- ~ Apache Ambari architecture
- ~ Internal workflow of Ambari

#### => Core applications of Ambari :

- ~ Server
- ~ Agent
- ~ Web UI
- ~ Database

#### => Ambari usage :

- ~ Provisioning of Hadoop cluster
- ~ Monitoring of Hadoop cluster
- ~ Management of Hadoop cluster

#### => How is Ambari is different from Zookeeper? :

- ~ Basic task
- ~ Nature
- ~ Status maintenance

#### => Introduction to Cloud Databricks :

- ~ Introduction about cloud
- ~ Why cloud is important
- ~ Introduction to Databricks
- ~ Creating zure free account
- ~ Azure Databricks architecture overview

#### => Databricks clusters :

- ~ Overview of clusters
- ~ Azure Databricks cluster types
- ~ Azure Databricks cluster configuration
- ~ Creating Azure Databricks cluster
- ~ Azure Databricks cluster pool

#### => Mounting Data Lake container to databricks :

- ~ Overview

- ~ *Databricks file system (DBFS)*
- ~ *Databricks mount overview*
- ~ *Creating Azure data lake storage*
- ~ *Creating Azure service principal*
- ~ *Mounting Azure data lake Storage*
- ~ *Secret scopes overview*
- ~ *Creating secret scope In key vault*
- ~ *Mounting Data lake using secrets*

#### => Data ingestion - CSV files :

- ~ *Data ingestion overview*
- ~ *What is circuits file*
- ~ *Requirements*
- ~ *DataFrame reader*
- ~ *Select columns*
- ~ *DataFrame writer*

#### => Data ingestion - JSON files :

- ~ *What is JSON File?*
- ~ *Write data*

#### => Introduction to Atlas :

- ~ *What is Apache Atlas*
- ~ *Features of Apache Atlas*

#### => Installation Atlas :

- ~ *Installation of Apache Atlas*

#### => Atlas terminology :

- ~ *Relationships*
- ~ *Attributes*
- ~ *System specific typews*
- ~ *Data lineage*
- ~ *Classification*

#### => Atlas UI :

- ~ *Basic search in Atlas UI*
- ~ *Advanced search in Atlas UI*
- ~ *What is a glossary term?*
- ~ *Use cases of glossary*

#### => Introduction to Confluent :

- ~ *Overview of Confluent*
- ~ *Features of Confluent*

#### => Getting started with Confluent :

- ~ *Free trail for Confluent cloud*
- ~ *Quick start for Apache Kafka using confluent cloud*
- ~ *Confluent cloud console basics*
- ~ *KSQLDB in confluent cloud*
- ~ *Manage schemas on confluent cloud*
- ~ *REST API quick start for confluent cloud developers*

#### => Kafka Clusters :

- ~ *Features and limits by cluster type*
- ~ *Create a cluster with a console*
- ~ *Expand a dedicated cluster with console*
- ~ *Shrink a dedicated cluster with console*
- ~ *Cluster management API overview*
- ~ *Migrate topics on confluent cloud clusters*

#### => Manage topics in cloud console :

- ~ *Overview*
- ~ *Create, edit and delete topics*
- ~ *Use the message browser*

#### => Stream governance :

- ~ *Overview*
- ~ *Stream Lineage*
- ~ *Stream Catalog*
- ~ *Stream Quality*

#### => Cluster linking :

- ~ *Overview*
- ~ *Quick tutorial*
- ~ *Share Data Across Clusters, Regions And Clouds*
- ~ *Mirror topics*
- ~ *Data migration*

#### => Confluent platform with cloud :

- ~ *Confluent platform with cloud*
- ~ *Connecting Control Center To Confluent Cloud*
- ~ *Connect Clients to Confluent Cloud*
- ~ *Connecting Kafka to Confluent Cloud*
- ~ *Connecting Kafka Streams to Confluent Cloud*

#### => Confluent cloud API :

- ~ *Confluent Cloud API*
- ~ *Metrics API*

#### => Confluent CLI :

- ~ *Installing Confluent CLI*
- ~ *Configuring Confluent CLI*



~ *Manage Confluent*

## => Introduction to AWS :

- ~ *What is AWS?*
- ~ *AWS solutions for BigData?*
- ~ *What is Data ingestion?*

## => Cloud computing on AWS :

- ~ *What is cloud computing?*
- ~ *Cloud services by AWS*
- ~ *Cloud Computing Tools on AWS*
- ~ *Cloud Computing Tools Pricing*
- ~ *Introduction to AWS S3*
- ~ *Creating your First S3 bucket*
- ~ *Uploading an object to your Bucket*
- ~ *Download an object*
- ~ *Copy your object to a Folder*
- ~ *Delete your object and Bucket*

## => AWS Storage :

- ~ *Introduction to AWS storage*
- ~ *What is Simple storage Service (S3)?*
- ~ *How S3 works?*
- ~ *Use cases of S3*
- ~ *Storage Hierarchy in S3*
- ~ *Buckets in S3*
- ~ *S3 pricing*
- ~ *Creating and S3 bucket*
- ~ *Uploading objects to the S3 bucket*
- ~ *What is Amazon S3 Glacier?*
- ~ *Glacier Vaults*
- ~ *Glacier Archives*
- ~ *Accessing Amazon S3 Glacier*

## => AWS Databases :

- ~ *Enabling object versioning in the S3 bucket*
- ~ *Databases on AWS*
- ~ *Introduction to Amazon Relational Database Service(RDS)*
- ~ *Features of RDS*
- ~ *Engine types Configuration*
- ~ *RDS Pricing*
- ~ *Creating a SQL Server DB Instance*
- ~ *Introduction to Amazon Aurora*
- ~ *Benefits of Amazon Aurora*
- ~ *Create an Aurora DB cluster*
- ~ *Introduction to Amazon Dynamo DB*
- ~ *Components of DynamoDB*
- ~ *Creating a DynamoDB table.*
- ~ *Connecting to the DB Instance From Your Machine*
- ~ *DynamoDB Items and Indexes*
- ~ *Dynamo Backup and Restore*

## => Collection :

- ~ *Collection*
- ~ *Collection Section Introduction*
- ~ *Kinesis Data Streams Overview*
- ~ *Hot shard*
- ~ *Kinesis Producers*
- ~ *Kinesis Consumers*
- ~ *Kinesis Enhanced Fan Out*
- ~ *Kinesis Scaling*
- ~ *Kinesis - Handling Duplicate Records*
- ~ *Kinesis Security*
- ~ *Kinesis Data Firehose*
- ~ *CloudWatch Subscription Filters with Kinesis*
- ~ *SQS Overview*
- ~ *SQS Hands On*
- ~ *Kinesis Data Streams vs SQS*
- ~ *IoT Overview*
- ~ *IoT Components Deep Dive*
- ~ *Database Migration Service (DMS)*
- ~ *Direct Connect*
- ~ *AWS Snow Family Overview*
- ~ *AWS Snow Family Hands On*
- ~ *MSK: Managed Streaming for Apache Kafka*
- ~ *Kinesis vs MSK*

## => Storage :

- ~ *S3 Overview*
- ~ *S3 Hands On*
- ~ *S3 Security: Bucket Policy*
- ~ *S3 Security: Bucket Policy Hands On*
- ~ *S3 Website Overview*
- ~ *S3 Website Hands On*
- ~ *S3 Versioning Overview*
- ~ *S3 Versioning Hands On*
- ~ *S3 Server Access Logging*
- ~ *S3 Server Access Logging Hands On*
- ~ *S3 Replication Overview*
- ~ *S3 Replication Hands On*

- ~ S3 Storage Classes Overview
- ~ S3 Storage Classes Hands On
- ~ S3 Glacier Vault Lock & S3 Object Lock
- ~ S3 Encryption
- ~ Shared Responsibility Model for S3
- ~ DynamoDB Overview
- ~ DynamoDB RCU & WCU
- ~ DynamoDB Partitions
- ~ DynamoDB APIs
- ~ DynamoDB Indexes: LSI & GSI
- ~ DynamoDB DAX
- ~ DynamoDB Streams
- ~ DynamoDB TTL
- ~ DynamoDB Security
- ~ DynamoDB: Storing Large Objects

#### => Processing :

- ~ Section Introduction: Processing
- ~ Lambda Overview
- ~ Lambda Hands On
- ~ [Exercise] AWS Lambda
- ~ Why Cloud & Big Data on Cloud
- ~ What is Virtual Machine
- ~ On-Premise vs Cloud Setup
- ~ Major Vendors of Hadoop Distribution
- ~ Hdfs vs S3
- ~ Important Instances in AWS
- ~ Spark Basics
- ~ Why spark is difficult
- ~ Overview of EMR
- ~ What is EMR
- ~ Tez vs mapreduce
- ~ Launching an emr cluster
- ~ connecting to your cluster
- ~ Create a tunnel for web ui
- ~ Use Hue to interact with EMR
- ~ Transient vs Long Running Cluster Running
- ~ Copy File From S3 to Local Zeppelin Notebook
- ~ How to Create a VM
- ~ S3 & EBS
- ~ Public ip Vs Private Ip
- ~ Aws Command Line Interface
- ~ AWS Glue
- ~ Introduction to Amazon Redshift
- ~ Redshift Master Slave Architecture
- ~ redshift demo
- ~ redshift spectrum
- ~ Redshift Distribution Styles
- ~ Redshift Fault Tolerance
- ~ Redshift Sort Keys

#### => Analysis :

- ~ Section Introduction: Analysis
- ~ Intro to Kinesis Analytics
- ~ Kinesis Analytics Costs; RANDOM\_CUT\_FOREST
- ~ Intro to Opensearch (formerly Elasticsearch)
- ~ Amazon Opensearch Service
- ~ Opensearch Features
- ~ What is Athena
- ~ When do we require Athena What problem Athena Solve How Athena Works
- ~ Athena Pricing
- ~ Athena Practical Demonstration

#### => Visualization :

- ~ The course overview
- ~ big data analytics and aws
- ~ How Quicksight is different than other BI Tools
- ~ BI solution based on quicksight
- ~ how to get started with quicksight
- ~ Performance Your first analysis
- ~ AWS Big data ecosystem
- ~ importing files to quicksight
- ~ importing databases to quicksight
- ~ importing data from saas services to quicksight
- ~ edit existing data sources in quicksight
- ~ Joining datasets
- ~ using functions
- ~ applying filters
- ~ understanding spice layer
- ~ Creating a Quicksight Analysis
- ~ Explore various charting options
- ~ Exploring various Map options
- ~ Exploring various table and other visual options
- ~ Mini project Overview
- ~ Mini Project Architecture
- ~ Data ingestion for mini project
- ~ Reports and dashboards

#### => Introducing Google Cloud Platform :

- ~ Google platform fundamentals overview.

~ Google cloud platform Big Data products.

## => Compute and Storage Fundamentals :

~ CPUs on demand (compute engine).

~ A global filesystem (cloud storage).

~ CloudShell.

~ Set up an Ingest-Transform-Publish data processing pipeline.

## => Data Analytics on the Cloud :

~ Stepping-stones to the cloud.

~ Cloud SQL: your SQL database on the cloud.

~ Importing data into CloudSQL and running queries.

~ Spark on Dataproc.

~ Machine Learning recommendations with Spark on Dataproc.

## => Scaling Data Analysis :

~ Fast random access.

~ Datalab

~ BigQuery.

## => Data Processing Architectures :

~ Message-oriented architectures with Pub/Sub.

~ Real time streaming using Pub/Sub

~ Creating pipelines with Dataflow.

~ Reference architecture for real-time and batch data processing.

~ Google Online Transfer

~ Cloud Storage Transfer

~ Google Cloud BigTable

~ Google Cloud Dataflow

~ Google Cloud Dataproc

~ Google Cloud Pub/Sub

~ Google Cloud Composer

~ Google Cloud Data Fusion

~ Automating ETL jobs with composer and fusion

~ Google Cloud data catalog

~ Google Data studio

~ Architecture: Optimizing large-scale Ingestion

~ GCP Big Data Outro

## => Introduction to cloud :

~ Introduction to Cloud Computing

~ Cloud models

~ Different cloud providers

## => Regions and Availability Zones :

~ Understanding regions and availability zones in Azure

~ Creating microsoft Azure account

## => Resource Hierarchy :

~ Understanding resource hierchy

~ Demo on resource hierchy

~ Resource groups, subscription and managment groups

~ Active directory

## => Introduction to azure cloud computing :

~ Azure services overview

~ Managed and unmanaged service

~ Demo create Azure SQL Database service

## => Introduction to data engineer profile :

~ Introduction to data engineer Technologies

~ Data engineer role and responsibility

~ Introduction to data engineer technologies

## => Azure sql database :

~ Module Introduction

~ Introduction

~ Why choosing SQL Server in Azure

~ Azure IaaS vs PaaS database offerings

~ SQL server paas deployment options

~ Introduction to Azure sql server in virtual machine

~ SQL Server in Azure virtual machine

~ SQL server in azure virtual machine

~ Introduction Azure single database

~ Demo Azure single database

~ Purchasing Models and service tier

~ Azure database vs azure datawarehouse

~ Introduction elastic data pool

~ Azure Elastic Database

~ Azure Elastic Database

~ Introduction managed instance Database

~ Azure managed instance Database

~ Difference between on premises and manged instance

~ Service tiers for managed instance

~ Mangement operations

~ Demo managed instance

## => Azure synapse :

~ Module introduction

~ Why warehouse in cloud?

~ Traditional vs modern warehouse architecture

~ What is synapse analytics service?

- ~ Demo create dedicated sql pool
- ~ Demo connect sql pool with ssms
- ~ Demo create Azure synapse analytics workspace
- ~ Demo explore synapse studio v2
- ~ Demo create dedicated sql pool and spark pool from inside synapse studio
- ~ Demo analyse data using dedicated sql pool
- ~ Analyse data using apache spark notebook
- ~ Demo analyse data using serverless sql
- ~ Demo data factory copy tool from synapse integrate tab
- ~ Demo monitor synapse analytics studio
- ~ Azure synapse a game changer
- ~ Azure synapse benefits

=> Azure databricks :

- ~ Spark Basics
- ~ Why spark is difficult?
- ~ Why databricks in cloud?
- ~ How to save databricks demo cost
- ~ Demo provision databricks, clusters and workbook
- ~ Demo mount data lake to databricks DBFS
- ~ Demo Explore, Analyze, Clean, Transform and load data in databricks
- ~ Azure databricks cluster
- ~ Azure databricks other important components
- ~ Databricks monitoring

=> Azure data factory :

- ~ What is Data Factory?
- ~ Data factory in azure ecosystem
- ~ Provision Azure data factory instance
- ~ Data factory components
- ~ Data factory pipeline and activities
- ~ Data factory linked service and datasets
- ~ Data factory integration runtime
- ~ Data factory triggers
- ~ Data factory copy data activity demo
- ~ Copy data activity using author demo
- ~ Secure input and output property
- ~ User properties
- ~ Data factory parameters
- ~ Data flow concept
- ~ Mapping data flow
- ~ Wrangling data flow
- ~ Monitoring
- ~ Metrics and diagnostic settings

=> Introduction to SQL :

- ~ Why SQL?
- ~ Application of SQL
- ~ Characteristics of SQL
- ~ Installation guide
- ~ Connection & set up
- ~ Create database
- ~ RENAME database
- ~ Drop database
- ~ SELECT database

=> Data type of SQL :

- ~ Binary datatypes
- ~ Approximate numeric datatype
- ~ Exact numeric datatype
- ~ Character string datatype
- ~ Date and time datatype

=> Introduction to SQL syntax :

- ~ SQL SELECT statement
- ~ SQL WHERE clause
- ~ SQL DISTINCT clause
- ~ SQL AND/OR clause
- ~ SQL IN clause
- ~ SQL LIKE clause
- ~ SQL BETWEEN clause
- ~ SQL ORDER BY clause
- ~ SQL GROUP BY clause
- ~ SQL COUNT clause
- ~ SQL HAVING clause
- ~ SQL CREATE TABLE statement
- ~ SQL DROP TABLE statement
- ~ SQL CREATE INDEX statement
- ~ SQL DROP INDEX statement
- ~ SQL DESC statement
- ~ SQL TRUNCATE TABLE statement
- ~ SQL ALTER TABLE statement
- ~ SQL ALTER TABLE statement(rename)
- ~ SQL INSERT INTO statement
- ~ SQL UPDATE statement
- ~ SQL DELETE statement
- ~ SQL CREATE DATABASE statement
- ~ SQL DROP DATABASE statement
- ~ SQL USE statement
- ~ SQL COMMIT statement

~ SQL ROLLBACK statement

## => Operators in SQL :

- ~ Arithmetic operators
- ~ Comparison operators
- ~ Logical operators
- ~ Operators used to negate conditions

## => SQL Query :

- ~ CREATE table
- ~ CREATE table with PRIMARY KEY
- ~ CREATE table with FOREIGN KEY
- ~ DELETE table
- ~ TRUNCATE table
- ~ TEMP table
- ~ RENAME table
- ~ DROP table
- ~ COPY table
- ~ ALTER table
- ~ INSERT query
- ~ UPDATE query
- ~ DELETE query

## => SELECT Query :

- ~ SELECT statement
- ~ SELECT UNIQUE
- ~ SELECT DISTINCT
- ~ SELECT COUNT
- ~ SELECT TOP
- ~ SELECT LAST
- ~ SELECT RANDOM
- ~ SELECT IN
- ~ SELECT RANDOM
- ~ SELECT MULTIPLE
- ~ SELECT DATE
- ~ SELECT SUM
- ~ SELECT NULL
- ~ SELECT group by

## => SQL Clause :

- ~ WHERE clause
- ~ AND clause
- ~ OR clause
- ~ WITH clause
- ~ AS clause
- ~ HAVING clause
- ~ Like clause
- ~ IS NULL clause
- ~ UNION clause
- ~ UNION All clause
- ~ Top clause

## => SQL Order By :

- ~ ORDER BY clause
- ~ ORDER BY ASC
- ~ ORDER BY DESC
- ~ ORDER BY

## => SQL Constraints :

- ~ NOT NULL constraint
- ~ DEFAULT constraint
- ~ UNIQUE constraint
- ~ PRIMARY key
- ~ FOREIGN key
- ~ CHECK constraint
- ~ INDEX
- ~ Introduction to views

## => Functions(Aggregate) :

- ~ Conditional aggregation
- ~ List concatenation
- ~ SUM
- ~ AVG()
- ~ Count
- ~ Min
- ~ Max

## => SQL Joins :

- ~ INNER JOIN
- ~ LEFT JOIN
- ~ RIGHT JOIN
- ~ FULL JOIN
- ~ SELF JOIN
- ~ CARTESIAN JOIN

## => Views in SQL :

- ~ Creating view
- ~ Creating view from single table
- ~ Creating view from multiple tables
- ~ Delete view

## => Window Functions :

- ~ Setting up a flag if other rows have a common property
- ~ Finding "Out-of-Sequence" records using the LAG() function
- ~ Getting a running total
- ~ Adding the total rows selected to every row
- ~ Getting the N most recent rows over multiple grouping

=> Subqueries :

- ~ Subquery in FROM clause
- ~ Subquery in SELECT clause
- ~ Subquery in WHERE clause
- ~ Correlated subqueries
- ~ Filter query results using query on different table
- ~ Subqueries in FROM clause
- ~ Subqueries in WHERE clause

=> Stored Procedures :

- ~ Create and call a stored procedure
- ~ In and out parameters
- ~ If, Elseif and Else
- ~ Case
- ~ While
- ~ Repeat
- ~ Cursor
- ~ Loop
- ~ Error handling
- ~ User defined errors
- ~ Transactions
- ~ Stored functions

=> Triggers :

- ~ CREATE TRIGGER
- ~ Use trigger to manage a "Recycle Bin" for deleted items

=> AWS Lambda :

- ~ What is AWS Lambda and Why it is needed?
- ~ Features & Limitations of Lambda
- ~ Hello world program using Lambda
- ~ Auto trigger Lambda Function based on S3 file upload notification
- ~ Access other services from Lambda

=> AWS Secret Manager :

- ~ Create and Maintain secrets
- ~ Accessing credentials from Secret Manager using Boto3

=> AWS Glue :

- ~ Setting up cluster in Glue
- ~ Properties of Glue
- ~ Creating Catalogs in Glue
- ~ Read partitioned Data
- ~ Bulk and Incremental data processing from S3 in Glue
- ~ Data Processing in Glue
- ~ Glue jobs and Triggers

=> AWS SQS :

- ~ What is SQS?
- ~ Different types of SQS?
- ~ At-Least once and Exactly once delivery via SQS
- ~ Ingesting data to SQS
- ~ Inflight messages
- ~ Consume data from SQS
- ~ Dead Letter Queue

=> AWS Kinesis :

- ~ Ingesting real time data in Kafka Streams
- ~ Consume real time data from Kafka Streams

=> AWS Cloudwatch :

- ~ Cron based triggers
- ~ Event pattern based triggers
- ~ Monitoring & Alerting using Cloudwatch

=> AWS QuickSight :

- ~ Creating business dashboards using Quick sight

# Computer Vision Crash Course

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/Computer-Vision-Crash-Course>

## Course Description :-

This specialisation is the first to cover the fundamentals of computer vision in depth. It is aimed at learners, practitioners, and researchers who have little or no experience with computer vision and focuses on the mathematical and physical foundations of vision. Any learner who completes this specialisation has the potential to succeed in the field of computer vision, which is a booming field that is predicted to grow in importance in the next decades.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Fundamentals of Computer Vision
- => CNN architectures, Classification
- => Various architecture usages with Computer Vision for advanced level works

## Requirements :-

- => Basic knowledge of Python programming
- => A system with stable internet connection
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> CNN overview :

- ~ Intro to CNN and Padding Preview
- ~ Batch Normalization & Implementation Preview

=> Advanced Computer Vision Part 1 :

- ~ Intro to Transfer learning - and its architectures - Lenet, Alexnet, vgg16/19 architecture understanding
- ~ Implementation of VGG16 on dogs and cats images in Tensorflow 2.x
- ~ RCNN and Fast RCNN and Object detection basic introduction
- ~ Faster RCNN architecture
- ~ Yolo architecture
- ~ SSD

=> Advanced Computer Vision Part 2 :

- ~ Mask RCNN and Tracking theory
- ~ GAN Part 1
- ~ GAN Part 2

# Build ETL Data Pipeline on AWS EMR Cluster

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA PROJECTS

Course link : <https://ineuron.ai/course/Build-ETL-Data-Pipeline-on-AWS-EMR-Cluster>

## Course Description :-

With the advent of powerful data warehouses like SnowFlake, BigQuery, redshift spectrum, etc that allow separation of storage and execution, it has become very economical to store data in the data warehouse and then transform them as required. This Project goes over how to design such a ELT system using AWS EMR and Hive. The main objective is to keep the code complexity and server management low, while automating as much as possible

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => Build ETL Data Pipeline on AWS EMR Cluster
- => Components of a Data Engineering Platform
- => Building ETL Pipeline
- => Store data in the data warehouse
- => Build Dashboard using Tableau
- => Hive

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- Build ETL Data Pipeline on AWS EMR Cluster :
  - ~ Introduction of Instructor
  - ~ Introduction to ETL
  - ~ Project Overview
  - ~ End Notes
  - ~ Problem Description
  - ~ Understand the application scope
  - ~ Tour to existing solution
  - ~ End Notes
  - ~ Data Infrastructure: Components used
  - ~ Aws services
  - ~ Data Visualization Tools
  - ~ End Notes
  - ~ Solution Description
  - ~ Data Architecture
  - ~ Tour to Architecture diagram
  - ~ Cost Involved
  - ~ End Notes
  - ~ Exploration of the dataset
  - ~ Creating EMR Cluster
  - ~ Login into EMR hive Project
  - ~ Upload Data into Amazon S3
  - ~ using Hive as ETL Tool
  - ~ Hive Data Insertion
  - ~ CXconnect Tableau to Amazon EMR Hive



- ~ *Plot Charts*
- ~ *Plot Dual Combination Charts*
- ~ *Other Carts*
- ~ *Building Dashboard*
- ~ *End Notes*
- ~ *Conclude the project*
- ~ *Assignments & External Resources*

# DBT

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/DBT>

## Course Description :-

DBT data build tool helps data teams work like software engineers, transform data and control the flow to ship trusted data, faster.

DBT data build tool is an exciting tool in modern data manipulation, due to the shift from ETL to ELT in companies that rely on MPP databases in the cloud for example Snowflake, Redshift, Big query and others. this course will teach you the fundamentals of DBT data build tool. you will learn the structure of DBT data build tool and the main components.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Connect DBT to Snowflake or another database
- => Create SQL transformations that use consistent logic
- => Learn DBT Best Practices

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => DBT :
  - ~ What is DBT Preview
  - ~ DBT Cloud Account Creation
  - ~ Intro to Data Build Tool- create your first project
  - ~ DBT New project part 1 Preview
  - ~ DBT New project part 2
  - ~ DBT New Project part 3
  - ~ Snowflake connection
  - ~ Git push
  - ~ Adding raw sources to dbt project part 1
  - ~ Adding raw sources to dbt project part 2
  - ~ How dbt compile queries
  - ~ How to Write custom schemas
  - ~ How to test and debug dbt models
  - ~ Change the materialization
  - ~ Package management and dbt hub

# Machine Learning Bootcamp

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-Bootcamp>

## Course Description :-

In this Machine Learning Bootcamp you will learn technologies like Python, API, database, statistics, ML algorithms, deployment of ML models in various cloud platforms, and all machine learning algorithms. You will also learn about chatbots like Dialogflow, Amazon Lex, Azure Luis & RASA NLU. 15+ live projects are included to make your journey interesting from Zero to ML Engineer.

## Course Features :-

- => Machine Learning in Depth
- => CI/CD pipeline for ML
- => End to End Model Deployment in Azure, GCP & AWS
- => Time Series end-to-end implementation in ML
- => 20 + hands-on industry real-time projects
- => Power BI and Tableau self-placed course
- => Machine Learning Masters Certificate
- => 150+ hours live interactive classes
- => Doubt clearing session after the live classes
- => Lifetime Dashboard access
- => Doubt clearing one-to-one
- => Doubt clearing through mail and support team
- => Assignment in all the modules
- => 20+ use case of Machine learning
- => Live project with real-time implementation
- => Resume building
- => Career guidance
- => Interview Preparation
- => Regular assessment
- => Online Instructor-led learning

## What you will learn :-

- => Python
- => APIs
- => Databases
- => Python projects
- => Numpy
- => Pandas
- => Visualizations
- => Stats
- => Supervised Machine learning Algorithms
- => Unsupervised Machine learning Algorithms
- => Dimensionality Reduction
- => Machine Learning Projects
- => Deep learning
- => PowerBI
- => Tableau
- => Chatbots

## Requirements :-

- => Dedication
- => Laptop with internet connectivity

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sunny Savita :

~ I'm an AI enthusiast, graduate in Computer science and engineering. Currently working with iNeuron.ai as a Data Scientist and having 2+ years of experience. I have skills in big data, machine learning, computer vision, Natural language processing. My expertise also includes project design development and implementation with AI/ML tools.

## Curriculum details :-

=> Course Introduction :

- ~ Introduction of Data Science, AI, ML, DL and its application in Day to Day life
- ~ Course overview and Dashboard description

=> Installation and setup of the required software :

- ~ Installation and setup of Anaconda Distribution
- ~ Installation and setup of Pycharm and VScode
- ~ Complete walk-through of Jupyter Notebook in local
- ~ Setup of Google Colab with GPU
- ~ Create a virtual environment through anaconda and project setup

=> Introduction of Python :

- ~ Python Introduction and comparison with other Programming languages
- ~ Important Features of python
- ~ Testing Python Installation with hello world
- ~ Introduction To Predefined Functions And Modules
- ~ How print() function works ?
- ~ How To Remove Newline From print( ) ?
- ~ Rules For Identifiers, Python Reserved Words, Data Types In Python
- ~ Operators Arithmetic, Bitwise, Comparison, and Assignment operators, Operators Precedence and associativity
- ~ Compound Operators, Identity Operators

=> String :

- ~ What Is A String ?
- ~ Creating A String
- ~ Different Ways Of Accessing Strings
- ~ Operators Which Work On Strings
- ~ Built-In String Functions
- ~ Printing string using f-string
- ~ Modifying Strings
- ~ String conversion methods
- ~ String comparison methods
- ~ String searching methods
- ~ String replace methods

=> List :

- ~ What Is A List ?
- ~ Creating A List
- ~ Accessing The List Elements
- ~ Adding New Data In The List
- ~ The Slice Operator With List
- ~ Modifying A List
- ~ Deletion In A List
- ~ Appending / Prepending Items In A List
- ~ Multiplying A List
- ~ Membership Operators On List
- ~ Built-In Functions For List
- ~ Methods Of List
- ~ List Comprehension

=> Tuples :

- ~ What Is A Tuple and how to create Tuple
- ~ Differences between List and Tuples
- ~ Benefits Of Tuple
- ~ Packing / Unpacking A Tuple
- ~ Accessing A Tuple
- ~ Changing The Tuple
- ~ Deleting The Tuple
- ~ Functions Used With Tuple
- ~ Methods Used With Tuple
- ~ Operations Allowed On Tuple

=> Dictionaries and set :

- ~ What Is A Dictionary ?
- ~ What Is Key-Value Pair ?
- ~ Creating A Dictionary
- ~ Important Characteristics Of A Dictionary
- ~ Different Ways To Access A Dictionary
- ~ Updating Elements In Dictionary
- ~ Removing Elements From Dictionary
- ~ Functions Used In Dictionary
- ~ Dictionary Methods
- ~ Set introduction
- ~ Set methods

=> Decision Control Statements and loops in python :

- ~ if Statement
- ~ Concept of Indentation

- ~ *if-else Statement*
- ~ *if-elif-else Statement*
- ~ *Types of loop supported by Python*
- ~ *while loop*
- ~ *while-else loop*
- ~ *break, continue and pass Statement*
- ~ *for Loop*
- ~ *for Loop In Python*
- ~ *Differences with other languages*
- ~ *range( ) Function*
- ~ *Using for with range( )*

#### => Python Functions :

- ~ *What Is A Function ?*
- ~ *Function V/s Method*
- ~ *Steps Required For Developing User-Defined Function*
- ~ *Calling A Function*
- ~ *Returning Values From Function*
- ~ *Arguments V/s Parameters*
- ~ *Types Of Arguments*
- ~ *Variable Scope*
- ~ *Local Scope*
- ~ *Global Scope*
- ~ *Argument Passing*
- ~ *Anonymous Functions OR Lambda Function*
- ~ *The map( ) Function*
- ~ *The filter( ) Function*
- ~ *Using map( ) and filter( ) with Lambda Expressions*
- ~ *Iterators Generator functions*

#### => OOPS Concepts :

- ~ *Procedure Oriented Programming vs Object Oriented Programming*
- ~ *What Is A Classes and Object ?*
- ~ *\_\_init\_\_() Method*
- ~ *Types Of variable in class*
- ~ *Types Of Methods in class*
- ~ *Difference Between local variable, class variable and Instance variable*
- ~ *Difference Between Instance Method, Class Method, and Static Methods*
- ~ *concept of Encapsulation*
- ~ *How To Declare Private Members In Python ?*
- ~ *The setattr( ) And getattr( ) Functions*
- ~ *object Class, \_\_repr\_\_() and \_\_str\_\_() methods*
- ~ *concept of Inheritance*
- ~ *Types Of Inheritance*
- ~ *Single Inheritance*
- ~ *Using super( )*
- ~ *Method Overriding*
- ~ *MultiLevel Inheritance*
- ~ *Hierarchical Inheritance*
- ~ *Multiple Inheritance*
- ~ *The MRO Algorithm*
- ~ *Hybrid Inheritance*
- ~ *The Diamond Problem*
- ~ *Operator Overloading*

#### => Exception Handling :

- ~ *Introduction To Exception Handling*
- ~ *Exception Handling Keywords*
- ~ *Exception Handling Syntax*
- ~ *Handling Multiple Exceptions*
- ~ *Handling All Exceptions*

#### => Python logging :

- ~ *What is logging?*
- ~ *When to use logging?*
- ~ *Logging to a file*
- ~ *Different level of logging*
- ~ *Logging from multiple modules*
- ~ *Logging variable data*
- ~ *Display Date&Time in logging file*

#### => Working With Files :

- ~ *Working with files*
- ~ *Reading and writing files*
- ~ *Buffered read and write*
- ~ *Other File methods*

#### => Database :

- ~ *What Is A Database ?*
- ~ *Steps Needed For Connecting To MySQL From Python*
- ~ *Exploring Connection And Cursor Objects*
- ~ *Executing The SQL Queries*
- ~ *Different Ways Of Fetching The Data*
- ~ *Executing INSERT Command*
- ~ *Executing Update Command*
- ~ *Executing Delete Command*
- ~ *Introduction MongoDB*
- ~ *What is MongoDB Atlas and features MongoDB Atlas*
- ~ *MongoDB atlas setup*
- ~ *Querying the documents*

- ~ Finding, Inserting, Deleting & Updating elements
- ~ Bulk insert operations
- ~ Updating multiple document
- ~ Understanding insertOne vs insertMany()
- ~ Updateone() vs updateMany()
- ~ Understanding find() & fetchall()
- ~ Understanding "deleteOne()" & "deleteMany()"
- ~ Filtering documents

=> API :

- ~ Flask Introduction
- ~ Flask variable rules
- ~ Flask templates and static files
- ~ App Routing Flask
- ~ URL Building Flask
- ~ HTTP Methods Flask
- ~ Flask requesting object
- ~ Flask sending Form data to Template

=> Python Pandas Modules :

- ~ Pandas Series
- ~ Pandas DataFrame
- ~ Pandas Panel
- ~ Pandas Basic functionality
- ~ Pandas read CSV
- ~ Pandas read JSON
- ~ Pandas reading data from MySQL
- ~ Pandas aggregations
- ~ Pandas group by
- ~ Pandas merging and joining
- ~ Pandas concatenation operation
- ~ Pandas date functionality
- ~ Pandas .loc() and .iloc() function
- ~ Pandas windows functions
- ~ Pandas indexing and selecting data
- ~ Cleaning data with pandas
- ~ Working with missing data
- ~ Working with categorical data

=> Python Numpy Modules :

- ~ NumPy Narray Object
- ~ NumPy Data Types
- ~ NumPy Array Attributes
- ~ NumPy Array Creation Routines
- ~ NumPy Array from Existing
- ~ Data Array From Numerical Ranges
- ~ NumPy Indexing & Slicing
- ~ NumPy Advanced Indexing
- ~ NumPy Broadcasting
- ~ NumPy Iterating Over Array
- ~ NumPy Array Manipulation
- ~ NumPy Binary Operators
- ~ NumPy String Functions
- ~ NumPy Mathematical Functions
- ~ NumPy Arithmetic Operations
- ~ NumPy Statistical Functions
- ~ Sort , Search & Counting Functions
- ~ NumPy Byte Swapping
- ~ NumPy Copies Views
- ~ NumPy Matrix Library
- ~ NumPy Linear Algebra

=> Python Visualization Modules :

- ~ Matplotlib Pyplot
- ~ Matplotlib Plotting
- ~ Matplotlib Subplot
- ~ Matplotlib Line Chart
- ~ Matplotlib Bar Chart
- ~ Matplotlib Histogram Chart
- ~ Matplotlib Pie chart
- ~ Seaborn Histogram
- ~ Seaborn Kernel density estimates
- ~ Seaborn Facet grid
- ~ Seaborn Pairgrid
- ~ Seaborn Boxplot, violin plot and contour plot
- ~ Seaborn Countplot
- ~ Seaborn Heatmap
- ~ Plotly Barchart histogram and pie chart
- ~ Plotly scatter plot and Bubble chart
- ~ Plotly distplot, density plot, and error bar plot
- ~ Plotly Heatmap
- ~ Plotly 3-D scatter plot and surface plot
- ~ Plotly with pandas and cufflinks
- ~ Plotly with matplotlib and chartstudio
- ~ Visualizing pairwise relationship
- ~ Finding statical estimation
- ~ Finding linear relationship
- ~ Finding correlation between variable

=> Statistics :

- ~ Introduction
- ~ Different types of Statistics
- ~ Population vs Sample
- ~ Mean, Median and Mode
- ~ Variance, Standard Deviation
- ~ Sample Variance why  $n-1$
- ~ Standard Deviation
- ~ Variables
- ~ Random Variables
- ~ Percentiles & quartiles
- ~ 5 number summary
- ~ Histograms
- ~ Gaussian - Normal distribution
- ~ Standard Normal distribution
- ~ Application Of Zscore
- ~ Basics Of Probability
- ~ Addition Rule In Probability
- ~ Multiplication rule in probability
- ~ Permutation
- ~ Combination
- ~ Log Normal Distribution
- ~ Central Limit theorem
- ~ Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median, And Mode
- ~ Covariance
- ~ Pearson And Spearman Rank Correlation
- ~ What is P Value?
- ~ What is Confidence Intervals
- ~ How To Perform Hypothesis Testing - Confidence Interval Z Test Statistics Derive Conclusion
- ~ Hypothesis testing part 2
- ~ Hypothesis testing part 3
- ~ Finalizing statistics

#### => Exploratory Data Analysis :

- ~ Feature Engineering and Selection
- ~ Create a profile of the data
- ~ Perform statical analysis
- ~ Building Tuning and Deploying Models
- ~ Perform EDA with automated library
- ~ Analyzing Bike Sharing Trends
- ~ Analyzing Movie Reviews Sentiment
- ~ Customer Segmentation and Effective Cross Selling
- ~ Analyzing Wine Types and Quality
- ~ Analyzing Music Trends and Recommendations
- ~ Forecasting Stock and Commodity Prices

#### => Machine Learning Module 1 :

- ~ Introduction of machine learning
- ~ Difference between Supervised, Unsupervised & Semi-supervised
- ~ Linear Regression Mathematical Institution
- ~ Linear Regression assumption.
- ~ OLS
- ~ Different Training methodology
- ~ Train, Test, Validation Split
- ~ Hands-on Linear regression in Python from scratch
- ~ Complete hands-on with Scikit learn
- ~ Overfitting, underfitting
- ~ Ridge Regression
- ~ Lasso Regression
- ~ Elastic Net Regression
- ~ Polynomial Regression
- ~ Logistics regression
- ~ Difference between Linear Regression and Logistic Regression
- ~ Performance matrix
- ~ Confusion matrix
- ~ Precision, Recall, ROC, AUC Curve
- ~ F-beta Score

#### => Machine Learning Module 2 :

- ~ SVR(support vector regressor)
- ~ SVC(support vector classifier)
- ~ SVM(Support vector machine)
- ~ KNN Classifier
- ~ KNN Regressor
- ~ K Nearest Neighbour
- ~ Lazy learners
- ~ KNN Issues
- ~ Performance measurement of KNN

#### => Machine Learning Module 3 :

- ~ Decision Tree Classifier
- ~ Decision tree Regressor
- ~ Cross Validation
- ~ Bias vs Variance
- ~ Ensemble approach
- ~ Bagging
- ~ Boosting
- ~ Stacking
- ~ Random Forest

#### => Machine Learning Module 4 :

- ~ Ada boosting
- ~ Gradient boosting
- ~ XGBoosting
- ~ Hands-on XgBoost

#### => Unsupervised Machine Learning :

- ~ Introduction to K-Means Clustering
- ~ Hard K-Means clustering
- ~ Soft K-Means clustering
- ~ Visualizing Each Step of K-Means
- ~ How to Choose K value
- ~ Advantages and Disadvantages of K-Means Clustering
- ~ Examples of where K-Means can fail
- ~ How to Evaluate a Clustering algorithm
- ~ Silhouette Coefficient
- ~ Dunn's Index
- ~ Python implementation using K-Means on Real Data
- ~ Real-time Clustering Application
- ~ Visual Walkthrough of Agglomerative Hierarchical Clustering
- ~ Using Hierarchical Clustering in Python and Interpreting the Dendrogram
- ~ python implementation of Agglomerative Clustering
- ~ DBSCAN: A Density-Based Clustering Algorithm
- ~ How to use DBSCAN: A Density-Based Clustering Algorithm for outlier detection
- ~ Python implementation of DBSCAN

#### => Dimension Reduction Techniques :

- ~ Principal Component Analysis (PCA)
- ~ T-distributed Stochastic Neighbor Embedding(t-SNE)
- ~ Curse of Dimensionality

#### => Natural Language Processing :

- ~ Text Analytics
- ~ Tokenizing, Chunking
- ~ Document term
- ~ Matrix TFIDF
- ~ Sentiment analysis hands-on
- ~ Naive Bayes classifier

#### => Deep Learning :

- ~ Deep Learning Introduction.
- ~ Neural Network Architecture.
- ~ Loss Function.
- ~ Cost Function.
- ~ Optimizers.
- ~ CNN architecture.
- ~ Build First Classifier in CNN.
- ~ Deploy Classifier over the cloud.
- ~ RNN overview.
- ~ GRU.
- ~ LSTM.
- ~ Time Series using RNN LSTM.
- ~ Customer Feedback analysis using RNN LSTM.

#### => Time series :

- ~ Arima
- ~ Sarima .
- ~ Auto Arima
- ~ Time series using RNN LSTM .
- ~ Prediction of NIFTY stock price.

#### => Machine Learning Deployment :

- ~ Deployment of projects in AWS AZURE, and GCP
- ~ Expose api to a web browser and mobile application retraining approach of Machine learning model
- ~ Devops infrastructure for machine learning model
- ~ Database integration and scheduling of machine learning model and retraining custom machine learning training approach.
- ~ AUTO ML
- ~ Discussion on infra cost and data volume
- ~ Prediction based on streaming data

#### => Machine Learning Extra Sessions :

- ~ Discussion on project explanation in interviews
- ~ Data scientist roles and responsibilities
- ~ Data scientist day to day work
- ~ Companies which hire a data scientist
- ~ Resume discussion with our team one to one

### Project details :-

#### => Python Projects :

- ~ Web Crawlers for Image Data, Product Review Sentiment Analysis
- ~ Integration with FrontEnd
- ~ Integration with Rest APIs with Web Apps and MongoDB
- ~ Deployment on Web Apps on Azure
- ~ Text Mining
- ~ Social Media Data Churn

#### => Machine Learning Projects :

- ~ Healthcare Analytics Prediction of medicines based on FITBIT band
- ~ Revenue Forecasting for Startups
- ~ Prediction of order cancellation at the time of ordering inventories
- ~ Anomaly detection in inventory packaged material.



- ~ *Fault detection in wafers based on sensors data*
- ~ *Demand forecasting for FMCG product.*
- ~ *Threat Identification in Security Systems*
- ~ *Defect detection in vehicle engine.*
- ~ *Food price forecasting with Zomato dataset.*
- ~ *Cement Strength regression*
- ~ *Credit Card Fraud*
- ~ *Forest Cover Classification*
- ~ *Income Prediction*
- ~ *Mushroom Classifier*
- ~ *Phishing Classifier*
- ~ *Thyroid Detection*

# AWS Cloud Master

---

Topic Name : CLOUD

Sub-topic Name : AWS

Course link : <https://ineuron.ai/course/AWS-Cloud-Master>

## Course Description :-

AWS is one of the major cloud providers across the globe comes with a variety of services in different tech stacks and getting certification in AWS plays a vital role to get various opportunities such as cloud administrator, cloud developer, Solution Architect, Machine Learning Speciality, Big Data Speciality, etc. This entire curriculum is designed keeping all the things in mind, so that person can learn, develop and implement in real-time.

## Course Features :-

- => Online Instructor-led learning
- => Doubt Clearing
- => Proper Roadmap for AWS Certification
- => Lifetime Dashboard access
- => Recording of Live Class
- => Material
- => Interview Questions
- => Resume Building
- => Career Guidance
- => Quiz in every module
- => Certificate

## What you will learn :-

- => AWS Certification
- => All Services
- => Real time implementation
- => Understanding of cloud services w.r.t ML, DL,etc by implementing Projects
- => ETL Pipeline
- => ML/DL Model Testing and Monitoring
- => Understand different Cloud Computings, EC2, Elastic Bean, AWS Lightsail, and more
- => Scenario based questions in AWS Certification

## Requirements :-

- => AWS account
- => A Windows, Linux, or MAC system
- => Your Dedication

## Instructors :-

- => Sachin Agarwal :
- ~

## Curriculum details :-

- => Cloud Introduction :
  - ~ *What is Cloud? Preview*
  - ~ *Why cloud is required?*
  - ~ *Introduction to AWS Portal*
  - ~ *Services provided by AWS*
  - ~ *Various level of Certification Exams by AWS*
  - ~ *Benefits of Certification*
  - ~ *Preparation Strategy*
  - ~ *Marks Distribution of various exams in AWS*
  - ~ *Comparison with Different Cloud*
- => Let's Get Started :
  - ~ *Account Creation Preview*
  - ~ *Internal Service Overview*
  - ~ *Costing*
  - ~ *AWS Cost Explorer*
  - ~ *AWS Budget*
  - ~ *AWS Marketplace Subscription*
  - ~ *AWS Application Post Profiler*

- ~ *Cost Anomaly Detection*
- ~ *Rightsizing recommendations*
- ~ *Saving*
- ~ *Utilization Report*
- ~ *Converge Report*
- ~ *Billing Console*
- ~ *Budget Report*
- ~ *Cost & Usage Reports*
- ~ *Cost Categories*
- ~ *Cost allocation tags*
- ~ *Orders and invoices*
- ~ *Credits*
- ~ *Purchase orders*

#### => Identity and Access Management (IAM) :

- ~ *What is IAM? Preview*
- ~ *Uses of IAM*
- ~ *Access management*
- ~ *User groups*
- ~ *Users*
- ~ *Roles*
- ~ *Difference between Users and Groups*
- ~ *Policies*
- ~ *Difference between Policies and Roles*
- ~ *Identity providers*
- ~ *Account settings*
- ~ *Access reports*
- ~ *Access analyzer*
- ~ *Archive rules*
- ~ *Analyzers*
- ~ *Settings*
- ~ *Credential report*
- ~ *Organization activity*
- ~ *Service control policies (SCPs)*
- ~ *Resource Access Manager*
- ~ *Shared by me*
- ~ *Resource shares*
- ~ *Shared resources*
- ~ *Principals*

#### => AWS Single Sign-On :

- ~ *AWS SSO prerequisites*
- ~ *Enable AWS SSO*
- ~ *Choose your identity source*
- ~ *Set up SSO to your AWS accounts*
- ~ *Set up SSO to your applications*

#### => AWS Signer :

- ~ *AWS Signer console*
- ~ *Signer CLI*
- ~ *AWS Serverless Application Model (AWS SAM) CLI*

#### => EC2 :

- ~ *EC2 Dashboard*
- ~ *Events*
- ~ *Tags*
- ~ *Limits*
- ~ *Instances*
- ~ *Instance Types*
- ~ *Launch Templates*
- ~ *Spot Requests*
- ~ *Savings Plans*
- ~ *Reserved Instances*
- ~ *Dedicated Hosts*
- ~ *Capacity Reservations*
- ~ *Images*
- ~ *AMIs*
- ~ *Elastic Block Store*
- ~ *Volumes*
- ~ *Snapshots*
- ~ *Lifecycle Manager*
- ~ *Network & Security*
- ~ *Security Groups*
- ~ *Elastic IPs*
- ~ *Placement Groups*
- ~ *Key Pairs*
- ~ *Network Interfaces*
- ~ *Load Balancing*
- ~ *Load Balancers*
- ~ *Target Groups*
- ~ *Auto Scaling*
- ~ *Launch Configurations*
- ~ *Auto Scaling Groups*
- ~ *Family of EC2*
- ~ *EC2 Costing*
- ~ *EC2 Pricing Models*

#### => VPC :

- ~ *VPC Dashboard*
- ~ *Filter by VPC*

- ~ Select a VPC
- ~ VIRTUAL PRIVATE CLOUD
- ~ Your VPCs
- ~ Subnets
- ~ Route Tables
- ~ Internet Gateways
- ~ Egress Only Internet Gateways
- ~ DHCP Options Sets
- ~ Elastic IPs
- ~ Managed Prefix Lists
- ~ Endpoints
- ~ Endpoint Services
- ~ NAT Gateways
- ~ Peering Connections
- ~ SECURITY
- ~ Network ACLs
- ~ Security Groups
- ~ REACHABILITY
- ~ Reachability Analyzer
- ~ DNS FIREWALL
- ~ Rule Groups
- ~ Domain Lists
- ~ AWS NETWORK FIREWALL
- ~ Firewalls
- ~ Firewall policies
- ~ Network Firewall rule groups
- ~ VIRTUAL PRIVATE NETWORK (VPN)
- ~ Customer Gateways
- ~ Virtual Private Gateways
- ~ Site-to-Site VPN Connections
- ~ Client VPN Endpoints
- ~ TRANSIT GATEWAYS
- ~ Transit Gateways
- ~ Transit Gateway Attachments
- ~ Transit Gateway Route Tables
- ~ Transit Gateway Multicast
- ~ Network Manager
- ~ TRAFFIC MIRRORING
- ~ Mirror Sessions
- ~ Mirror Targets
- ~ Mirror Filters
- ~ Settings

#### => AWS LightSail :

- ~ What is LightSail?
- ~ Difference between EC2 and Light Sail
- ~ Instances
- ~ Containers
- ~ Databases
- ~ Networking
- ~ Storage
- ~ Snapshots

#### => AWS Lambda :

- ~ What is AWS Lambda
- ~ What is a Lambda function?
- ~ Extend other AWS services with custom logic
- ~ Build custom back-end services
- ~ Bring your own code
- ~ Completely automated administration
- ~ Built-in fault tolerance
- ~ Package and deploy functions as container images
- ~ Automatic scaling
- ~ Connect to relational databases
- ~ Fine grained control over performance
- ~ Connect to shared file systems
- ~ Run code in response to Amazon CloudFront requests
- ~ Orchestrate multiple functions
- ~ Integrated security model
- ~ Trust and integrity controls
- ~ Flexible resource model
- ~ "Integrate Lambda with your favorite operational tools "

#### => AWS Storage :

- ~ What are storages?
- ~ Different types of Storage
- ~ S3
- ~ EFS
- ~ FSx
- ~ S3 Glacier
- ~ Storage Gateway
- ~ AWS Backup
- ~ What are the uses of S3?
- ~ S3 Setup
- ~ What are buckets?
- ~ Policy Setup
- ~ Accessing Data from S3
- ~ Pricing of S3
- ~ Elastic File System

- ~ AWS Data Sync
- ~ AWS Transfer
- ~ Uses and Access
- ~ Security
- ~ Flexibility

#### => AWS Databases :

- ~ What are Databases?
- ~ Different Types of Databases
- ~ RDS
- ~ DynamoDB
- ~ ElastiCache
- ~ Neptune
- ~ Amazon QLDB
- ~ Amazon DocumentDB
- ~ Amazon Keyspaces
- ~ Amazon Timestream
- ~ RDS Dashboard
- ~ RDS Databases
- ~ RDS Query Editor
- ~ RDS Performance Insights
- ~ RDS Snapshots
- ~ RDS Automated backups
- ~ RDS Reserved instances
- ~ RDS Proxies
- ~ RDS Subnet groups
- ~ RDS Parameter groups
- ~ RDS Option groups
- ~ RDS Events
- ~ RDS Event subscriptions
- ~ RDS Recommendations
- ~ RDS Certificate update

#### => AWS Management & Governance :

- ~ CloudWatch
- ~ AWS Auto Scaling
- ~ CloudFormation
- ~ CloudTrail
- ~ Config
- ~ OpsWorks
- ~ Service Catalog
- ~ Systems Manager
- ~ AWS AppConfig
- ~ Trusted Advisor
- ~ Control Tower
- ~ AWS License Manager
- ~ AWS Well-Architected Tool
- ~ Personal Health Dashboard
- ~ AWS Chatbot
- ~ Launch Wizard
- ~ AWS Compute Optimizer
- ~ Resource Groups & Tag Editor
- ~ Amazon Grafana
- ~ Amazon Prometheus
- ~ AWS Proton
- ~ Incident Manager

#### => AWS Advanced Stack :

- ~ Amazon SageMaker
- ~ Amazon Augmented AI
- ~ Amazon CodeGuru
- ~ Amazon DevOps Guru
- ~ Amazon Comprehend
- ~ Amazon Forecast
- ~ Amazon Fraud Detector
- ~ Amazon Kendra
- ~ Amazon Lex
- ~ Amazon Personalize
- ~ Amazon Polly
- ~ Amazon Rekognition
- ~ Amazon Textract
- ~ Amazon Transcribe
- ~ Amazon Translate
- ~ AWS DeepComposer
- ~ AWS DeepLens
- ~ AWS DeepRacer
- ~ AWS Panorama
- ~ Amazon Monitron
- ~ Amazon HealthLake
- ~ Amazon Lookout for Vision
- ~ Amazon Lookout for Equipment
- ~ Amazon Lookout for Metrics

#### => AWS Analytics :

- ~ Analytics
- ~ Athena
- ~ Amazon Redshift
- ~ EMR
- ~ CloudSearch
- ~ Elasticsearch Service

- ~ *Kinesis*
- ~ *QuickSight*
- ~ *Data Pipeline*
- ~ *AWS Data Exchange*
- ~ *AWS Glue*
- ~ *AWS Lake Formation*
- ~ *MSK*
- ~ *AWS Glue DataBrew*
- ~ *Amazon FinSpace*

#### => AWS Machine Learning :

- ~ *AWS Sagemaker*
- ~ *AWS Sagemaker Groundtruth*
- ~ *AWS Sagemaker Neo*
- ~ *ML Training*
- ~ *AWS Machine Learning Embark*
- ~ *Enterprise search using Amazon Kendra*
- ~ *Automated code reviews using AWS Codeguru*
- ~ *Chatbots using Amazon Lex*

#### => AWS Data Analytics :

- ~ *Fraud Prevention using Amazon Fraud Detector*
- ~ *Demand forecasting using Amazon Forecast*
- ~ *Detect Anomalies in Metrics*

#### => AWS Reinforcement Learning :

- ~ *AWS DeepComposer*
- ~ *AWS DeepRacer*
- ~ *AWS DeepLens*

#### => AWS Deep Learning :

- ~ *AWS DL AMI's*
- ~ *AWS DL Containers*
- ~ *EC2 P3 GPU Instances*
- ~ *EC2 Inf1 AWS Inferentia Instances*
- ~ *EC2 G4 GPU Instances*
- ~ *EC2 C5 CPU Instances*
- ~ *EC2 F1 FPGA Instances*
- ~ *Elastic Inference*
- ~ *AWS Inferentia*
- ~ *AWS Neuron SDK*
- ~ *Edge*
- ~ *Tensorflow on AWS*
- ~ *Pytorch on AWS*
- ~ *Apache MXNet on AWS*

#### => AWS Computer Vision :

- ~ *Image and video analytics using Amazon Rekognition*
- ~ *Document analysis using Amazon Textract*
- ~ *Spot Products defects and automate quality inspection*
- ~ *Improve operations with CV at the edge*
- ~ *Image and Video analysis*

#### => AWS NLP :

- ~ *Bots and Virtual Agent*
- ~ *Speech to Text*
- ~ *Text to Speech*
- ~ *Personalized Recommendations using Amazon Personalize*
- ~ *Real Time Translations using Amazon Translate*
- ~ *Text to speech using Amazon Polly*
- ~ *Transcription using Amazon Transcribe*
- ~ *Advanced text analytics using Amazon Comprehend*

#### => AWS Big Data :

- ~ *Apache Spark*
- ~ *Apache Hive*
- ~ *Apache HBase*
- ~ *Apache Flink*
- ~ *Apache Hudi*
- ~ *Presto*

#### => Closure :

- ~ *Final Touch for AWS Certifications*
- ~ *Exam Preparation*
- ~ *Interview Question Discussion*
- ~ *Certification Question Discussion*
- ~ *End to End Setup and Revision*

# FastAPI

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FAST API

Course link : <https://ineuron.ai/course/FastAPI>

## Course Description :-

FastAPI is a modern, high-performance, web framework for building APIs with Python. This course is specifically developed for beginners! This implies that irrespective of your background, you will be able to master one of the most popular frameworks on the market. All you need is a basic understanding of Python.

## Course Features :-

- => Completion Certificate
- => Quiz in every module
- => Real-time Implementations

## What you will learn :-

- => Introduction to API
- => Getting Started with FastAPI
- => Features of FastAPI
- => Python Type Hints

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

=> Course Introduction :

- ~ Welcome to FastAPI course
- ~ Course pre-requisites
- ~ Who is this course for
- ~ What you will get from this course
- ~ What is API
- ~ Why API is used
- ~ Different ways to create API
- ~ Advantage of API
- ~ How to get access to course materials
- ~ What career path you can follow after completion

=> Getting Started with FastAPI :

- ~ What is FastAPI
- ~ Installing Python
- ~ Installing FastAPI
- ~ Your first 'hello world' program using default GET
- ~ GET and PUT
- ~ What is JSON format
- ~ JSON response
- ~ Practical GET and PUT
- ~ Interactive API docs
- ~ Alternative API docs
- ~ What is uvicorn
- ~ Features of using FastAPI

=> Python Hint Types :

- ~ Python Type hints introduction
- ~ Why they are important
- ~ Using hint types in function
- ~ Specifying function return type

=> FastAPI :

- ~ import FastAPI
- ~ Creating FastAPI Instance
- ~ URL
- ~ HTTP methods
- ~ What is decorator

- ~ *Defining a decorator*
- ~ *Defining path operation function*
- ~ *Path parameter*
- ~ *Path parameter with types*
- ~ *Working with python enumeration*
- ~ *Query parameters*
- ~ *Multiple path and query parameters*
- ~ *Request Body*
- ~ *Query Parameters and String Validations*
- ~ *Form data*
- ~ *Request files*
- ~ *Practical Demo Request forms and files*
- ~ *Handling errors*

=> Summary :

- ~ *Course Outro*



# Building Chatbot using Google DialogFlow

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Building-Chatbot-using-Google-DialogFlow>

## Course Description :-

This course will help students gain knowledge of building Chatbot using Google DialogFlow. In this course you will learn how to create various projects on chatbots. Upon successful completion of the course students will be able to create chatbots on their own using Google DialogFlow.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Chatbot Introduction
- => Basics of Dialogflow
- => Applications using GCP Dialogflow with UI
- => Integration
- => Projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shivan Kumar :

~ Associate Data Scientist, Mentor, and Kaggle Master with 2 Years of Experience. Experience in building models that translate data points into business insights. Highly accurate at collecting, analyzing, and interpreting large datasets, developing machine learning, deep learning, Chatbots models, and deploying the solutions on the cloud. I also love to participate in Hackathons. In my free time, I like to mentor students to learn Data Science and achieve their goals.

## Curriculum details :-

=> Chatbot Introduction :

- ~ What is a Chatbot?
- ~ How does Chatbot work?
- ~ Types of Chatbot
- ~ Applications of Chatbot
- ~ Architecture of Chatbot
- ~ Comparison of different Chatbot Platforms
- ~ Different types of Chatbot

=> Basics of Dialogflow :

- ~ Introduction to Google Dialogflow
- ~ How does Dialogflow Work?
- ~ Features of Dialogflow
- ~ Use cases of Dialogflow
- ~ Advantages of Dialogflow
- ~ Why should we choose Dialogflow
- ~ Components of the Dialogflow

=> Applications using GCP Dialogflow with UI :

- ~ How to create your first Dialogflow Agent
- ~ How to create new Intents
- ~ How to create Parameters in Dialogflow
- ~ How to create your own Entities
- ~ How to use your new Entities

- ~ Contexts in Dialogflow
- ~ How to customize the Default Welcome Intent
- ~ How to create a Custom Intent
- ~ How to customize the Fallback Intent
- ~ Create an intent with Parameters
- ~ Knowledge base in Dialogflow
- ~ Training in Dialogflow
- ~ Intent matching with Follow-up Intent
- ~ Webhooks
- ~ Testing
- ~ Integration
- ~ Training and Fallbacks

=> Projects :

- ~ History based Chatbot (No need to remember anything, you can ask directly to your chatbot)
- ~ General knowledge based Chatbot
- ~ Bio Chatbot
- ~ Syllabus bot (This type of chatbot can be used in school, colleges or educational institute to provide their syllabus)

# Data Structures and Algorithms using python

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH PYTHON

Course link : <https://ineuron.ai/course/Data-Structures-and-Algorithms-using-python>

## Course Description :-

The Data Structure and Algorithm program focused on learning algorithmic strategies for addressing a myriad of challenges while having complete control of memory and time. Develop a thorough understanding of how data structures work and how to create efficient algorithms.

## Course Features :-

- => Algorithm Analysis
- => Recurrence Relation
- => Array Data Structure
- => Divide and Conquer
- => Tree Data Structure
- => Heap Data Structure
- => Linked List
- => Skip List
- => Dynamic Programming
- => Quizzes
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Acquire knowledge of various popular algorithms like hash-table, Binary Search, Merge Sort and more
- => Dynamic programming
- => Develop some analytical skills and use them efficiently in Data Structure Algorithms
- => Acquire knowledge on various Sorting Algorithms
- => Implementation of Data Structures by using Python

## Requirements :-

- => A system with Internet Connection
- => Basic knowledge of Python

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

=> INTRODUCTION :

~ Course Overview Preview

=> ANALYSIS IN ALGORITHMS :

~ Introduction to Algorithms Preview

~ Analyzing Algorithms

~ Asymptotic Notations - Big O, Theta and Omega Notations

=> RECURRENCE RELATION :

~ Introduction to Recurrence Relation Solving Preview

~ Substitution Method - Problem 1

~ Substitution Method - Problem 2

~ Substitution Method - Problem 3

~ Substitution Method - Problem 4

~ Recursive Tree Method - Problem 1

~ Recursive Tree Method - Problem 2

~ Recursive Tree Method - Problem 3

~ Master's Theorem - Case 1

~ Master's Theorem - Case 2

~ Master's Theorem - Case 3

=> ARRAY DATA STRUCTURE :

~ Introduction to Arrays

~ One Dimensional Array - How to find the address of an Element

~ Two Dimensional Array - Row major order and column major order

- ~ Searching Algorithm - Linear search in an Array
- ~ Comparison Sort in an Array - Selection sort
- ~ Comparison Sort in an Array - Bubble sort
- ~ Comparison Sort in an Array - Insertion sort
- ~ Non-Comparison Sort in an Array - Count sort
- ~ Non-Comparison Sort in an Array - Radix sort
- ~ Non-Comparison Sort in an Array - Bucket sort
- ~ Interview-Based Problem Statement - Missing Number in an array
- ~ Solution Discussed - Missing Number in an array
- ~ Interview-Based Problem Statement and Brute Force Approach - Divide two Integer without division operator
- ~ Solution Discussed - Optimised Approach with Complexity Analysis

**=> DIVIDE AND CONQUER :**

- ~ Introduction to Divide and Conquer

# Class 10th Math

---

Sub-topic Name : Null

Course link : <https://ineuron.ai/course/Class-10th-Math>

## Course Description :-

This course is useful for Grade 10 students. It is focused on providing assistance to students who have to appear for board exams. In this course, the entire NCERT will be covered, Various questions from NCERT, NCERT exemplar and previous year will also be discussed. Dedicated doubt clearing sessions will also be conducted for helping the students regularly throughout their learning journey. Previous year board questions & sample papers will also be discussed.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Algebra
- => Geometry
- => Statistics
- => Probability
- => Coordinate Geometry

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :  
~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => Real Numbers :
  - ~ Lecture 1 : Theorem 1.1 Euclids Division Lemma statement Preview
  - ~ Lecture 2 : Euclids Division Algorithm Preview
  - ~ Lecture 3 : NCERT Solutions Ex1.1 Question 1&2
  - ~ Lecture 4 : NCERT Solutions Ex1.1 Question 3&4
  - ~ Lecture 5 : Theorem 1.2 Fundamental Theorem of Arithmetic & Example 6
  - ~ Lecture 6 : NCERT Solutions Ex1.2 Question 2 part1
  - ~ Lecture 7 : NCERT Solutions Ex1.2 Question 2&3
  - ~ Lecture 8 : NCERT Solutions Ex1.2 Question 4&5
  - ~ Lecture 9 : NCERT Solutions Ex1.2 Question 6&7
  - ~ Lecture 10 : Theorem 1.3&Theorem 1.4
  - ~ Lecture 11 : NCERT Solutions Ex1.3 Question 1&2
  - ~ Lecture 12 : Theorem 1.5,1.6&1.7
  - ~ Lecture 13 : NCERT Solutions Ex1.4
- => Polynomials :
  - ~ Lecture 1 : Introduction to Polynomials Preview
  - ~ Lecture 2 : Geometrical Meaning of Zeroes Preview
  - ~ Lecture 3 : NCERT Solutions Ex2.1
  - ~ Lecture 4 : Coefficient & Zeroes Relation
  - ~ Lecture 5 : NCERT Solutions Ex2.2
  - ~ Lecture 6 : Division Algorithm
  - ~ Lecture 7 : NCERT Solutions Ex2.3 Question 2 Parts1&2
  - ~ Lecture 8 : NCERT Solutions Ex2.3 Question 3
  - ~ Lecture 9 : NCERT Solutions Ex2.3 Question 4
- => Pair of Linear Equations in Two Variables :
  - ~ Lecture 1 : Introduction
  - ~ Lecture 2 : General form & Solution of Linear Equation
  - ~ Lecture 3 : Graphical Representation
  - ~ Lecture 4 : Solution of Simultaneous Equations
  - ~ Lecture 5 : Graphical method to solve L.E. & Consistency
  - ~ Lecture 6 : Graphical Method
  - ~ Lecture 7 : Number of solutions
  - ~ Lecture 8 : Algebraic Methods & SUBSTITUTION METHOD
  - ~ Lecture 9 : Elimination Method
- => Quadratic Equations :
  - ~ Lecture 1 : Quadratic Equations Introduction
  - ~ Lecture 2 : Introduction Quadratic Equations 1.2

- ~ Lecture 3 : Quadratic Equations Factorisation
- ~ Lecture 4 : Quadratic Equations Completing Square
- ~ Lecture 5 : Quadratic Equations Quadratic Formula
- ~ Lecture 6 : Nature of Roots
- ~ Lecture 7 : Ex4.2 Question 3-5
- ~ Lecture 8 : Completing Square Method Ex4.3 Question 1
- ~ Lecture 9 : NCERT Solutions Ex4.3 Question 2 Quadratic Formula
- ~ Lecture 10 : NCERT Solutions Ex4.3 Question 3
- ~ Lecture 11 : NCERT Solutions Ex4.3 Question 3 part2
- ~ Lecture 12 : NCERT Solutions Ex4.3 Question 5&6
- ~ Lecture 13 : NCERT Solutions Ex4.3 Question 5
- ~ Lecture 14 : NCERT Solutions Ex4.3 Question11
- ~ Lecture 15 : NCERT Solutions Ex4.4 Question1 V.NO.6
- ~ Lecture 16 : NCERT Solutions Ex4.4 Question 2
- ~ Lecture 17 : NCERT Solutions Ex4.4 Question 3
- ~ Lecture 18 : NCERT Solutions Ex4.4 Question 4
- ~ Lecture 19 : NCERT Solutions Ex4.4 Question 5

=> Arithmetic Progressions :

- ~ Lecture1\_AP\_Introduction
- ~ Lecture2\_nthTermAP
- ~ Lecture3\_MixedProblem\_AP
- ~ Lecture4\_NCERT\_Ex5.4
- ~ Lecture5\_WordProblem\_AP
- ~ Lecture6\_AP\_Proofs&Formulae
- ~ Lecture7\_NCERT\_Ex5.3\_ImpProblem

=> Triangles :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Similar Figures / Scale Factor
- ~ Lecture 3 : Basic Proportionality Theorem / Thales Theorem
- ~ Lecture 4 : Converse of Basic Proportionality Theorem / Thales Theorem
- ~ Lecture 5 : Mid Point Theorem
- ~ Lecture 6 : Angle Bisector Theorem
- ~ Lecture 7 : AAA Similarity Criteria
- ~ Lecture 8 : SSS Similarity Criteria
- ~ Lecture 9 : SAS Similarity Criteria
- ~ Lecture 10 : Result , In two equiangular Triangles the ratio of corresponding sides is same as the ratio of corresponding altitudes and corresponding medians
- ~ Lecture 11 : Result , In two equiangular Triangles the ratio of corresponding sides is same as the ratio of corresponding angle bisectors and perimeter of given Triangle
- ~ Lecture 12 : Questions Practice
- ~ Lecture 13 : Ratio of Areas of Two Similar Triangles
- ~ Lecture 14 : Pythagoras Theorem

=> Coordinate Geometry :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Location of a Point on a Coordinate Plane , DISTANCE FORMULA
- ~ Lecture 3 : Question Practise on Distance Formula
- ~ Lecture 4 : Properties of Various Types of Quadrilateral
- ~ Lecture 5 : Section Formula
- ~ Lecture 6 : Mid Point Formula
- ~ Lecture 7 : Question Practice , CENTROID OF A TRIANGLE
- ~ Lecture 8 : Area of a Triangle, Heron's Formula

=> Trigonometry :

- ~ Lecture1 Trigo Intro
- ~ Lecture2 Trigo Ratios
- ~ Lecture3 NCERT Ex8.1 Trigo Ratios Problem Discussion
- ~ Lecture4 Trigo Specific Angles Discussion
- ~ Lecture5 NCER Ex8.2 Specific Angles Problem Discussion
- ~ Lecture6 More Trigo Angles Problems
- ~ Lecture7 Trigo Complementary AngleDiscussion
- ~ Lecture8 NCERT Ex8.3 Trigo Complementary Angles Problem Discussion
- ~ Lecture9 Trigo Identities
- ~ Lecture10 NCERT Ex8.4 Trigo Identities Problem Discussion
- ~ Lecture11\_HOTS\_Trigo\_Identities\_Problem

=> Some Applications of Trigonometry :

- ~ Lecture12\_Trigo\_Applications\_Introduction
- ~ Lecture13\_Height&Distance\_Problems\_Discussion
- ~ Lecture14\_NCERT\_Ex9.1\_Trigo\_Applications\_Problem\_Discussion

=> Circles :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Understanding Circle, Secant, Circumference, Arc, Segment, Sector
- ~ Lecture 3 : Tangent, No. of Tangents, Length of Tangent, & Theorem 1
- ~ Lecture 4 : Theorem 2 & Theorem 3
- ~ Lecture 5 : THEOREM 4 , THEOREM 5 & THEOREM 6
- ~ Lecture 6 : Some Results
- ~ Lecture 7 : Some important question discussion
- ~ Lecture 8 : Alternate Segment Theorem

=> Construction :

- ~ Lecture 1 : Dividing Line Segment in Ratio m:n
- ~ Lecture 2 : Construction 11.1 Concept
- ~ Lecture 3 Construction 11.1
- ~ Lecture 4 : Construction 11.2
- ~ Lecture 5 : Concept Construction 11.2
- ~ Lecture 6 : Construction 11.2
- ~ Lecture 7 : concept 11.2 Construction

~ Lecture 8 : Construction Tangents to a Given Circle Complete 11.3

## => AREAS RELATED TO CIRCLES :

- ~ Lecture 1 : Circle NCERT Ex10.1
- ~ Lecture 2 : Circle NCERT Ex10.2
- ~ Lecture 3 : Circle NCERT Ex12.1
- ~ Lecture 4 : Circle NCERT Ex12.2
- ~ Lecture 5 : NCERT Ex12.3
- ~ Lecture 6 : Alternate Segment Theorem

## => Surface Area & Volume :

- ~ Lecture 1 : Introduction to Surface Area & Volume
- ~ Lecture 2 : NCERT Solutions Ex13.1 Question 1&2
- ~ Lecture 3 : Surface Area & Volume NCERT Solutions Ex13.1 Question 4
- ~ Lecture 4 : NCERT Solutions Ex13.1 Question 5
- ~ Lecture 5 : NCERT Solutions Ex13.1 Question 6
- ~ Lecture 6 : NCERT Solutions Ex13.1 Question 7&8
- ~ Lecture 7 : NCERT Solutions Ex13.2 Question 1&2
- ~ Lecture 8 : NCERT Solutions Ex13.2 Question 3&4
- ~ Lecture 9 : NCERT Solutions Ex13.3 Question 1
- ~ Lecture 10 : NCERT Solutions Ex13.3 Question 2
- ~ Lecture 11 : NCERT Solutions Ex13.3 Question 3&5
- ~ Lecture 12 : NCERT Solutions Ex13.4 Question 1 & Frustum Introduction
- ~ Lecture 13 : NCERT Solutions Ex13.4 Question 2&3

## => Statistics :

- ~ Lecture 1 Introduction to Statistics
- ~ Lecture 2 Mean Median Mode Discussion
- ~ Lecture 3 Mean Grouped Data Direct Method
- ~ Lecture 4 Assumed Mean Method
- ~ Lecture 5 Step Deviation Method
- ~ Lecture 6 NCERT Ex14.1 Problem Discussion Mean
- ~ Lecture 7 Mode Grouped Data
- ~ Lecture 8 NCERT Ex14.2 Mode & Mean Discussion
- ~ Lecture 9 Median Grouped Data Discussion
- ~ Lecture 10 Summary Mean Median Mode
- ~ Lecture 11 Extra Problem Mean Median Mode
- ~ Lecture 12 NCERT Ex14.3 Mean Median Mode Problem Discussion
- ~ Lecture 13 Graphical Representation Cumulative Frequency
- ~ Lecture 14 NCERT Ex14.2 Ogives Problem Discussion

## => Probability :

- ~ Lecture 1 Probability Introduction
- ~ Lecture 2 Probability Basic Problems Solved
- ~ Lecture 3 Cards Probability Introduction
- ~ Lecture 4 Cards Problem
- ~ Lecture 5 Dice Related Problems
- ~ Lecture 6 Coins Related Problem
- ~ Lecture 7 Optional Problems

# TypeScript Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : JAVASCRIPT

Course link : <https://ineuron.ai/course/TypeScript-Crash-Course>

## Course Description :-

Grab the fundamentals of TypeScript through this crash course. This course helps you to get started with Typescript.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Basics of Typescript
- => Installation of TypeScript

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => TypeScript :
  - ~ Why to learn Typescript
  - ~ Typescript is not what you think
  - ~ How to install Typescript
  - ~ Your first intro to Typescript docs
  - ~ Number, Boolean & Type Inference
  - ~ Don't use ANY in Typescript



# Complete iOS 16 Developer with Swift and 8 Apps

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : IOS

Course link : <https://ineuron.ai/course/Complete-iOS-16-Developer-with-Swift-and-8-Apps>

## Course Description :-

Learn iOS development with SwiftUI and building a lot of apps.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to iOS development
- => Xcode
- => Operators and Range in Swift
- => String and interpolation
- => Array and methods in Array in swift
- => Dictionary in depth in swift
- => Sets in swift programming
- => Tuples in swift
- => Structs in swift
- => Structs Vs Class
- => Building Project 1 - Profile app
- => Project 2 - Custom shape and slots
- => Project 3 - Calculator with animation
- => Project 4 Splash screen
- => Project 5 - Shopping app with multi screen

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

- => Introduction to iOS development :
  - ~ Introduction to iOS development and prerequisites
  - ~ A tour of XCode
  - ~ Hello world in Swift
  - ~ A bit of history of swift with Objective C
- => Getting started with swift :
  - ~ Variables and Constants in Swift
  - ~ Operators and Range in Swift
  - ~ String and interpolation
  - ~ Methods in Strings
  - ~ A caution in type conversion
  - ~ Can user pay Logical Operators
  - ~ Optional binding and forced unwrapping
  - ~ We missed reading the docs
- => More datatypes in swift :
  - ~ Array and methods in Array in swift

- ~ Dictionary in depth in swift
- ~ Sets in swift programming
- ~ Tuples in swift

=> Going all indepth of swift :

- ~ if else and optional unwrapping
- ~ Control flow statements
- ~ Functions in swift programming
- ~ Indepth of Closure 2C autoclosure and escaping
- ~ Enums and indirect enums
- ~ Structs in swift
- ~ Structs Vs Class
- ~ Classes and reference type
- ~ Properties in swift
- ~ Methods in swift

=> Advance swift programming concept :

- ~ Inheritance in swift
- ~ init in depth in swift
- ~ Deinit in swift
- ~ Error handling in swift
- ~ Protocols in swift

=> Building Project 1 - Profile app :

- ~ Zstack 2C HStack and VStack
- ~ Create a new app in XCode
- ~ Getting started with Zstack and VStack
- ~ Moving into VStack
- ~ Nested Stacks in swift UI
- ~ Finishing our first app

=> Project 2 - Custom shape and slots :

- ~ Theory behind custom shapes in iOS
- ~ From figma to XCode shape
- ~ State 2C rawValue and Identifiable
- ~ More on State and HStack
- ~ Getting button in our app
- ~ Finishing up slot machine game

=> Project 3 - Calculator with animation :

- ~ RawValue in swift
- ~ Starting a calculator project - assets
- ~ Defining Model for calculator
- ~ Getting keys sorted out for calculator
- ~ Animation in swift ui
- ~ Adding buttons for calculator
- ~ Learn to calculate element width and height
- ~ Loading up views on home screen
- ~ Finishing up the calculator logic part

=> Project 4 Splash screen :

- ~ Getting started with Splash screen
- ~ Finishing up a splash screen

=> Project 5 - Shopping app with multi screen :

- ~ Demo of Shopping app with Navigation
- ~ Importing all assets of fruits
- ~ Building on boarding screen with navigation
- ~ Models for fruit and near you
- ~ Handling the fruit card
- ~ Horizontal scroll view
- ~ Passing value from one screen to another
- ~ Design detail view part 1
- ~ Counter in detail screen
- ~ Vertical scroll view
- ~ Assemble fruit cart app
- ~ Resolving minor UI issue

=> Project 6 - Building LinkedIn UI clone :

- ~ What we will build - LinkedIn
- ~ Search bar component
- ~ Models in linkedin UI
- ~ Each connection request
- ~ Building my Network screen
- ~ Making home cards
- ~ Home screen top view
- ~ Building Home Screen
- ~ Launch linkedin UI in simulator

=> Project 7 - Todo App - Read the docs :

- ~ What are user defaults
- ~ What is Codable protocol
- ~ Model with Identifiable and Codable
- ~ What are ObservableObject and Published
- ~ UserDefaults with unique key
- ~ Get values from UserDefaults
- ~ CRUD operations in Todo list
- ~ DispatchQueue in depth
- ~ Navigation View and Link
- ~ State management in swift ui
- ~ Take user input and add it to Model

- ~ Adding Todo 27s on Home screen
- ~ Finishing up todo app with gesture implementation

=> Project 8 - Handling API and building pokemon app :

- ~ What is API and formatting
- ~ Create a model for API response
- ~ Fetching data from API endpoint
- ~ List and async calls
- ~ Kingfisher - Third party packages
- ~ Install third party packages
- ~ What are extensions in swift
- ~ Issues in Data and API call
- ~ Creating a data extension
- ~ Using KFImage
- ~ Gridviews and LazyVStack
- ~ Debugging the pokemon app

# Salesforce Developer

---

Topic Name : SALESFORCE

Sub-topic Name : SALESFORCE DEVELOPER

Course link : <https://ineuron.ai/course/Salesforce-Developer>

## Course Description :-

You may master the fundamentals of Salesforce programming in this Salesforce developer course. This tutorial gives you hands-on experience with Apex Programming, Triggers, and Form Building on the Visualforce website.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Apex
- => Collections
- => List Class and Method
- => Set Class and Methods
- => Select Option Class
- => SOQL ( Object Query Language)
- => Outbound Email Services
- => Single Email Message
- => Sending Pdf attachment
- => Sending Email Template
- => Attaching VF page as an attachment
- => Attaching the Email to activities/li>
- => Trigger Events
- => Trigger Context variables
- => Insert Triggers
- => Trigger. New in before insert and after Insert .

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Course Introduction :
  - ~ MVC Architecture
  - ~ Comparision of MVC to APPS
  - ~ OOPS Basics
  - ~ Lexical
  - ~ Tokens
  - ~ Identifiers
  - ~ Variables
  - ~ Data Types
  - ~ Methods
  - ~ Access Modifiers
  - ~ Comparission between Method/Function and Procedure
  - ~ Classes
  - ~ Creation of Variables
  - ~ Getter Methods
  - ~ Setter Methods
  - ~ Creation of Methods
  - ~ with Sharing
  - ~ With Out Sharing
  - ~ Creation of Objects

- ~ Reference Variables
- ~ Constructors
- ~ Class Scope
- ~ Static Variables
- ~ Static Methods
- ~ Static Blocks
- ~ Final Variables
- ~ Final Static Variables
- ~ Arrays
- ~ Advantages and Disadvantages of Arrays

=> Apex :

- ~ Collections
- ~ List Class and Method
- ~ Set Class and Methods
- ~ Select Option Class
- ~ SOQL ( Object Query Language)
- ~ Group By/li>
- ~ Having
- ~ Limit
- ~ Parent to Child object Query
- ~ Child to Parent object Query
- ~ Aggregate Result
- ~ Compression of Database. Query and static query
- ~ Governing limits for SOQL
- ~ DML operations
- ~ Database. DML operations
- ~ Compression of DML with Database.DML
- ~ Database. Save Point
- ~ Database. Rollback operations
- ~ Inheritance in Class
- ~ Overriding classes
- ~ Working with workbench
- ~ Writing a test classes
- ~ SOQL Governing limit
- ~ DML governing limit

=> Email Services :

- ~ Outbound Email Services
- ~ Single Email Message
- ~ Sending Pdf attachment
- ~ Sending Email Template
- ~ Attaching VF page as an attachment
- ~ Attaching the Email to activities/li>
- ~ Creating a PDF from futur

=> Schedule Apex :

- ~ Implementing Schedule Apex
- ~ Cron Trigger
- ~ Invoking Batch Apex
- ~ Invoking Future Methods from Schedule Apex
- ~ Invoking the Callouts from Schedule Apex
- ~ Sending Email from Schedule Apex
- ~ Test Classes
- ~ Governing Limits

=> Triggers :

- ~ Trigger Events
- ~ Trigger Context variables
- ~ Insert Triggers
- ~ Trigger. New in before insert and after Insert .
- ~ DML in before insert and after Insert.

# Azure Databricks

---

Topic Name : CLOUD

Sub-topic Name : AZURE

Course link : <https://ineuron.ai/course/Azure-Databricks>

## Course Description :-

Building a solution architecture for a data engineering solution using Azure Databricks, Azure Data Lake Gen2, Azure Data Factory and Power BI, creating and using Azure Databricks service and the architecture of Databricks within Azure, creating, configuring and monitoring Databricks clusters, cluster pools and jobs, passing parameters between notebooks as well as creating notebook workflows.

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => learn how to build a real world data project using Azure Databricks
- => learn how to create notebooks, dashboards, clusters, cluster pools and jobs in Azure Databricks
- => learn how to create Azure Data Factory triggers to schedule pipelines as well as monitor them.

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Azure Databricks :
  - ~ Spark Basics Preview
  - ~ Why spark is difficult
  - ~ Why databricks in cloud?
  - ~ How to save databricks demo cost
  - ~ demo overview Preview
  - ~ Demo provision databricks, clusters and workbook
  - ~ demo mount data lake to databricks DBFS
  - ~ Demo Explore, Analyze, Clean, Transform and load data in databricks
  - ~ azure databricks cluster
  - ~ azure databricks other important components
  - ~ databricks monitoring

# DSA for FAANG preparation with Python and JavaScript Tech Neuron

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/DSA-for-FAANG-preparation-with-Python-and-JavaScript-Tech-Neuron>

## Course Description :-

A comprehensive chase to excel any interview for the Data Structures and Algorithms. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

## Course Features :-

- => Free LCO DSA Bundle
- => 18 hrs live support all seven day
- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Analysis in Algorithms
- => Data Structure Introduction
- => Array Data Structure
- => Interview Question on array
- => Recursion in depth
- => Divide and Conquer algorithm
- => Applications of Divide and Conquer
- => Linked List Data Structure
- => Interview Question on Linked List
- => Circular Linked List
- => Doubly Linked List
- => Skip List
- => Stack and Queue
- => Interview Question on Stack and Queue
- => Hashing Data Structure
- => Collision Resolution Techniques
- => Tree Data Structure
- => Tree Traversal
- => Binary Search Tree
- => Height Balanced Tree: AVL Tree

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

=> Anurag Tiwari :

~ Hey, I am Anurag Tiwari, a developer at learncodeonline. We have built a scalable system handled by 300K users on a daily basis. I'm a software developer who constantly seeks innovative solutions to everyday problems. I have been teaching students for the last 24 months.

## Curriculum details :-

=> Introduction :

- ~ Course Overview

=> Analysis in Algorithms :

- ~ Why we need Data structures and algorithms
- ~ Introduction to algorithms and its analysis : Time and Space Complexity
- ~ Asymptotic Notation: Big O, Omega and Theta Notation
- ~ Recurrence Relation Solving: Substitution, Recursive Tree and Master's Theorem

=> Data Structure Introduction :

- ~ Memory Process - Stack and Heap
- ~ Physical and logical data structures
- ~ Abstract data types

=> Array Data Structure :

- ~ Introduction to arrays
- ~ Concept of 1D and 2D array (row-major order and column-major order)
- ~ Searching algorithm: linear, binary, ternary search
- ~ Concept of inplace and outplace sorting algorithm
- ~ Concept of stable and unstable sorting algorithm
- ~ Sorting algorithm: comparison(selection, bubble, insertion, quicksort, mergesort, heapsort, shellsort)
- ~ Sorting algorithm: Non-comparison(count sort, bucket sort, radix sort)

=> Interview Question on array :

- ~ Rotation of an array
- ~ Finding of missing number in an array
- ~ Division of two integers without using division operator
- ~ Search in rotated array
- ~ Target triplet
- ~ Stock buy sell to maximize profit

=> Recursion in depth :

- ~ Introduction to recursion
- ~ Tracing the recursion tree
- ~ Types of recursion
- ~ Complex recursion tree
- ~ Classic Tower of Hanoi problem

=> Divide and Conquer algorithm :

- ~ Introduction to Divide and Conquer

=> Applications of Divide and Conquer :

- ~ Finding of maxima and minima
- ~ Finding of power of an element
- ~ Binary Search
- ~ MergeSort
- ~ QuickSort
- ~ Selection Procedure
- ~ Finding of number of inversions
- ~ Strassens' matrix multiplication

=> Linked List Data Structure :

- ~ Introduction to linked list
- ~ Insertion of a node(beginning, end and at any position) in linked list
- ~ Deletion of a node(beginning, end and at any position) in linked list
- ~ Searching of a node in linked list

=> Interview Question on Linked List :

- ~ Reversal of a node in linked list
- ~ Count of all nodes in linkedlist
- ~ Floyd's cycle detection algorithm
- ~ Merge two linked list

=> Circular Linked List :

- ~ Circular Linked List Theory
- ~ Insertion of a node in circular linked list
- ~ Traversal of a node in circular linked list
- ~ Deletion of a node in circular linked list
- ~ Count of number of nodes in circular linked list
- ~ Conversion of linked list to circular linked list

=> Doubly Linked List :

- ~ Doubly Linked List Theory
- ~ Insertion of a node in doubly linked list
- ~ Traversal of a node in doubly linked list
- ~ Deletion of a node in doubly linked list

=> Skip List :

- ~ Introduction to skip list
- ~ Build-in skip list
- ~ Search in skip list
- ~ Insertion in skip list
- ~ Deletion in skip list

=> Stack and Queue :

- ~ Stack: Push and Pop operation
- ~ Implementation of Stack using array and linked list
- ~ Queue concept theory



- ~ Implementation of Queue using array and linked list
- ~ Circular Queue theory
- ~ Implementation of Circular Queue

=> Interview Question on Stack and Queue :

- ~ Stack using queue conceptual understanding
- ~ Implementation of stack using queue
- ~ Queue using stack conceptual understanding
- ~ Implementation of queue using stack
- ~ Valid brackets
- ~ Stock Spanning

=> Hashing Data Structure :

- ~ Introduction to Hashing Data Structure
- ~ Hash Function and its types

=> Collision Resolution Techniques :

- ~ Chaining
- ~ Open Addressing: Linear Probing, Quadratic Probing, Double Hashing, Perfect Hashing, Consistent Hashing
- ~ Application: Bloom Filters
- ~ Two Sum Problem

=> Tree Data Structure :

- ~ Introduction to Binary Tree
- ~ Complete Binary Tree and almost complete binary tree
- ~ Full binary tree and representation using array and linked list

=> Tree Traversal :

- ~ Introduction to tree traversal
- ~ Inorder Traversal
- ~ Preorder Traversal
- ~ Postorder Traversal

=> Binary Search Tree :

- ~ Introduction to Binary Search Tree
- ~ Insertion and Deletion in BST
- ~ Inorder traversal in BST gives sorted array
- ~ Searching in Binary Search Tree
- ~ Deletion in Binary Search Tree

=> Height Balanced Tree: AVL Tree :

- ~ Introduction: Why AVL Tree?
- ~ Creation of an AVL Tree
- ~ Insertion in AVL Tree
- ~ Searching in AVL Tree
- ~ Deletion in AVL Tree

=> Height Balanced Tree: Red Black Tree :

- ~ Introduction: Why Red Black Tree?
- ~ Properties of Red Black Tree
- ~ Creating of Red Black Tree
- ~ Insertion Rules in Red Black Tree
- ~ Searching in Red Black Tree
- ~ Deletion in Red Black Tree

=> B and B+ Tree: Usage in Databases :

- ~ Creation of B and B+ Tree
- ~ Insertion in B and B+ Tree
- ~ Searching in B and B+ Tree
- ~ Deletion in B and B+ Tree

=> Interview Question on Tree :

- ~ Checking of whether the tree is symmetric or not
- ~ Count of number of possible BSTs in a given number of nodes
- ~ Catalan number concept to find the number of BST
- ~ Level order traversal of a tree
- ~ Flip or inverse of a binary tree
- ~ Same tree problem
- ~ Inorder iterator
- ~ Binary Tree Zigzag level order traversal

=> Graph Traversal Algorithms :

- ~ Introduction to Graph Traversal Algorithms
- ~ Introduction to Depth First Search
- ~ DFS Psuedocode and illustration using an example
- ~ DFS Coding Implementation
- ~ Introduction to Breadth First Search
- ~ BFS Psuedocode and illustration using an example

=> Interview Questions on Graph :

- ~ Clone of a graph
- ~ DFS and Cycle detection with University course problem
- ~ Island problem

=> Heap Data Structure :

- ~ Introduction to Heap Data Structure
- ~ Maxheap and Minheap Overview
- ~ Insertion in Minheap
- ~ Deletion in Minheap
- ~ Creation of Minheap
- ~ Mathematical derivation to analyse the complexity of creation of minheap
- ~ HeapSort algorithm and why it is not stable algorithm

=> Interview Based Question on Heap Data Structure :

- ~ *Maximum Product of three numbers in an array*
- ~ *Finding of K-closest points from an origin*

=> Greedy Algorithm :

- ~ *Introduction to greedy algorithm*

=> Application of greedy algorithm :

- ~ *Fractional Knapsack Problem*
- ~ *Minimum Spanning Tree: Kruskal and Prim's Algorithm*
- ~ *Single Source Shortest Path: Dijkstra's algorithm*
- ~ *Huffman Coding*
- ~ *Optimal Merge Pattern*
- ~ *Job Sequencing with Deadline*

=> Dynamic Programming :

- ~ *Introduction to Dynamic Programming*
- ~ *Overlapping subproblem in dynamic programming*
- ~ *Tabulation in dynamic programming*
- ~ *Memoization in dynamic programming*

=> Application of Dynamic Programming :

- ~ *Fibonacci Series*
- ~ *Longest Common Subsequence*
- ~ *0/1 Knapsack Problem*
- ~ *Sum of subset*
- ~ *All Pair Shortest Path: Floyd Warshall Algorithm*
- ~ *Bellman Ford Algorithm*

=> Interview Problems on Dynamic Programming :

- ~ *Knapsack - Coke, Pepsi, Redbull*
- ~ *Largest sum of subset*
- ~ *Coin change problem*
- ~ *Largest sum*
- ~ *Minimum path to reach target*

=> String Matching Algorithms :

- ~ *Introduction to String matching algorithms*
- ~ *Naive String Matching algorithms*
- ~ *Rabin Karp Algorithm*
- ~ *Kuth-Morris-Pratt(KMP) Pattern Matching*

=> Interview Problems on String :

- ~ *Word in a sentence*
- ~ *Inplace duplicates*
- ~ *Longest substring*
- ~ *Palindrome makes and breaks*

=> NP-Hard and NP-Complete Problem :

- ~ *NP-Hard Problem*
- ~ *NP-Complete Problem*

=> Approaching Design :

- ~ *Understanding and clarification*
- ~ *Business usecase of the problem and knowing the consumers*
- ~ *Iron out the Functional requirements*
- ~ *Importance of discussing the trade-offs based on the usecase in picture*
- ~ *Mastering the art of selling design*
- ~ *Data model approaches and fitment*
- ~ *LLD modelling and future readiness of design*
- ~ *Explaining the features of design like adherence to proper design patterns*

=> Introduction to System Design :

- ~ *Introduction to system design*
- ~ *Importance of architecture*
- ~ *Distinction between HLD and LLD*
- ~ *Importance of data modelling*
- ~ *Importance of documentation in design*

=> Practicing some real designs :

- ~ *Rate limiting*
- ~ *Uber riders app*
- ~ *Whatsapp messaging*
- ~ *food delivery app building*
- ~ *Booking app building*
- ~ *Video streaming systems*
- ~ *Q&A*

# Web Automation Foundations

---

Topic Name : TESTING

Sub-topic Name : AUTOMATION TESTING

Course link : <https://ineuron.ai/course/Web-Automation-Foundations>

## Course Description :-

During this community sessions we are going to have lot of Q&A sessions on Automation Career Guidance, Interview preparations , Web Automation using Selenium, TestNG and Maven too.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Roadmap to learn Automation Testing
- => Different tools for automation in each category
- => How to write first Automation Script
- => XPath and Css Selector
- => Automating CRM End-to-End Scenarios Using Selenium

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Mukesh Otvani :

~ Myself Mukesh Otvani having 10 years of experience in Automation Testing and worked with Dell International and SAP Labs India. I am also passionate teacher, mentor, have been into teaching for 8 years now and running a YouTube channel with 137000 subscribers and 480+ videos on different tools and libraries.

## Curriculum details :-

- => Day1 :
  - ~ Getting Started With Automation Testing - Orientation Program
- => Day2 :
  - ~ Roadmap to learn Automation Testing
- => Day3 :
  - ~ Different tools for automation in each category
- => Day4 :
  - ~ Q & A Session
- => Day5 :
  - ~ Web Automation using Selenium - Download and Installation
- => Day6 :
  - ~ How to write first Automation Script
- => Day7 :
  - ~ Selectors in detail
- => Day8 :
  - ~ XPath and Css Selector
- => Day9 :
  - ~ Automating CRM End-to-End Scenarios Using Selenium
- => Day10 :
  - ~ Interview Preparation For Automation - Session 1
- => Day11 :
  - ~ Interview Preparation For Automation - Session 2
- => Day12 :
  - ~ Interview Preparation For Automation - Session 3

=> Day13 :

~ Interview Preparation For Automation - Session 4

=> Day14 :

~ Interview Preparation For Automation - Session 5

=> Day15 :

~ Testing - Session 1

=> Day16 :

~ Testing- Session 2

=> Day17 :

~ Testing- Session 3

=> Day18 :

~ Maven- Session 4

=> Day19 :

~ Q & A Session

# Power BI Projects

---

Topic Name : DATA ANALYTICS

Sub-topic Name : POWER BI PROJECTS

Course link : <https://ineuron.ai/course/Power-BI-Projects>

## Course Description :-

This course aims to make you aware of the project on a real scenario basis. Solve real-world business challenges using reports, visualizations, and other analytics tools that allow you to gather and exchange data from diverse domains.

## Course Features :-

- => Roadmap
- => Quizzes
- => Assignment
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Gain knowledge from end-to-end project reports in PowerBI
- => Industry-level experience
- => Power pivots, slicers, power view, data analysis expressions

## Requirements :-

- => Prior knowledge of PowerBI tool
- => A system with internet connection
- => Microsoft power BI desktop account
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Introduction :

~ Overview Preview

=> Spend :

~ Project overview Preview  
~ Data load  
~ Table report  
~ Matrix report  
~ Funnel chart  
~ Pie chart  
~ Scatter plot  
~ QnA

=> Product :

~ Project overview  
~ Load data  
~ Create measure  
~ Card total units sold  
~ Donut chart  
~ World flag

=> Acquisition :

~ Project overview  
~ Load data  
~ Create measure  
~ Card  
~ Top performer  
~ Region sales  
~ Sales last year  
~ Profit comparison  
~ Moving average

# C++ Bootcamp

---

Topic Name : PROGRAMMING

Sub-topic Name : C++

Course link : <https://ineuron.ai/course/C++-Bootcamp>

## Course Description :-

Ultimate modern C++ Bootcamp. A modern approach to understand C++.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to CPP
- => Getting Started with CPP
- => Basics but indepth of CPP
- => Functions in CPP
- => Object Oriented Programming
- => Smart Pointers in CPP

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction to CPP :

- ~ CPP20 A note
- ~ CPP20 section1
- ~ Welcome to Cpp bootcamp
- ~ Prerequisite and tools for cpp
- ~ Understand the entry point of hello world
- ~ Compare the 2 hello world
- ~ Version history and official documentation of cpp

=> Getting Started with CPP :

- ~ CPP20 section2
- ~ Return type and comments
- ~ Redefining program in cpp
- ~ What is namespace in cpp
- ~ First iteration of program
- ~ Can I name that
- ~ Get the color and assignment

=> A Little fast pace CPP :

- ~ CPP20 section3
- ~ Your first introduction to pointers
- ~ Reference is the actual tough thing in cpp
- ~ Cpp array are different with pointers
- ~ A formal introduction to integers
- ~ Conditionals and ternary
- ~ Conditionals as switch
- ~ While and do while loops
- ~ Introduction to for and range based for loops
- ~ Loop with pointers and shortcuts

=> Basics but indepth of CPP :

- ~ CPP20 section4
- ~ Always use float with caution

- ~ *Why always divide by zero for try catch block*
- ~ *Sneek peek to functions in cpp*
- ~ *linkers qualifiers prefix and postfix*
- ~ *Basics of operations on cpp*
- ~ *Logical AND OR and NOT*
- ~ *bitwise operation in cpp*
- ~ *Memory leaks in cpp*

=> More datatypes in CPP :

- ~ *CPP20 section5*
- ~ *Get started with structs in cpp*
- ~ *Enums and Preprocessors*
- ~ *A challenge to strongly types language*
- ~ *Heap and Stack memory with a version discussion*

=> Functions in CPP :

- ~ *CPP20 section6*
- ~ *Detailed introduction to functions*
- ~ *How to create a header file in cpp*
- ~ *Your first introduction to templates*
- ~ *What are functional pointers*
- ~ *nullptr saves the day*
- ~ *Factorial and recursion are close friend*
- ~ *Lets talk about MACROS*
- ~ *Variadic templates and recursion*

=> Object Oriented Programming :

- ~ *CPP20 section7*
- ~ *A design example*
- ~ *Get started with class and objects*
- ~ *Getters and Setters for a data member*
- ~ *Method separation and const qualified methods*
- ~ *Constructor destructor and rule of 3*
- ~ *Disable the constructor*
- ~ *THIS is not easy in cpp*

=> Little more OOPS :

- ~ *CPP20 section8*
- ~ *Inheritance is my favourite*
- ~ *Base class Derived class and overriding*
- ~ *Friend keyword come with caution*
- ~ *Multiple Inheritance*
- ~ *polymorphism and virtual*

=> Smart Pointers in CPP :

- ~ *CPP20 section9*
- ~ *What are smart pointers*
- ~ *Unique pointers and issues*
- ~ *Shared pointers in smart pointers*
- ~ *Weak pointers in smart pointers*

=> Move Semantics file & lambda :

- ~ *CPP20 section10*
- ~ *Move semantics Lvalue and Rvalue*
- ~ *Vectors - Dynamic array from STD template library*
- ~ *Lambda - a small hello*
- ~ *Create, rename and delete files*
- ~ *Reading and writing into files and MODES*

=> STL - Standard Template Library :

- ~ *CPP20 section11*
- ~ *Introduction to STL and generic programming*
- ~ *Main components in STL*
- ~ *Functors in STL*
- ~ *SORT algorithms in STL*
- ~ *SEARCH algorithms in STL*
- ~ *Partition and Stable partition in STL*

=> STL - a little more :

- ~ *CPP20 section12*
- ~ *Revisiting vectors in STL*
- ~ *List in STL*
- ~ *Queue and priority queue in STL*
- ~ *Deque in STL*
- ~ *Stack in STL and assignment*
- ~ *Sets and MultiSets in STL*
- ~ *MAPS and assignment*

# Pro Aptitude - C Language

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Aptitude---C-Language>

## Course Description :-

This course is designed mostly for C test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> C Theoretical Test

=> C Practical Test

=> C Aptitude Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> C Coding Test :

~ C Test 1

~ C Test 2

~ C Test 3

~ C Test 4



# Java Foundations

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Java-Foundations>

## Course Description :-

This course has been designed to help you become a complete and professional Java engineer at the conclusion of the course, rather than only teaching essential Java skills. To guarantee that you grasp the Java language, the course has been designed to be very thorough, covering the majority of Java language features and explaining them in great detail. Tons of best practises and design ideas are described and illustrated in code to guarantee you are industry-ready and can create well-designed, professional code.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Variables and Operators
- => Conditional statements and Loops
- => Arrays
- => Class and Objects
- => Inheritance
- => Abstraction
- => Package and Interface
- => Exception Handling
- => Multithreading
- => Collection API
- => Lambda Expression

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Navin Reddy :

~ I am Corporate Java trainer. Since past few years successfully trained many professionals at JP Morgan, Accenture, Polaris and L&T infotech. My youtube channel "Telusko" presently has 1.7 million subscribers. Passionate about Java Technology for over a decade and moved on as a corporate trainer. I am certified blockchain developer and Currently, building Applications running on Blockchain (dapps).

## Curriculum details :-

- => Day 1 :
  - ~ Introduction to Java and Setup
- => Day 2 :
  - ~ Variables and Operators
- => Day 3 :
  - ~ Conditional statements and Loops
- => Day 4 :
  - ~ Arrays
- => Day 5 :
  - ~ Class and Objects
- => Day 6 :
  - ~ Inheritance, Encapsulation
- => Day 7 :
  - ~ Abstraction, Polymorphism

=> Day 8 :

~ *Package and Interface*

=> Day 9 :

~ *Exception Handling*

=> Day 10 :

~ *Multithreading*

=> Day 11 :

~ *Collection API*

=> Day 12 :

~ *Lambda Expression*

# Publishing Custom Cocoapads

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Publishing-Custom-Cocoapads>

## Course Description :-

This course will help you to publish custom Cocoapads.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => How to publish your own cocoapods
- => Installing cocoapods
- => Creating and exploring cocoapods to be published
- => Editing podsec and github push
- => Writing custom function for cocoapods
- => Writing a test case for cocoapods
- => Finally 2C lets push our first cocoapod

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Cocoapads :

- ~ How to publish your own cocoapods
- ~ Installing cocoapods
- ~ Creating and exploring cocoapods to be published
- ~ Editing podsec and github push
- ~ Writing custom function for cocoapods
- ~ Writing a test case for cocoapods
- ~ Finally 2C lets push our first cocoapod

# Tableau Course

---

Topic Name : DATA ANALYTICS

Sub-topic Name : TABLEAU

Course link : <https://ineuron.ai/course/Tableau-Course>

## Course Description :-

The demand for effective, accessible, and actionable interfaces grows daily as enterprises gather more and more critical data. New business intelligence tools like Tableau drastically decrease the time and technical knowledge necessary to extract insights from data and present them in a style that is easily understood by the end user. In this course, you will learn to harness the power of Tableau for every step in the dashboard building process.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Tableau Vs Excel
- => Pie Chart
- => Data Types in Tableau
- => Scatter Plot

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Pawan Lalwani :

~ Pawan is a highly skilled and self motivated trainer who has expertise in various business intelligence tools like Power BI, Tableau and Microsoft Excel. He comes with 10 years of experience in training individuals in different industry sectors like Banking, Finance, Healthcare, IT, Automobile, Manufacturing and Pharmaceutical.

## Curriculum details :-

=> Introduction to Tableau :

- ~ Tableau Introduction
- ~ Download and Install Tableau
- ~ Tableau Vs Excel

=> Charts - 1 :

- ~ Column Chart
- ~ Horizontal Bar Chart
- ~ Stacked Column Chart
- ~ Stacked Bar Chart
- ~ Keep Only, Exclude
- ~ Keep Only, Exclude2, Normal
- ~ Publish to Tableau Public

=> Charts - 2 :

- ~ Pie Chart
- ~ Multiple Pie Chart
- ~ TreeMap\_Editing
- ~ Packed Bubble Chart
- ~ Word Cloud OR Word Map
- ~ Formatting payal

=> Charts - 3 :

- ~ Data Types in Tableau
- ~ Filled Map
- ~ Symbol Maps
- ~ India Map
- ~ Histogram

=> Charts - 4 :

- ~ Text Table

- ~ *Text Table with Multiple Measures*
- ~ *Measure Names and Measure Values*
- ~ *Line Chart*
- ~ *Line Chart with Multiple Measures*
- ~ *Discrete Vs Continuous Line Chart*
- ~ *Discrete Vs Continuous*

=> Charts - 5 :

- ~ *Lollipop Chart*
- ~ *Line Vs Column Chart*
- ~ *Dual Axis Chart*
- ~ *Column vs Shapes*
- ~ *Bar in Bar Chart*

=> Charts - 6 :

- ~ *Calculated fields*
- ~ *Conditional Column Chart*
- ~ *Column chart with Shapes based on condition*
- ~ *Conditional Maps*

=> Charts - 7 :

- ~ *Map with Pie Chart*
- ~ *Map with WMS*

=> Charts - 8 :

- ~ *Funnel Chart*
- ~ *Advanced Funnel Chart*
- ~ *Calendar*
- ~ *Dumbbell Chart*
- ~ *Donut Chart*
- ~ *Multiple Donut Chart*

=> Charts - 9 :

- ~ *Bullet Chart 1*
- ~ *Bullet Chart 2*
- ~ *Table Calculations Part 1*
- ~ *Table Calculations - Compute Using - Part 2*
- ~ *Table Calculations - Relative - Part 3*
- ~ *Bump Chart*
- ~ *Bump Chart with Circle*
- ~ *100 Percent Stacked Column Chart*

=> Charts - 10 :

- ~ *Scatter Plot*
- ~ *Scatter Plot with Images OR Shapes*
- ~ *Bubble Chart*
- ~ *Animation - Column Chart*
- ~ *Animation - Line Chart*
- ~ *Animation - Column vs Line Chart*

=> Charts - 11 :

- ~ *Heat Maps*
- ~ *Heat Map with Shapes*
- ~ *Heat Map with Conditional Formatting*
- ~ *Pareto Chart*
- ~ *Rounded Bar Chart*

# Full Stack Blockchain Development Tech Neuron

---

Topic Name : BLOCKCHAIN

Sub-topic Name : BLOCKCHAIN MASTERS

Course link : <https://ineuron.ai/course/Full-Stack-Blockchain-Development-Tech-Neuron>

## Course Description :-

Full Stack Blockchain Development course is a live mentor-led certification program with by iNeuron. In this course you will learn the entire stack required to work in Permissionless Blockchain development. This course focuses on latest Blockchain industry standards like Ethereum Blockchain, Solidity, Decentralized Autonomous Organisations, Decentralized Finance, Non Fungible Tokens, Polygon Network, Polkadot Blockchain, Oracles along with complete development stack in Javascript and many more Blockchain concepts.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Web 1.0 vs Web 2.0 vs Web 3.0
- => What is Blockchain technology?
- => Bitcoin Blockchain
- => Ethereum Blockchain
- => Solidity
- => Oracles
- => DAO
- => DeFi
- => NFT
- => Layer 2 Blockchain
- => Truffle Suite
- => Hardhat
- => Polkadot

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Course Introduction :
  - ~ Course overview
  - ~ A brief history of internet technologies
  - ~ Web 1.0 vs Web 2.0 vs Web 3.0
  - ~ What is Blockchain technology?
  - ~ Why do we need Blockchain technology?
  - ~ The connected world and the Blockchain: A disruptive computing paradigm
  - ~ Centralized vs Decentralized networks
  - ~ Distributed Systems overview
- => Web Development :
  - ~ What is Web Development?
  - ~ Client-Server Architecture
  - ~ What are APIs?
  - ~ What is Front-end web development?
  - ~ What is Back-end web development?
  - ~ Components of Full-Stack Web Development Applications
- => HTML :
  - ~ How do websites work?
  - ~ Preview
  - ~ HTML vs CSS vs Javascript

- ~ HTML files
- ~ Doctype & HTML Boilerplate
- ~ Spaces & Line Breaks
- ~ Heading Tag
- ~ Paragraph & Pre Tag
- ~ Difference between Elements, Attributes & Tags
- ~ Comments
- ~ Useful Tags
- ~ Nesting of Tags
- ~ Extensions in HTML
- ~ Live Server in VSCode
- ~ Formatting Tags
- ~ Article in HTML
- ~ Time & Address Tag
- ~ Quote & Cite
- ~ Strike
- ~ Progress Bar
- ~ Anchor Tag Styling
- ~ Image Tag
- ~ HTML Table
- ~ List
- ~ Input Tags,iframe
- ~ Forms
- ~ Video & Audio
- ~ iframe
- ~ Embed pdf
- ~ Maps
- ~ Symbols
- ~ Meta Tags
- ~ SVG
- ~ Emoji

=> CSS :

- ~ CSS Introduction
- ~ Inline vs Internal vs External
- ~ Priority between Inline, Internal & External
- ~ Multiple Properties in Single Element
- ~ Types of Selectors
- ~ Priority between Id, Class & Element
- ~ Comments
- ~ Colors
- ~ Background
- ~ Border
- ~ Height & Width
- ~ Padding
- ~ Margin
- ~ Box Model
- ~ Text Properties
- ~ Anchor Tag Styling
- ~ Fonts
- ~ Cursor
- ~ !Important in CSS
- ~ Box Shadow
- ~ Opacity
- ~ Filter
- ~ Gradient
- ~ Overflow
- ~ List
- ~ Tables
- ~ Box Sizing
- ~ Inherit & Initial
- ~ Object Fit
- ~ Pseudo Classes
- ~ Pseudo Elements
- ~ Display
- ~ Position
- ~ Z-Index
- ~ Floats
- ~ 2D Transform
- ~ Transitions
- ~ Flex
- ~ Flex Direction & Wrap
- ~ Justify & Align in Flex
- ~ Order in Flex
- ~ Grow & Basis in Flex
- ~ Align Items in Flex
- ~ Grids
- ~ Rows, Columns & Gap in Grids
- ~ Justify & Align in Grids
- ~ CSS Validator (Final Video)

=> Javascript :

- ~ Introduction
- ~ Running Javascript in Browser
- ~ Console
- ~ Strings & Numbers
- ~ var, let & const
- ~ Data Types
- ~ Type Conversions

- ~ Arithmetic Operators
- ~ Assignment Operator
- ~ Comparison Operator
- ~ Logical Not, Or and And
- ~ Swap Numbers
- ~ String Handling
- ~ String Searching
- ~ Arrays
- ~ Objects
- ~ Dates
- ~ Maths
- ~ If & Else
- ~ Challenge - If & Else
- ~ Switch Case
- ~ Challenge - Switch Case
- ~ JS Loops
- ~ For Loops
- ~ Nested Loops
- ~ Break & Continue
- ~ Arrays, Strings & Objects
- ~ For-in
- ~ For-of
- ~ While Loops
- ~ Do while Loops
- ~ Loops Exercises
- ~ Functions
- ~ Variable Scopes in Functions
- ~ Nested Functions
- ~ Parameters & Arguments
- ~ How function is useful?
- ~ Return in Function
- ~ Anonymous Functions
- ~ Calculator Exercise
- ~ Arrow Functions
- ~ forEach
- ~ maps
- ~ String Literals
- ~ Filter, Reduce & Every
- ~ Spread Operator
- ~ Challenge
- ~ Window & Document
- ~ Document Access
- ~ innerText & innerHTML
- ~ HTML Calculator
- ~ Query Selector
- ~ Styling in JS
- ~ Advance DOM Manipulation
- ~ Events
- ~ Basic Events
- ~ Time Events
- ~ Pop-up Boxes
- ~ Error Handling
- ~ Form Validation
- ~ Asynchronous JS
- ~ this keyword
- ~ useStrict
- ~ Hoisting
- ~ Local Storage
- ~ Session Storage
- ~ Cookies
- ~ Cookies vs Local Storage vs Session Storage
- ~ JSON vs Object literals
- ~ API
- ~ Fetching
- ~ Methods & Status Codes
- ~ Post Method
- ~ Put Method
- ~ Guess the Number
- ~ Generators
- ~ Regex

#### => The JavaScript Standard Library :

- ~ The JavaScript Standard
- ~ Sets and Maps
- ~ Typed Arrays and Binary Data
- ~ Pattern Matching with Regular Expressions
- ~ Dates and Times
- ~ Error Classes
- ~ JSON Serialization and Parsing
- ~ The Internationalization API
- ~ The Console API
- ~ URL APIs
- ~ Timers

#### => Iterators and Generators :

- ~ What are Iterators and Generators?
- ~ How Iterators Work?
- ~ Implementing Iterable Objects
- ~ Generators



~ *Advanced Generator Features*

## => Asynchronous JavaScript :

- ~ *What is Asynchronous JavaScript?*
- ~ *Asynchronous Programming with Callbacks*
- ~ *Promises*
- ~ *Async and await*
- ~ *Asynchronous Iteration*

## => Working with Web Browsers :

- ~ *JavaScript in Web Browsers*
- ~ *Web Programming Basics*
- ~ *Events*
- ~ *Scripting Documents*
- ~ *Scripting CSS*
- ~ *Document Geometry and Scrolling*
- ~ *Web Components*
- ~ *SVG: Scalable Vector Graphics*
- ~ *Audio APIs*
- ~ *Location, Navigation, and History*
- ~ *Networking Concepts*
- ~ *Storage*
- ~ *Worker Threads and Messaging*

## => Node js :

- ~ *What is Node.js?*
- ~ *Client-Server Architecture*
- ~ *Single-Threaded Model*
- ~ *Multi-Threaded Model*
- ~ *Multi-Threaded vs Event-Driven*
- ~ *What is Node.js?*
- ~ *Node.js Features*
- ~ *Node.js Installation*
- ~ *Node.js First Example*
- ~ *Blocking vs Non-blocking*
- ~ *Global Objects*
- ~ *File System*
- ~ *Callbacks*
- ~ *Events*
- ~ *Node.js Architecture*
- ~ *NPM(Node Package Manager)*
- ~ *Node.js Modules*
- ~ *Node.js Modules Types*
- ~ *Core Modules*
- ~ *Local Modules*
- ~ *3rd Party Modules*
- ~ *JSON File*
- ~ *Variables*
- ~ *Operators*
- ~ *Functions*
- ~ *Objects*
- ~ *File Systems*
- ~ *Events*
- ~ *HTTP Module*
- ~ *Creating a Web Server using Node.js*
- ~ *Node.js NPM Tutorial*
- ~ *What is NPM?*
- ~ *Main Functions of NPM*
- ~ *Need For NPM*
- ~ *NPM Packages*
- ~ *NPM Installation*
- ~ *JSON File*
- ~ *Node.js Express Tutorial*
- ~ *Introduction to Express.js*
- ~ *Features of Express.js*
- ~ *Getting Started with Express.js*
- ~ *Routing Methods*
- ~ *Building RESTful API with Node.js*
- ~ *What is REST API?*
- ~ *Features of REST API*
- ~ *Principles of REST API*
- ~ *Methods of REST API*
- ~ *Building REST API with Node.js*
- ~ *Contact List MERN App*

## => React JS :

- ~ *Introduction to React*
- ~ *Why should you learn React?*
- ~ *Features of React*
- ~ *React applications*
- ~ *React App & JSX*
- ~ *Functional Components*
- ~ *Applying CSS Styles*
- ~ *Click Events*
- ~ *useState Hook*
- ~ *Lists & Keys*
- ~ *Props & Prop Drilling*
- ~ *Controlled Component Inputs*
- ~ *Project Challenge*
- ~ *useEffect Hook*

- ~ JSON Server
- ~ Fetch API Data
- ~ CRUD Operations
- ~ Fetch Data Challenge
- ~ React Router
- ~ Router Hooks
- ~ Links
- ~ Flexbox Components
- ~ Axios API Requests
- ~ Custom React Hooks
- ~ Context API & useContext Hook
- ~ Build & Deploy Your React Apps

=> Javascript Projects :

- ~ Creating shopping cart app with User Interface

=> Bitcoin Blockchain :

- ~ History of currencies
- ~ Fiat currencies
- ~ Disadvantages of fiat currencies
- ~ Global financial system
- ~ How Central Banks work?
- ~ The 2008 Global Financial Crisis
- ~ Aftermath of 2008 recession
- ~ Creation of Bitcoin- A new decentralised digital currency
- ~ Bitcoin message hash implementation in Javascript
- ~ Immutable ledger practical implementation
- ~ Genesis block
- ~ Timestamp server
- ~ Merkel trees
- ~ Bitcoin as a State Transition System
- ~ Unspent Transaction outputs(UTXOs) Javascript implementation
- ~ Bitcoin whitepaper
- ~ What is a block?
- ~ Components of a Bitcoin block
- ~ Bitcoin Blockchain live implementation
- ~ Distributed Blockchain
- ~ Centralized vs Distributed Blockchain
- ~ consensus mechanism
- ~ Why do we need consensus mechanism in Blockchain networks?
- ~ Byzantine generals problem
- ~ Byzantine fault tolerance- A solution to Byzantine generals problem
- ~ BFT javascript implementation
- ~ Bitcoin nodes
- ~ Bitcoin miners
- ~ Blockchain mining operation
- ~ Mempool
- ~ Bitcoin difficulty adjustment
- ~ Bitcoin halving cycle
- ~ Competing chain problem
- ~ Maintaining immutability - Longest Chain rule
- ~ Block validation
- ~ consensus rules
- ~ Double Spend Validation
- ~ Transaction Input and Output Validation
- ~ Coinbase Transaction Reward Validation
- ~ Coinbase Maturity
- ~ Coinbase Transaction Block Height
- ~ Signature Check Counting
- ~ SigChecks
- ~ Mining incentive
- ~ Mining optimized hardware
- ~ CPU processing power
- ~ GPUs for mining
- ~ Application Specific Integrated Circuits(ASIC) miners
- ~ CPU vs GPU vs ASIC miners
- ~ Distributed peer to peer Blockchain live implementation
- ~ Distributed peer to peer Blockchain Javascript implementation
- ~ Token transaction live implementation using distributed peer to peer blockchain
- ~ Coinbase transaction live implementation using distributed peer to peer blockchain
- ~ Token and Coinbase transaction Javascript implementation
- ~ Bitcoin public key and private key
- ~ Public key and private key generation
- ~ Bitcoin addresses
- ~ Bitcoin digital signatures
- ~ Signing a peer to peer message with private key- Javascript implementation
- ~ Verifying peer to peer message using public key and digital signature-implementation
- ~ Signing and verifying currency transaction- implementation
- ~ Complete Bitcoin Blockchain implementation with transaction signatures

=> Probable attacks in Bitcoin blockchain :

- ~ Sybil Attack
- ~ Race Attack
- ~ Finney Attack
- ~ Vector76 Attack
- ~ 51% Attack

=> Bitcoin Project :

- ~ Building a Blockchain using Javascript

## => Ethereum Blockchain :

- ~ Module overview
- ~ Understanding the drawbacks of Bitcoin blockchain
- ~ Lack of Turing-completeness
- ~ Value-blindness
- ~ Lack of state
- ~ Blockchain-blindness
- ~ Origin of Ethereum- The programmable currency
- ~ The Decentralized Applications revolution and modern state of blockchain systems
- ~ Decentralized Applications vs Centralized Applications
- ~ Ethereum Accounts overview
- ~ Contract Accounts(CA)
- ~ Externally Owned Accounts(EOA)
- ~ Fields in Ethereum accounts
- ~ Ethereum Account messages
- ~ Ethereum Account transactions
- ~ Ethereum Addresses
- ~ Units of Ether
- ~ Ether Gas
- ~ Computing total gas cost for Ethereum transactions
- ~ Ethereum gas price Javascript implementation
- ~ Ethereum as a State Transition Function
- ~ Ethereum Architecture
- ~ Ethereum Virtual Machine(EVM)
- ~ EVM nodes vs mining nodes
- ~ EVM Bytecode
- ~ EVM Instruction Set
- ~ EVM Opcode
- ~ EVM Storage
- ~ EVM Memory
- ~ EVM Stack
- ~ Geth setup and EVM practical
- ~ Converting bytecode to opcode
- ~ Application Binary Interface(ABI)
- ~ Understanding end-to-end Ethereum Blockchain transaction in Javascript
- ~ Ethereum Smart Contracts architecture

## => Ethereum 2.0 :

- ~ Why was Ethereum 2.0 proposed?
- ~ Energy usage in Proof of Work
- ~ Gas costs in Ethereum 1.0
- ~ Potential scalability issues
- ~ Moving from Proof of Work to Proof of Stake
- ~ Proof of Stake in Ethereum 2.0
- ~ Validators
- ~ Staking
- ~ Attestation
- ~ Crosslinks
- ~ Finality
- ~ consensus clients
- ~ Execution clients
- ~ Sharding
- ~ Shard chains
- ~ Beacon chain
- ~ Data rollup in Ethereum 2.0
- ~ Forking in Blockchain
- ~ Hard Fork
- ~ Soft Fork
- ~ The DAO attack and Ethereum Hard Fork

## => Solidity :

- ~ What is Solidity?
- ~ Why should you learn Solidity programming?
- ~ Introduction to Smart Contracts
- ~ Solidity Installation
- ~ Remix IDE
- ~ Installing Solidity in npm / Node.js
- ~ Layout of a Solidity Source File
- ~ SPDX License Identifier
- ~ Pragmas
- ~ Comments in Solidity
- ~ Structure of a Smart Contract

## => Solidity Value Types :

- ~ Solidity datatypes
- ~ Booleans
- ~ Integers
- ~ Address Type
- ~ Address Literals
- ~ Contract Types
- ~ Byte Type
- ~ String Types
- ~ Enums in Solidity

## => Solidity Reference Types :

- ~ Data locations- storage, memory and callback
- ~ Solidity Arrays
- ~ Fixed Arrays
- ~ Dynamic Arrays

- ~ Bytes and Strings as Arrays
- ~ Array Slicing
- ~ Structs
- ~ Mapping Types

#### => Solidity Units and Global Variables :

- ~ Ether Units
- ~ Time Units

#### => Solidity Control Structures :

- ~ If statement
- ~ If/else statement
- ~ Nested if/else statements
- ~ Solidity Loops
- ~ For loop
- ~ While loop
- ~ Do-while loop
- ~ Break statement
- ~ Continue statement

#### => ABI Encoding and Decoding Functions :

- ~ ABI encoder
- ~ ABI decoder

#### => Cryptographic Functions :

- ~ Keccak256
- ~ SHA256
- ~ Ripemd160
- ~ Ecrecover

#### => Smart Contracts :

- ~ Creating Smart Contracts
- ~ Constructor
- ~ Scope visibility
- ~ State variable visibility
- ~ Functions
- ~ Function visibility
- ~ Getter functions
- ~ Setter functions
- ~ Function modifiers
- ~ Return variables and returning multiple values
- ~ Immutable state variables
- ~ Payable functions
- ~ Fallback functions
- ~ View functions
- ~ Pure functions
- ~ Function overloading
- ~ Function overriding
- ~ Solidity Events
- ~ Block and Transaction details
- ~ Solidity Inheritance
- ~ Single Inheritance
- ~ Multiple Inheritance
- ~ Heirarchical Inheritance
- ~ Multilevel Inheritance
- ~ Abstract Contracts
- ~ Solidity Interfaces
- ~ Solidity Libraries

#### => Solidity Programming Applications :

- ~ Ether Wallet
- ~ Multi Sig Wallet
- ~ Iterable Mapping
- ~ ERC20
- ~ ERC721
- ~ Uni-directional Payment Channel
- ~ Bi-directional Payment Channel
- ~ NFT Auction
- ~ Crowd Fund
- ~ Time Lock

#### => Common Ethereum Blockchain Hacks and Loopholes :

- ~ Re-Entrancy Attack
- ~ Self Destruct
- ~ Accessing Private Data
- ~ Denial of Service
- ~ Phishing with tx.origin
- ~ Hiding Malicious Code with External Contract
- ~ Honeypot
- ~ Front Running
- ~ Block Timestamp Manipulation
- ~ Signature Replay
- ~ Bypass Contract Size Check

#### => Introduction to Blockchain Development Frameworks :

- ~ Introduction to Smart Contract Development in Production
- ~ Web3 libraries for Javascript
- ~ Smart Contract development tools
- ~ Web3 Providers
- ~ Wallets

## => Truffle Suite :

- ~ Truffle overview
- ~ Truffle Installation
- ~ Creatin a new project in Truffle
- ~ Exploring project directories in Truffle
- ~ Compiling Smart Contracts
- ~ Building Artifacts
- ~ Handling Dependencies
- ~ Reading and writing Smart Contract data
- ~ Smart Contract Transactions in Truffle
- ~ Function calls in Truffle
- ~ Abstractions
- ~ Executing Contract functions
- ~ Making Transactions
- ~ Processing Transaction results
- ~ Catching events
- ~ Add a new contract to the network
- ~ Sending ether to a contract
- ~ Invoking overloaded methods
- ~ Using enumerations
- ~ Preserving Files and Content to Storage Platforms
- ~ Inter Planetary File System(IPFS)
- ~ Filecoin
- ~ Textile Buckets
- ~ Running Migrations
- ~ Initial Migration
- ~ Truffle Deployer
- ~ Network considerations
- ~ Truffle Deployer API
- ~ Integrating Truffle with Metamask
- ~ Using Truffle Dashboard
- ~ Using truffle Debugger
- ~ Truffle Develop and Truffle Console
- ~ Writing and executing external scripts
- ~ Testing Smart Contracts
- ~ Writing Automated Tests in Javascript
- ~ Writing Automated Tests in Solidity
- ~ Truffle Build Process
- ~ Truffle Boxes
- ~ Ethereum Name Service
- ~ Truffle Event System
- ~ Network Configuration and Dapp Deployment
- ~ Ganache- Ethereum Client for Truffle Suite
- ~ Installing Ganache
- ~ Ganache Workspaces
- ~ Ganache Ethereum Workspace
- ~ Understanding Workspace Default Configuration in Ganache
- ~ Managing Ganache configurations and settings
- ~ Configuring Truffle to connect to Ganache
- ~ Managing Truffle projects in Ganache
- ~ Exploring the Contracts page
- ~ Exploring the Transactions page
- ~ Linking and unlinking a Truffle project
- ~ Ganache Workspaces
- ~ Creating Workspaces
- ~ Deleting Workspaces
- ~ Editing Workspaces
- ~ Ethereum Workspace
- ~ Loading Existing Workspaces
- ~ Switching Workspaces

## => Hardhat :

- ~ Introduction To Hardhat - Ethereum development environment for professionals
- ~ Hardhat Installation
- ~ Creating a Hardhat project
- ~ Configuring Ethereum Networks
- ~ Configuring the compiler
- ~ Compiling your contracts
- ~ Artifacts
- ~ Writing deployment scripts
- ~ Deploying the Contracts
- ~ Testing Smart Contracts
- ~ Running tests with Ganache
- ~ Running tests on Visual Studio Code
- ~ Running multiple tests in parallel
- ~ Running tasks
- ~ Hardhat Console
- ~ Creating custom tasks
- ~ Hardhat Runtime Environment(HRE)
- ~ Hardhat Plugins
- ~ Optimizing Plugins
- ~ Verbose Logging for debugging
- ~ Solutions to common runtime problems

## => Web3.js :

- ~ Introduction to Web3.js
- ~ Why should you learn Web3.js?
- ~ Applications of Web3.js

- ~ Installing Web3.js using NPM
- ~ Web3 modules
- ~ Creating a new Web3 instance
- ~ Introduction to Web3 Providers
- ~ Setting up a Web3 Provider
- ~ Batch request
- ~ Extending Web3 modules
- ~ Introduction to Web3.eth
- ~ Checksum addresses overview
- ~ Fetching default blockchain details
- ~ Transaction methods
- ~ Block Node methods
- ~ Subscriber Methods
- ~ Web3.js Smart Contract objects and methods
- ~ User wallet and account methods
- ~ Interacting with Ethereum node accounts using web3.eth.personal
- ~ Working with ABI in web3.js
- ~ Commonly used utilities in web3.js
- ~ Hardhat automated testing with Web3.js and Truffle

=> Ethers.js :

- ~ What is Ethers?
- ~ Ethers.js Features
- ~ Installing Ethers.js using NPM
- ~ Connecting to Ethereum: MetaMask
- ~ Connecting to Ethereum: RPC
- ~ Building blocks of Ethers.js- Signers, Providers and Contracts

=> Ethers.js Providers :

- ~ What are Providers?
- ~ Ethers.js provider API overview
- ~ Provider Account methods
- ~ Blocks Methods
- ~ Ethereum Naming Service (ENS) Methods
- ~ EnsResolver
- ~ Logs Methods
- ~ Network Status Methods
- ~ Transactions Methods
- ~ Event Emitter Methods
- ~ Inspection Methods
- ~ BaseProvider
- ~ JsonRpcProvider
- ~ JsonRpcSigner
- ~ JsonRpcUncheckedSigner
- ~ StaticJsonRpcProvider
- ~ Node-Specific Methods
- ~ API Providers
- ~ EtherscanProvider
- ~ InfuraProvider
- ~ AlchemyProvider
- ~ CloudflareProvider
- ~ PocketProvider
- ~ AnkrProvider
- ~ Other Providers
- ~ FallbackProvider
- ~ IpcProvider
- ~ JsonRpcBatchProvider
- ~ UrlJsonRpcProvider
- ~ Web3Provider
- ~ WebSocketProvider

=> Ethers.js Signers :

- ~ What are Signers?
- ~ Wallet Signer
- ~ JsonRPC Signer
- ~ Signer class and member functions
- ~ Ethers.js Wallet class and member functions
- ~ VoidSigner
- ~ Interacting with Externally Owned Accounts (EOA)

=> Smart Contract Interaction :

- ~ Creating new Smart Contract instance
- ~ Contract Properties
- ~ Contract Methods
- ~ Events
- ~ ContractFactory
- ~ Creating ContractFactory Instances
- ~ ContractFactory Interface Properties
- ~ ContractFactory Methods
- ~ Meta-Class
- ~ Deploying a Contract
- ~ Connecting to a Contract
- ~ Properties
- ~ Methods
- ~ Events
- ~ Meta-Class Methods
- ~ Meta-Class Filters
- ~ Hardhat automated testing with Ether.js and Waffle

=> Ethereum Blockchain Projects :

- ~ Building cryptocurrency with ICO
- ~ Building decentralized ecommerce website
- ~ Building decentralized voting application
- ~ Decentralized music sharing app
- ~ Token contract swap application
- ~ Full stack email dapp

#### => Oracles :

- ~ What is a Blockchain Oracle?
- ~ Solving the Oracle problem
- ~ Decentralized Oracles
- ~ Types of Blockchain Oracles
- ~ Applications of Blockchain oracles

#### => Chainlink overview :

- ~ Introduction to Chainlink
- ~ Understanding the Chainlink Ecosystem
- ~ Chainlink Features
- ~ Chainlink Applications as Decentralized Oracles
- ~ Chainlink Architecture
- ~ ERC677 Standard
- ~ The LINK token
- ~ Decentralized Data Model
- ~ Chainlink Off-chain Reporting
- ~ Chainlink Whitepaper

#### => Data Feeds :

- ~ Introduction to Data Feeds
- ~ Using Data Feeds
- ~ Fetchin Historical Cryptocurrency Price Data
- ~ Chainlink Feed Registry
- ~ Using ENS with Data Feeds
- ~ Contract Addresses
- ~ Ethereum Data Feeds
- ~ Binance Smart Chain Data Feeds
- ~ Polygon (Matic) Data Feeds
- ~ Gnosis Chain (xDai) Data Feeds
- ~ HECO Chain Data Feeds
- ~ Avalanche Data Feeds
- ~ Fantom Data Feeds
- ~ Arbitrum Data Feeds

#### => Chainlink VRF :

- ~ Introduction to Chainlink VRF(Verifiable Random Function)
- ~ Applications of randomness in Blockchain
- ~ Generating randomness
- ~ Some security considerations in Chainlink VRF
- ~ Smart Contract Integration

#### => Custom Data Feeds :

- ~ Using any API
- ~ Make a GET Request
- ~ Multi-Variable Responses
- ~ Large Responses
- ~ Make an Existing Job Request
- ~ Find Existing Jobs
- ~ Contract Addresses

#### => Chainlink Keepers :

- ~ Automating Smart Contracts
- ~ Introduction to Chainlink Keepers
- ~ Keepers Architecture
- ~ Keepers-compatible Contracts
- ~ Register an Upkeep
- ~ Manage your Upkeeps
- ~ Utility Contracts
- ~ EthBalanceMonitor
- ~ Supported Networks
- ~ Chainlink Keepers Economics

#### => Oracle Projects :

- ~ Live cryptocurrency trading using chainlink
- ~ Insurance Dapp using chainlink

#### => The Graph :

- ~ The Graph Protocol
- ~ The Graph architecture
- ~ Edge and Node
- ~ Everest Registry
- ~ Graph Protocol
- ~ The Graph vs Etherscan
- ~ Graph-cli Installation
- ~ Creating new subgraphs
- ~ Writing subgraphs
- ~ Publishing a Subgraph to the Decentralized Network

#### => GraphQL API :

- ~ Queries
- ~ Sorting
- ~ Pagination
- ~ Filtering

- ~ Time-travel queries
- ~ Fulltext Search Queries
- ~ Validation
- ~ Schema
- ~ Entities
- ~ Signalling
- ~ Curation
- ~ Delegators
- ~ Consumers
- ~ Deploying subgraphs
- ~ Subgraph logging
- ~ Graph protocol testnet using docker compose
- ~ Ethereum node monitoring using The Graph, Prometheus and Grafana

#### => The Graph Networking :

- ~ Introduction to indexers
- ~ Revenue streams
- ~ Distribution
- ~ Allocation life cycles
- ~ Querying and indexing subgraphs
- ~ IPFS Hash convertor

#### => AssemblyScript API for The Graph :

- ~ Installing AssemblyScript API
- ~ API Reference
- ~ Versions
- ~ Built-in Types
- ~ Store API
- ~ Ethereum API
- ~ Logging API
- ~ IPFS API
- ~ Crypto API
- ~ JSON API
- ~ Type Conversions Reference
- ~ Data Source Metadata
- ~ Entity and DataSourceContext

#### => The Graph Unit Testing :

- ~ Installing dependencies
- ~ WSL (Windows Subsystem for Linux)
- ~ Usage
- ~ CLI options
- ~ Docker
- ~ System Configuration
- ~ Demo subgraph
- ~ Asserts
- ~ Writing a Unit Test
- ~ Common test scenarios
- ~ Hydrating the store with a certain state
- ~ Calling a mapping function with an event
- ~ Calling all of the mappings with event fixtures
- ~ Mocking contract calls
- ~ Asserting the state of the store
- ~ Interacting with Event metadata
- ~ Asserting variable equality
- ~ Asserting that an Entity is not in the store
- ~ Printing the whole store (for debug purposes)
- ~ Expected failure
- ~ Logging
- ~ Testing derived fields
- ~ Testing dynamic data sources
- ~ Test Coverage

#### => Project :

- ~ Building a Full-stack Blockchain Application using Ethereum, Polygon, Next.js and GraphQL

#### => Decentralized Autonomous Organisations(DAO) :

- ~ What are DAOs?
- ~ Why do we need DAOs?
- ~ DAO membership
- ~ Token-based membership
- ~ Share-based membership
- ~ How do DAOs work?
- ~ Properties of DAOs
- ~ Ethereum and DAOs
- ~ Understanding Governance Mechanisms
- ~ DAOs and the principal-agent problem
- ~ Building Decentralized Autonomous Organisations
- ~ Defining the DAO purpose
- ~ Building the DAO voting mechanism
- ~ Creating the governance token
- ~ DAO fund management
- ~ Initial Coin Offering (ICO)
- ~ Creating a DAO on Aragon
- ~ Creating a DAO using Snapshot
- ~ Building a DAO using DAOstack Alchemy

#### => Creating a Custom DAO Project :

- ~ Understanding custom DAOs
- ~ Finding the purpose for our Custom DAO



- ~ Designing the voting architecture
- ~ Implementing the voting architecture in Solidity
- ~ Designing the components of the governance token(DAO cryptocurrency)
- ~ Creating the governance token in Solidity
- ~ Fund Management for our custom DAO
- ~ Designing the Multi-signature wallet for Fund Management
- ~ Creating the Multi-signature wallet in Solidity
- ~ Testing DAO Smart Contracts
- ~ Deploying the DAO to testnet

#### => Decentralized Finance(DeFi) :

- ~ The Traditional Financial Institutions
- ~ Centralization & Transparency
- ~ The Banks
- ~ General Public Accessibility
- ~ Decentralized Finance
- ~ The DeFi Ecosystem
- ~ How does DeFi work?
- ~ DeFi Categories
- ~ Decentralized Stablecoins
- ~ Lending and Borrowing
- ~ Decentralized Exchanges
- ~ Derivatives
- ~ Fund Management
- ~ Lottery
- ~ Decentralized payments systems
- ~ Insurance
- ~ Yield Farming
- ~ Liquidity Mining
- ~ Airdrops
- ~ Decentralized Prediction Markets

#### => Famous DeFi Protocols :

- ~ Aave
- ~ yEarn
- ~ Compound
- ~ Uniswap
- ~ Sushiswap
- ~ Maker
- ~ Numerai
- ~ Curve Finance
- ~ Alpha Finance

#### => DeFi projects :

- ~ Understanding DeFi Project Architecture
- ~ Components of Full-Stack DeFi applications
- ~ Designing DeFi project workflows
- ~ Building a Decentralized lottery system
- ~ Building a Decentralized borrowing and lending platform
- ~ Building a Decentralized stablecoin

#### => Non Fungible Tokens (NFT) :

- ~ What is a Non Fungible Token(NFT)?
- ~ How does a NFT work?
- ~ Fungible Tokens vs Non Fungible Tokens
- ~ Exploring uses of NFTs
- ~ NFT as an internet of assets
- ~ NFT as a store of value
- ~ The Metaverse and NFT's role in it

#### => NFT Platforms :

- ~ What are NFT Platforms/Marketplaces?
- ~ CryptoKitties
- ~ Opensea
- ~ Rarible
- ~ Decentraland
- ~ Binance NFT
- ~ Enjin Marketplace
- ~ Axie Marketplace
- ~ Foundation
- ~ Nifty Gateway
- ~ Mintable
- ~ Theta Drop

#### => NFT Transaction Fees :

- ~ Gas Fees in NFT
- ~ What are one-time Gas Fees NFT?
- ~ Recurring Gas Fees
- ~ Actions in Gas Fees
- ~ Check Ethereum Gas Fee
- ~ Create and Sell NFTs without Gas Fees
- ~ NFT Marketplace Fees

#### => NFT Programming :

- ~ Getting data for generating NFTs
- ~ Assigning trait rarity for digital assets
- ~ Classifying traits
- ~ Defining image traits
- ~ Validating uniqueness
- ~ Trait Counting
- ~ Generate the Images

- ~ Understanding NFT metadata
- ~ Uploading NFT images to IPFS
- ~ Generate NFT metadata
- ~ Upload the metadata to IPFS
- ~ Environment Setup for Smart Contract deployment
- ~ Creating Alchemy account
- ~ Writing NFT smart contract
- ~ Integrating Metamask, Alchemy and your Project

=> NFT project :

- ~ Building a complete NFT Marketplace with User Interface

=> Polygon Blockchain(MATIC) :

- ~ Introduction to Polygon Blockchain
- ~ Why should you use Polygon network?
- ~ Layer 1 vs Layer 2 Blockchains
- ~ Features of Polygon Blockchain
- ~ Polygon Architecture
- ~ Zero-Knowledge cryptography
- ~ Zero-Knowledge rollups

=> Polygon Network :

- ~ Introduction to Polygon Mainnet and Testnet
- ~ Mapped Tokens
- ~ Matic Gas Token
- ~ Genesis Contracts
- ~ Minimum Technical Requirements
- ~ Snapshot Instructions for Heimdall and Bor
- ~ Full Node Binaries
- ~ Full Node Deployment
- ~ Polygon Wallets
- ~ Arkane
- ~ Formatic
- ~ Metamask

=> Polygon-Ethereum Bridge :

- ~ Introduction to Polygon POS bridge
- ~ Matic.js
- ~ Installing matic.js using NPM
- ~ Polygon Web3.js Setup
- ~ Polygon Ethers.js Setup
- ~ Supported libraries
- ~ Web3js setup
- ~ Ethers setup
- ~ Matic.js POS
- ~ Matic.js POSClient
- ~ Matic.js ERC20
- ~ Matic.js ERC721
- ~ Matic.js ERC1155
- ~ isCheckpointed
- ~ isDeposited
- ~ deposit ether
- ~ FxPortal
- ~ Set ProofApi

=> Advanced Concepts :

- ~ ABIManager
- ~ Plugins
- ~ ExitUtil
- ~ PoS Bridge
- ~ Using Polygon Edge
- ~ Instantiating Polygon Edge
- ~ Deposit and Checkpoint Event Tracking
- ~ Deployment Details
- ~ Mapping Assets using POS
- ~ Tools
- ~ Wallet Widget
- ~ Submit Mapping Request
- ~ Polygon Mintable Assets
- ~ IPFS - Filecoin
- ~ Using IPFS
- ~ Using Filecoin
- ~ Mint with NFT.storage on Polygon
- ~ Polygon with Oracles
- ~ Chainlink integration

=> Polygon Projects :

- ~ Retail supply chain Application using Polygon Network
- ~ Building a Social media Dapp on Polygon

=> Polkadot :

- ~ Polkadot Overview
- ~ Polkadot Whitepaper
- ~ Polkadot Architecture
- ~ Parachains
- ~ Parathreads
- ~ Substrate Installation

=> Substrate Fundamentals :

- ~ Runtime environment and setup
- ~ Extrinsic

- ~ *Account Abstractions*
- ~ *Transaction Pool*
- ~ *Session Keys*
- ~ *Transaction Weight*
- ~ *Execution*
- ~ *Off-Chain Features*

=> Runtime Development :

- ~ *Frames*
- ~ *Macros*
- ~ *Metadata*
- ~ *Storage*
- ~ *Origins*
- ~ *Events and Errors*
- ~ *Weights and Fees*
- ~ *Benchmarking*
- ~ *Debugging*
- ~ *Testing*
- ~ *Randomness*
- ~ *Chain Specification*
- ~ *Upgrades*
- ~ *Pallet Coupling*
- ~ *Custom RPCs*
- ~ *Smart Contract Toolkits*

=> Development Integration :

- ~ *Polkadot-JS*
- ~ *Client Libraries*
- ~ *Substrate Connect*

=> Development Tools :

- ~ *SR tool*
- ~ *Subxt*
- ~ *Tx Wrapper*
- ~ *Sub Flood*
- ~ *Substrate Archive*
- ~ *Sidecar*
- ~ *Polkadot Launch*

=> Advanced topics in Polkadot :

- ~ *Account Info*
- ~ *SCALE Codec for Substrate*
- ~ *Consensus*
- ~ *Block Import*
- ~ *Executor*
- ~ *Cryptography*
- ~ *Storage*
- ~ *SS58 Address Format*
- ~ *Hash Collections*

# SQL Job Preparation

---

Topic Name : DATA ANALYTICS

Sub-topic Name : SQL INTERVIEW QUESTIONS

Course link : <https://ineuron.ai/course/SQL-Job-Preparation>

## Course Description :-

The bulk of businesses rely on large, relational databases, and they're always looking for SQL specialists. The purpose of this course is to improve your SQL abilities through interview questions and to prepare you for the questions that will most likely be asked during the interview.

## Course Features :-

- => Roadmap
- => Resume preparation
- => Quizzes
- => Downloadable resources
- => Interview Questions
- => Completion certificate

## What you will learn :-

- => SQL practice questions
- => Tackle different questions in interview
- => Project presentation skills
- => Database schema design

## Requirements :-

- => Prior knowledge of SQL
- => A System with internet connection
- => Dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Beginner :
  - ~ What is SQL? Preview
  - ~ How to create a table in SQL?
  - ~ How to delete a table in SQL?
  - ~ How to change a table name in SQL?
  - ~ How to delete a row in SQL?
  - ~ How to create a database in SQL?
  - ~ What is Normalization in SQL?
  - ~ What is join in SQL?
  - ~ What is SQL Server ?
  - ~ How to insert date in SQL?
  - ~ What is Primary Key in SQL?
  - ~ How do I view tables in SQL ?
  - ~ What is PL/SQL
  - ~ What is MYSQL?
  - ~ How can see all tables in SQL ?
  - ~ What is ETL in SQL?
  - ~ How to install SQL?
  - ~ What is the Update Command in SQL ?
  - ~ How to rename column name in SQL server ?
  - ~ What are the types of SQL Queries?
  - ~ Write a Query to display the number of employees working in each region ?
  - ~ What are nested Triggers ?
  - ~ Write a Query the employee names having a salary greather than 20000 or equal to or less than 10000
  - ~ Given a table Employee having columns empName and empid, what will be the result of the SQL query below? Select empName from Employee order by 2 asc;
  - ~ What is OLTP?
  - ~ What is Data Integrity?
  - ~ What is OLAP?
  - ~ Find the constraints information from the table
  - ~ Can you get the list of employees with same salary ?
  - ~ What is an alternative for TOP clause in SQL
  - ~ Will following statement give error or 0 as output ? SELECT AVG (NULL)
  - ~ What is cartesian product of the Table ?
  - ~ What is a schema in SQL?

- ~ What is the where clause in SQL ?
- ~ How to delete a column in SQL?
- ~ What is a unique key in SQL?
- ~ How to Implement multiple conditions using WHERE Clause ?

=> Intermediate :

- ~ What is SQL Injection ? Preview
- ~ What is a Trigger in SQL ?
- ~ How to insert multiple rows in SQL ?
- ~ How to find the nth Highest salary in SQL ?
- ~ How to Copy table in SQL ?
- ~ How to add a new column in SQL ?
- ~ How to use LIKE in SQL ?
- ~ If we drop a table, does it also drops related objects like constraints, indexes, columns, default, views and sorted procedures ?
- ~ Can we disable trigger? If yes, How ?
- ~ What is a Live Lock ?
- ~ How to fetch alternate records from a table ?
- ~ Define COMMIT and give an example ?
- ~ Can you join table by itself ?
- ~ Example Equi join with example ?
- ~ How do we avoid getting duplicate entries in a query ?
- ~ How can you create an empty table from an existing table ?
- ~ Write a query to display odd records from student table ?
- ~ Explain Non Equi join with example ?
- ~ How can you delete duplicate records in a table with no primary key ?
- ~ Difference between NVL and NVL2 functions ?
- ~ What is the difference between clustered and non-clustered indexes ?
- ~ What does this query says ? GRANT privilege\_name ON object\_name TO {user\_name|PUBLIC|role\_name } [WITH GRANT OPTION];
- ~ Is semicolon used after sql? If yes/No, please justify the reason
- ~ Difference between JOIN and UNION
- ~ Difference between order by and group by.
- ~ What is a candidate key?
- ~ When can you compare the dates in SQL
- ~ What is SQL injection? When does SQL injection occurs?
- ~ What is ENUM?
- ~ What is the difference between ATAN and ATAN2 function
- ~ What is the difference between CEIL, FLOOR and ROUND function?
- ~ What is a RAND() function?
- ~ What is the difference between LOCALTIMESTAMP and CURRENT\_TIMESTAMP
- ~ Name three functions that specify current date and time.
- ~ Which function returns the difference between two periods. And the result will be in which format?
- ~ How to fetch common records from two tables?
- ~ How to fetch alternate records from a table?
- ~ How to select unique records from a table?
- ~ What is the command used to fetch first 5 characters of the string?
- ~ How to copy table in SQL?
- ~ How to use LIKE in SQL?
- ~ If we drop a table, does it also drop related objects like constraints, indexes, columns, default, views and sorted procedures?

=> Advanced :

- ~ Can we disable a trigger? If yes, How?
- ~ What is a Live Lock?
- ~ Can you join table by itself?
- ~ Explain Equi join with example
- ~ Explain Non Equi Join with example?
- ~ Difference between NVL and NVL2 functions?
- ~ What does this query says? GRANT privilege\_name ON object\_name TO {user\_name|PUBLIC|role\_name} [WITH GRANT OPTION];
- ~ Where MyISAM table is stored?
- ~ What does myisamchk do?
- ~ How to store Videos inside SQL Server table ?
- ~ Write an SQL query to show the second highest salary from a table.
- ~ How would you select all the users, whose phone number is NULL?
- ~ Write an SQL query to fetch three max salaries from a table.
- ~ Write a SQL query to create a new table with data and structure copied from another table.
- ~ Difference between having vs where.
- ~ When to use NoSQL and SQL?
- ~ What is SYSTEM privilege?
- ~ What are Object Privileges?
- ~ What does a BCP command do?
- ~ Does the data stored in the stored procedure increase access time or execution time? Explain
- ~ Can a View be active if the Base table is dropped?
- ~ What is the difference between the RANK() and DENSE\_RANK() functions? Provide an example.
- ~ What us CTE?
- ~ Does view contain data?
- ~ Define a temp table.
- ~ What is referential integrity?
- ~ What do you mean by query optimization?
- ~ What are nested triggers?
- ~ What is SQL injections attacks?
- ~ What is schema in SQL server?
- ~ How to change SQL server password?
- ~ What is CTE in SQL Server?
- ~ Let say, you have employee details in employee table and project details in project table. Find the list of employeeed that are assigned to some projects.
- ~ Let say, you have employee details in employee table and project details in project table. Find the total list of employees along with their respective project names (Employees with no project details should have their project as null).
- ~ Let say you have wifi\_id, wifi\_speed, wifi\_latency, date. Find the average wifi speed for each wifi.
- ~ Let say you have wifi\_id, wifi\_speed, wifi\_latency, date. Find the average wifi speed for each wifi in the last 2 days.
- ~ Write a query to fetch top N records?

- ~ Write a query to calculate the even and odd records from a table
- ~ Write a query to fetch 50% records from the EmployeeInfo table.
- ~ Write a query to add email validation to your database

# Angular

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : ANGULAR JS

Course link : <https://ineuron.ai/course/Angular>

## Course Description :-

"Angular 13 simply is the latest version of Angular 2. It will be started from scratch so need to be prepared for anything. Angular is one of the most popular, performance-efficient, and powerful frontend frameworks you can learn as of today. You can build great web apps using this framework. And as always you will have the support of the Google Community."

## Course Features :-

- => Build single-page applications with Angular and Typescript
- => Master fundamental concepts behind structuring Angular applications
- => Reusable Components
- => Covers all the major concepts

## What you will learn :-

- => Angular use
- => Use of TypeScript in between
- => Building Components
- => Databinding
- => Routing
- => Pipes and their uses
- => Resful services(API)
- => Modules

## Requirements :-

- => Prior knowledge of Javascript
- => Prior Knowledge of Typescript is helpful

## Instructors :-

- => Syed Ashraf :  
~ Full Stack Engineer at TensorGo Technologies

## Curriculum details :-

- => Introduction :
  - ~ Introduction Preview
  - ~ Installation and Setup
  - ~ Files and Folder Structure
  - ~ Interpolation
- => Basics :
  - ~ CLI Commands Preview
  - ~ Components
  - ~ Styles and Templates
  - ~ Modules
  - ~ Events
  - ~ Counter
  - ~ Property Binding
- => Control Flow :
  - ~ Ng-ifElse
  - ~ Ng-SwitchCase
  - ~ Ng-for
  - ~ Style Binding
- => Forms & Reusing Components :
  - ~ Basic Form
  - ~ Passing Data from Parent to Child
  - ~ Reusable Components
  - ~ Two Way Binding
  - ~ Template Reference Variable
- => Pipes & Forms :
  - ~ Pipes
  - ~ Reactive Forms
  - ~ Directives
- => Routes :
  - ~ Routing

~ Child Routes & 404 Page

=> Services & API :

~ Services

~ API Handling(Part-1)

~ API Handling(Part-2)

=> Project :

~ Calculator



# Azure Machine Learning DP 100

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Azure-Machine-Learning-DP-100>

## Course Description :-

This course is designed for data scientists who want to develop and manage machine learning solutions on the cloud and who already have some familiarity with Python and machine learning frameworks like Scikit-Learn, PyTorch, and Tensorflow. In this course, students will learn how to build comprehensive Microsoft Azure solutions.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Create the AzureML Workspace
- => Create a Dataset
- => Explore the AzureML Dataset
- => Understanding the AzureML Compute Resources
- => Deploy a real-time endpoint using Designer
- => Consume Model
- => Access Workspace, Datastore and Datasets using SDK
- => Pandas Dataframe and AzureML Dataset conversions
- => Upload local data to storage account via datastore
- => Simple Python Script in Designer
- => Execute Python Script using Zip Bundle

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Set up Azure Machine Learning Workspace :
  - ~ Understand the AzureMLService Architecture
  - ~ Create the AzureML Workspace
  - ~ View and Manage Workspace Settings
  - ~ Overview of New AzureML Studio
  - ~ What is AzureML Datastore and Dataset
  - ~ Create and Register a Datastore
  - ~ Create a Dataset
  - ~ Explore the AzureML Dataset
  - ~ Understanding the AzureML Compute Resources
  - ~ Create a Compute Cluster and Compute Instance
- => Model Training and Run Experiment :
  - ~ What is an AzureML Pipeline
  - ~ Create a pipeline using AzureML Designer
  - ~ Submit the Designer Pipeline run
- => Deploy and Consume the Models :
  - ~ Create an Inference Pipeline
  - ~ Deploy a real-time endpoint using Designer
  - ~ Consume Model
  - ~ Create a batch inference pipeline using Designer

- ~ *Run a Batch Inference Pipeline from Designer*
- ~ *Result*

=> **Data Processing using AzureML Designer :**

- ~ *Get Data to the workspace*
- ~ *Import Data to the workspace from external sources*
- ~ *Edit Metadata - Column Names*
- ~ *Understanding the Run*
- ~ *Edit Metadata - Data Type*
- ~ *Export Data to the Blob Storage*
- ~ *Add Columns to the Dataset*
- ~ *Add Rows to the Dataset*
- ~ *Normalization of Data Part 1*
- ~ *Normalization of Data Part 2*
- ~ *Clean Missing Data*
- ~ *Partition and Sample Data Part 1*
- ~ *Partition and Sample Data Part 2*

=> **Azure Machine Learning with Azure :**

- ~ *Introduction to AzureML SDK*

=> **Set Up Azure Machine Learning Workspace using sdk :**

- ~ *Create AzureML Workspace using SDK part 1*
- ~ *Verify the Workspace and Write the Workspace Config File*
- ~ *Create and Register a Datastore using AzureML SDK*
- ~ *Create and Register a Dataset using SDK part 1*
- ~ *Create and Register a Dataset using SDK part 2*
- ~ *Access Workspace, Datastore and Datasets using SDK*
- ~ *Pandas Dataframe and AzureML Dataset conversions*
- ~ *Upload local data to storage account via datastore*

=> **Run Experiments and Train Models :**

- ~ *Set up*
- ~ *Overview of Architecture*
- ~ *Create Sample Experiment part 1*
- ~ *Create Sample Experiment part 2*
- ~ *Run Sample Experiment*
- ~ *Azureml Environment part 1*
- ~ *Azureml Environment part 2*
- ~ *Azureml Environment part 3*
- ~ *Azureml Environment part 4*
- ~ *Azureml Environment part 5*
- ~ *Train and Run a Model Script in AzureML Part 1*
- ~ *Train and Run a Model Script in AzureML Part 2*
- ~ *Train and Run a Model Script in AzureML Part 3*
- ~ *Provisioning Compute Cluster Using SDK Part 1*
- ~ *Provisioning Compute Cluster Using SDK Part 2*
- ~ *Automate Model Training using AzureML SDK*
- ~ *Define Pipeline Steps*
- ~ *Define Training Steps*
- ~ *Built the pipeline*
- ~ *Command Line Arguments*
- ~ *Data preparation script*
- ~ *Training script*
- ~ *Run the pipeline part 1*
- ~ *Run the pipeline part 2*
- ~ *Run the pipeline part 3*

=> **Using Python Scripts in AzureML :**

- ~ *Simple Python Script in Designer*
- ~ *Execute Python Script using Zip Bundle*
- ~ *Execute Python Script using Zip Bundle Demo*

# Mathematics for Machine Learning

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Mathematics-for-Machine-Learning>

## Course Description :-

This course suits for students who want to know the application of mathematics in Machine Learning and also for the working professionals who want to know the mathematical framework behind the machine learning algorithms.

## Course Features :-

- => Assignment
- => Quiz
- => Downloadable Resources
- => Completion Certificate

## What you will learn :-

- => Vectors Operations
- => Matrix Operations
- => Eigen Decomposition
- => Applications of Eigen Decomposition
- => Differentiation
- => Partial Differentiation
- => Regression Project
- => Integration
- => Practical Implementations

## Requirements :-

- => Prior knowledge in Maths
- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Basics of Python

## Instructors :-

=> Bharath J P V :

~ Enthusiast Data Scientist with a strong background in Mathematics and Statistics. Completed My Master in Statistics. Have experience teaching Mathematics and Statistics for more than a year. I thought for more than 1000 students and helped them make their careers in their respective fields. I believe in "we rise by lifting others". Following this principle, I hope to make your life easier.

## Curriculum details :-

=> Introduction :

- ~ Course introduction Preview
- ~ Who is this course for?
- ~ Course overview
- ~ Course outcome

=> Linear Algebra :

- ~ Introduction to Linear Algebra Preview
- ~ Vectors, Matrices & Tensors Preview

=> Vector Operations :

- ~ Transposition and Norm of a Vector
- ~ Dot Product
- ~ Dot Product with Itself
- ~ Orthogonal Vectors
- ~ Projection of Vectors
- ~ Line, Plane and Hyperplane

=> Matrix :

- ~ Transposition of Matrix
- ~ Arithmetic Operation
- ~ Hadamard Operations and Reduction of Matrix
- ~ Hands-on Code demo with Python
- ~ Solving system of Linear Equations
- ~ Types of Solutions
- ~ Plotting Equation
- ~ Hands-on Plotting equations

- ~ *Matrix Norms and Properties*
- ~ *Linear Transformation*
- ~ *Matrix Multiplication*
- ~ *Matrix Inversion*
- ~ *Identity Matrix*
- ~ *Diagonal Matrix*
- ~ *Symmetric Matrix*
- ~ *Determinant of a Matrix*

=> Eigen Vectors and Eigen Values :

- ~ *Eigen Vectors and Eigen Values*
- ~ *Properties of Eigen Values*
- ~ *Eigen Decomposition*

=> Matrix Operations in Machine Learning :

- ~ *Affine Transformations*
- ~ *Singular Vector Decomposition*
- ~ *Image Compression Preview*
- ~ *Moore-Penrose Pseudoinverse*
- ~ *Application of Pseudoinverse*
- ~ *Principle Component Analysis*

=> Limits :

- ~ *Introduction, tangent and Slope*
- ~ *Infinitesimals and Area under Curve*
- ~ *Limits*

=> Differential Calculus :

- ~ *Rate of change and Slope as a Variable*
- ~ *Differential Calculus*
- ~ *Differentiation using Delta Method*
- ~ *Standard Differentiation rules*
- ~ *Sum, Product and Quotient rule*
- ~ *Chain rule*
- ~ *Higher order Derivatives*
- ~ *Application of Derivatives*
- ~ *AutoDiff using PyTorch and Tensorflow*
- ~ *Partial Derivatives*
- ~ *Partial Derivatives using Pytorch and TensorFlow*
- ~ *Application of Partial Derivatives*
- ~ *Chain rule for Partial derivatives Preview*
- ~ *Regression Project Theory*
- ~ *Hands-on Regression Project*
- ~ *Gradient of Point Regression*
- ~ *Gradient of Group Regression*

=> Integral Calculus :

- ~ *Introduction to Integral Calculus Preview*
- ~ *Standard Integrals and Integration Rules*
- ~ *Indefinite and Definite Integrals*
- ~ *Area under Curve(AUC) Using scikit-learn*
- ~ *Receiver Operating Characteristic(ROC) curve*
- ~ *Hands-on ROC AUC*

# Amazon Lex Chatbot

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Amazon-Lex-Chatbot>

## Course Description :-

Amazon Lex is an AWS service for building conversational interfaces for applications using voice and text. With Amazon Lex, the same conversational engine that powers Amazon Alexa is now available to any developer, enabling you to build sophisticated, natural language chatbots into your new and existing applications. Amazon Lex provides the deep functionality and flexibility of natural language understanding (NLU) and automatic speech recognition (ASR) so you can build highly engaging user experiences with lifelike, conversational interactions, and create new categories of products.

## Course Features :-

=> Lifetime Dashboard

=> Free Course

## What you will learn :-

=> Basics of a LexBot

=> Customer support Bot

=> Fulfillment with Amazon lambda, python and using Ngrok

=> Integrating Bot with Facebook Messenger

=> conclusion of Lex Bot

## Requirements :-

=> Python Programming Understanding

=> NLP Concepts

## Instructors :-

=> Mohit Sharma :

~

## Curriculum details :-

=> Amazon Lex |Introduction to Chat bots :

~ *Introduction Preview*

=> Amazon Lex |Basics of a LexBot Part 1

=> Amazon Lex |Basics of a Lex Bot Part 2

=> Amazon Lex|Customer support Bot

=> Amazon Lex | Fulfillment with Amazon lambda, python and using Ngrok part - 1

=> Amazon Lex|Fulfillment with Amazon lambda, python and using Ngrok part 2

=> Amazon Lex|Integrating Bot with Facebook Messenger

=> Amazon Lex |conclusion of Lex Bot

# NLP Foundations

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP

Course link : <https://ineuron.ai/course/NLP-Foundations>

## Course Description :-

NLP Community Class

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to NLP
- => Roadmap of NLP
- => NLTK
- => Stanford NLP
- => Spacy
- => Gensim
- => Tokenization
- => Stemming And Lemmatization
- => Bag of Words in NLP
- => TFIDF
- => Word Embedding
- => Word2vec
- => Implementation of spam classifier using python

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Introduction to NLP :
  - ~ Introduction to NLP
  - ~ Roadmap of NLP
- => Famous Libraries in NLP :
  - ~ NLTK
  - ~ Stanford NLP
  - ~ Spacy
  - ~ Gensim
- => Tokenization & Stemming :
  - ~ Tokenization
  - ~ Stemming And Lemmatization
- => Word Embeddings :
  - ~ Bag of Words in NLP
  - ~ TFIDF
  - ~ Word Embedding
  - ~ Word2vec
  - ~ Implementation of spam classifier using python
- => Networks :

- ~ *RNN*
- ~ *Bidirectional RNN*
- ~ *LSTM*
- ~ *Bi- LSTM*
- ~ *GRU*
- ~ *Fake News Classifier Using LSTM RNN*

# Fastify Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Fastify-Crash-Course>

## Course Description :-

This course will help you to grab the fundamentals of Fastify.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Getting fastify up and running
- => Model the course using mongoose
- => Handling controllers in fastify
- => Handling routes in fastify

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Fastify :

- ~ Getting fastify up and running
- ~ Model the course using mongoose
- ~ Handling controllers in fastify
- ~ Handling routes in fastify

=> NaN :

- ~ NaN
- ~ NaN
- ~ NaN



# HTML and CSS for Web Development

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : HTML

Course link : <https://ineuron.ai/course/HTML-and-CSS-for-Web-Development>

## Course Description :-

This course will help you to grab the fundamentals of HTML and CSS for web development.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => HTML and DOM
- => We do not write without emmet
- => Starting with CSS
- => Mobile responsive webpages
- => CSS animation and libraries
- => Flexbox in CSS

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started :

- ~ Introduction to web dev
- ~ Get the development tools
- ~ your first hello world

=> We do not write without emmet :

- ~ Getting started with emmet
- ~ Emmet and speedy html
- ~ Parent child and grouping
- ~ emmet in css

=> HTML and DOM :

- ~ An old style blog
- ~ Why DOM is important
- ~ Inline vs Block and bring in images
- ~ Lists and interlinking pages
- ~ Getting a video on service page
- ~ 3 Plans in a table
- ~ GET and POST forms
- ~ Types of input forms

=> Starting with CSS :

- ~ Secret to learn CSS
- ~ Explore and bring in fonts
- ~ Bring in colors and styles
- ~ Transition and box shadow DOCS
- ~ Margin and padding
- ~ Button gets all and assignment

=> Working on coming soon template :

- ~ Introduction to CSS variables and new project
- ~ Browser defaults and variables
- ~ Getting more control over elements

=> Mobile responsive webpages :

- ~ *What are media queries*
- ~ *Media query in action*
- ~ *App landing page - setup*
- ~ *Navigation bar for website*
- ~ *Bring content in columns*
- ~ *Cover image in css*
- ~ *Start with flexbox*
- ~ *Buttons and columns in flexbox*
- ~ *Absolute position in CSS*
- ~ *Media query for 2 screens*

=> Register a new account :

- ~ *Handle conflict in CSS*
- ~ *Strategy and placing html*
- ~ *Classes and ID for testers*
- ~ *Where to use z index*
- ~ *Bootstrap style of CSS*
- ~ *Fixing CSS on form*
- ~ *Doing your assignment*

=> CSS animation and libraries :

- ~ *Animation and keyframes*
- ~ *Third party animation library*
- ~ *Razorpay style clipping*
- ~ *Not a payment gateway integration*

=> Flexbox in CSS :

- ~ *What is flexbox*
- ~ *Get to know the power of flexbox*
- ~ *Flexbox series-Axis and Flex direction*
- ~ *Flexbox series -justify content*
- ~ *Align items in flexbox*
- ~ *Flexbox series ordering the elements*
- ~ *Flex grow in flexbox*

# Class 7th Biology

---

Topic Name : K12

Sub-topic Name : CLASS7

Course link : <https://ineuron.ai/course/Class-7th-Biology>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of science and its importance in our daily life. It will make the foundation strong for the higher classes. The Biology section focuses on concepts like food, bodily movements, surroundings of living organisms, garbage, etc.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Nutrition in Plants
- => Nutrition in Animals
- => Weather, Climate and Adaptations of Animals to Climates

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Dr Nishtha Jain :

*~ I am a doctor by profession but a teacher by passion. I have been into the teaching profession for the last 3 years. I have been and am still a mentor for various courses which include technical as well as non-technical ones. These include MS-Excel, Tableau, Computer basics, Biology, English, etc. I love to learn, explore and share my knowledge to whatever extent possible. Being an ardent educator, I have always helped all my students and will continue to do the same.*

## Curriculum details :-

=> Nutrition in Plants :

- ~ Lecture 1 : Introduction & Photosynthesis Preview
- ~ Lecture 2 : Other modes of Nutrition Preview
- ~ Lecture 3 : Cells and nutrient replenishment of Soil Preview
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions

=> Nutrition in Animals :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Digestion in Humans (Mouth)
- ~ Lecture 3 : Tooth decay, Esophagus & Stomach
- ~ Lecture 4 : Small Intestine, Large Intestine, Anus, Diarrhoea
- ~ Lecture 5 : - Digestion in grass-eating animals, nutrition in amoeba
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions

=> Weather, Climate and Adaptations of Animals to Climates :

- ~ Lecture 1 : Introduction and Weather
- ~ Lecture 2 : Change of Seasons, Climate and types
- ~ Lecture 3 : Climate & Adaptations (Polar region)
- ~ Lecture 4 : Climate & Adaptations (tropical rainforests)
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions

# Pro Data Structure and Algorithms

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Data-Structure-and-Algorithms>

## Course Description :-

This course is designed mostly for Data Structure and Algorithms test takers.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => DSA Theoretical Test
- => DSA Practical Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

=> Data structure and Algorithms Test :

- ~ *DSA Test 1*
- ~ *DSA Test 2*
- ~ *DSA Test 3*
- ~ *DSA Test 4*
- ~ *DSA Test 5*
- ~ *DSA Test 6*
- ~ *DSA Test 7*
- ~ *DSA Test 8*
- ~ *DSA Test 9*
- ~ *DSA Test 10*

# Azure Data Factory

---

Topic Name : CLOUD

Sub-topic Name : AZURE

Course link : <https://ineuron.ai/course/Azure-Data-Factory>

## Course Description :-

Microsoft Azure Data Factory - You will understand Azure Data Factory's key components and advantages. You will be able to create, schedule and monitor simple pipelines.

## Course Features :-

- => Self paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Creating pipelines to execute Databricks notebooks
- => Designing robust pipelines to deal with unexpected scenarios
- => Creating dependencies between activities as well as pipelines
- => Scheduling the pipelines using data factory

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Curriculum details :-

- => Azure Data Factory :
  - ~ What is Data Factory Preview
  - ~ data factory in azure ecosystem Preview
  - ~ Provision Azure data factory instance
  - ~ data factory components
  - ~ data factory pipeline and activities
  - ~ data factory linked service and datasets
  - ~ data factory integration runtime
  - ~ data factory triggers
  - ~ data factory copy data activity demo
  - ~ copy data activity using author demo
  - ~ secure input and output property
  - ~ user properties
  - ~ Data factory parameters
  - ~ data flow concept
  - ~ mapping data flow
  - ~ Wrangling data flow
  - ~ monitoring
  - ~ metrics and diagnostic settings

# Build Modern ETL Data Pipeline using Informatica cloud

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA PROJECTS

Course link : <https://ineuron.ai/course/Build-Modern-ETL-Data-Pipeline-using-Informatica-cloud>

## Course Description :-

A very common use case in data engineering is to build an ETL system for a data warehouse, to have data loaded in from multiple separate databases to enable data analysts/scientists to be able to run queries on this data since the source databases are used by your applications and we do not want these analytic queries to affect our application performance and the source data. In this project, we will build an ETL system with Informatica cloud. Informatica Cloud is an on-demand subscription service that provides a complete platform for cloud integration and data management.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self-Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => Build Modern ETL Data Pipeline using Informatica cloud
- => Components of a Data Engineering Platform
- => Building ETL Pipeline
- => How to store data in the data warehouse
- => Build Dashboard using Tableau
- => Informatica Cloud

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- Build Modern ETL Data Pipeline using Informatica cloud :
  - ~ Introduction of Instructor
  - ~ Introduction to ETL and Informatica
  - ~ Project Overview
  - ~ End Notes
  - ~ Problem Description
  - ~ Understand the application scope
  - ~ Tour to existing solution
  - ~ End Notes
  - ~ Informatica Cloud set up
  - ~ Aws services
  - ~ Data Visualization Tools
  - ~ End Notes
  - ~ Solution Description
  - ~ Data Architecture
  - ~ Tour to Architecture diagram
  - ~ Cost Involved
  - ~ End Notes
  - ~ upload data to AWS S3
  - ~ set up Postgres SQL and create schemas
  - ~ EL for AWS s3 to data warehouse
  - ~ EL for app database to data warehouse
  - ~ Transformation setup

- ~ *Creat models*
- ~ *schedule monitor and alerting setup*
- ~ *Conclude the project*
- ~ *Assignments & External Resources*

# RPA Foundation

---

Topic Name : RPA

Sub-topic Name : RPA MASTERS

Course link : <https://ineuron.ai/course/RPA-Foundation>

## Course Description :-

UiPath is considered one of the fastest RPA solutions in the industry as well often 3-4x faster than other RPA products. Its ease of development is also significantly greater than competitors like Automation Anywhere, Blue Prism or Power automate where notably higher coding skills are required to make implementation much more time-consuming.

## Course Features :-

- => Online Live Classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Complete Understanding of UiPath tool from basic to advance
- => Learn to Automate - (Web, PDF, Excel, Text and Database)
- => Building workflow automation using three available workflow types
- => How to handle different types of data inside a workflow
- => Data scraping and Screen Scraping
- => Recorders like Basic, Desktop, Web and Citrix
- => Debugging and exception handling
- => Reusing automation to invoke workflows and templates
- => Automating desktop and web Applications
- => Orchestrators Concepts like Assets, Queue, Logs, Triggers and Jobs

## Requirements :-

- => Prior knowledge in Marketing
- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Basics of Python

## Curriculum details :-

- => RPA Demo :
  - ~ RPA Introduction
  - ~ RPA Tools in the Market
  - ~ UiPath Components
  - ~ UiPath Software Installation
  - ~ Types of applications can be automated
- => UiPath Interface & Workflows :
  - ~ UiPath Ribbons, Panels
  - ~ Types of workflows
  - ~ Sequence
  - ~ Flowchart
  - ~ State Machine
  - ~ Re-Frame Work
- => UiPath Core Activities :
  - ~ Package Install /uninstall
  - ~ UI Automation Activities
- => Variables, Data Types :
  - ~ What is variable?
  - ~ Types of variables
  - ~ Data Types of variables
  - ~ Assign values to variables



- ~ Variable scope

#### => Basic Activities 1 :

- ~ If activity
- ~ While
- ~ Do While
- ~ For Each
- ~ Switch

#### => Basic Activities 2 :

- ~ Flow decision
- ~ Flow Switch
- ~ Flow Chart

#### => Invoke Activities :

- ~ Invoke workflow file
- ~ In
- ~ Out
- ~ In Out
- ~ Invoke Code
- ~ Invoke Method

#### => Excel, Workbook and DataTable :

- ~ Data Table Activities
- ~ Excel Activities
- ~ Work book Activities
- ~ Relation between Data Table & Excls

#### => Recording :

- ~ Basic
- ~ Desktop
- ~ Web
- ~ Citrix

#### => Advanced UI Interaction, Selectors :

- ~ About Selectors
- ~ Selectors with Wildcards
- ~ Full versus Partial Selectors
- ~ UiPath Explorer
- ~ Image and Text Automation

#### => Screen Scraping Methods :

- ~ Full Text
- ~ Native
- ~ OCR

#### => Data Scraping :

- ~ Extract Table data

#### => NaN :

- ~ NaN

#### => PDF :

- ~ Open PDF Applications
- ~ Extract fields from PDF
- ~ PDF Activities

#### => String Methods :

- ~ Split
- ~ Replace
- ~ Index Of
- ~ Sub String

#### => Files and Folders :

- ~ Create Directory, File
- ~ Delete file
- ~ Copy file

#### => Email Automation :

- ~ Outlook Incoming and Outgoing Mails

#### => Debugging & Exception Handling :

- ~ Step by step execution
- ~ finding validation errors
- ~ Types of exceptions

#### => Data base Activities :

- ~ Establish connectivity
- ~ Retrieving data
- ~ Data Updation

#### => Ui Automation :

- ~ Find image
- ~ Image exists
- ~ Anchor base
- ~ Retry scope

#### => Re-Framework :

- ~ What is Re-frame work?
- ~ Advantages of Re-Frame work
- ~ Re-frame work Implementation

#### => Orchestrator :

- ~ Project Publish / deployment

- ~ *Create an Robot and environment*
- ~ *Connection b/w UiPath Robot and Orchestrator*
- ~ *Creating job*
- ~ *Job Execution*
- ~ *Job Scheduling*
- ~ *canceling and terminating jobs*
- ~ *How the job queue works, handling pending jobs*
- ~ *How to monitor of all Robots registered to the Orchestrator*
- ~ *How levels of error messages are communicated*
- ~ *What are UiPath Orchestrator assets?*
- ~ *How to store credentials in the Orchestrator*
- ~ *What are Orchestrator queues?*
- ~ *How to use queues to work with lists of items that are handled by Robot*

# Kubernetes

---

Topic Name : DEVOPS

Sub-topic Name : KUBERNETES

Course link : <https://ineuron.ai/course/Kubernetes>

## Course Description :-

Kubernetes is a toolkit for automating the deployment, scaling and running of containerized applications in production. This course is to teach you how to manage a containerized application infrastructure. This includes both current IT administrators and individuals interested in pursuing a cloud career.

## Course Features :-

- => Complete understanding of kubernetes
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Kubernetes Overview
- => Deploying Kubernetes
- => Kubernetes Architecture
- => Deploying Containerized Apps
- => Pods
- => Services
- => Multi-Container Pods

## Requirements :-

- => Prior Knowledge of Linux, Docker and Git
- => A system with a good internet connection
- => Your Dedication

## Instructors :-

=> Ritesh Yadav :

~ Ritesh is truly passionate about data science, machine learning and DevOps in general, he likes what he does, and is keen to learn. Currently, He is working as a Jr. Data Scientist at Ineuron.ai. He also loves to Contribute to Open Source Projects, which are mainly under CNCF Landscape. Ritesh loves to work in Cloud-Native technologies and Golang ( Go ). Apart from this, Ritesh has been actively involved in the open-source community for over a year, helping many open-source DevOps tools and CNCF Projects like Porter, Meshery, Keptn, TensorFlow, and Thanos through his contributions.

## Curriculum details :-

=> What is Kubernetes? :

- ~ What is Kubernetes? Preview
- ~ Introduction to Kubernetes
- ~ Kubernetes History
- ~ Kubernetes Architecture Preview
- ~ Kubernetes Architecture - In-depth

=> Provisioning Infrastructure :

- ~ Provisioning Kubernetes Infrastructure on AWS
- ~ Provisioning Kubernetes Infrastructure on GCP
- ~ Installing Kubernetes using kubeadm
- ~ Setting up K8 using kubeadm

=> Installing kubectl and minikube :

- ~ What is minikube? Preview
- ~ What is kubectl?
- ~ Install minikube and kubectl

=> Installing Kubernetes Using microk8s :

- ~ Setting up K8 using microk8s

=> Installing Kubernetes Using K3s :

- ~ Setting up K8's using K3's

=> Kubernetes Components :

- ~ Node & Pod
- ~ Service & Ingress
- ~ ConfigMap & Secret
- ~ Volumes
- ~ Deployment & StatefulSet

=> Create and start a minikube cluster in the local environment Kubernetes CLI :

- ~ *Commands with Example (kubectl)*
- ~ *Create a pod/deployment*
- ~ *Change the pod/deployment configuration*
- ~ *Debugging pods*
- ~ *Delete pod/deployment*
- ~ *Kubernetes YAML Configuration*
- ~ *Different attributes of a Kubernetes config file*
- ~ *Creating config files*

#### => Kubernetes Namespace :

- ~ *What is a Namespace?*
- ~ *4 Default Namespaces*
- ~ *Create a Namespace and resources*
- ~ *Why use Namespaces?*

#### => Kubernetes Healthchecks :

- ~ *What is Ingress?*
- ~ *Creating YAML Config Files for Ingress*
- ~ *How to configure Ingress in your cluster?*
- ~ *What is Ingress Controller?*
- ~ *Demo: Configure Ingress in Minikube*
- ~ *Ingress Config based on Paths*
- ~ *Ingress Config based on Domain and Subdomain*

#### => Statefulset in Kubernetes :

- ~ *What is StatefulSet?*
- ~ *Deployment of Stateful and Stateless Application*
- ~ *Deployment vs StatefulSet*
- ~ *Pod Identity*
- ~ *Scaling database applications: Master and Worker Pods*

#### => Kubernetes Services :

- ~ *What is a Service?*
- ~ *ClusterIP Services*
- ~ *Headless Services*
- ~ *NodePort Services*
- ~ *LoadBalancer Services*

#### => Volumes in Kubernetes :

- ~ *Persistent Volume (PV)*
- ~ *Persistent Volume Claim (PVC)*
- ~ *Storage Class (SC)*

#### => Deploying Microservices App to Kubernetes Cluster :

- ~ *Microservice Overview*
- ~ *Adding Dockerfile and Dockerfile Plugins*
- ~ *Adding configurations for Service Registry*
- ~ *Creating Kubernetes Config files (YAML)*
- ~ *Implementing API Gateway*
- ~ *Deploying applications to Kubernetes Cluster*
- ~ *Scaling Application*
- ~ *Kubernetes Dashboard*
- ~ *Deleting resources from Kubernetes Cluster*

# Deep Learning Foundation

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/Deep-Learning-Foundation>

## Course Description :-

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. It is a function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Learn Deep Learning, Transfer Learning and Neural Networks using the latest frameworks. Become a Deep Learning Guru!

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Certificate
- => Assignment
- => Quiz

## What you will learn :-

- => Neural Network
- => Back propagation
- => CNN

## Requirements :-

- => Computer with Internet connectivity
- => Basic Programming understanding

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Complete Road Map To Prepare For Deep Learning :
  - ~ Roadmap Preview
- => Tutorial 1- Introduction to Neural Network and Deep Learning
- => Tutorial 2- How does Neural Network Work
- => Tutorial 3-Activation Functions Part-1
- => Tutorial 4: How to train Neural Network with BackPropogation
- => Tutorial 5- How to train MultiLayer Neural Network and Gradient Descent
- => Tutorial 6-Chain Rule of Differentiation with BackPropagation
- => Tutorial 7- Vanishing Gradient Problem
- => Tutorial 8- Exploding Gradient Problem in Neural Network
- => Tutorial 9- Drop Out Layers in Multi Neural Network
- => Tutorial 10- Activation Functions Rectified Linear Unit(relu) and Leaky Relu Part 2
- => Deep Learning-Activation Functions-Elu, PRelu,Softmax,Swish And Softplus
- => Tutorial 11- Various Weight Initialization Techniques in Neural Network
- => Tutorial 12- Stochastic Gradient Descent vs Gradient Descent
- => Tutorial 13- Global Minima and Local Minima in Depth Understanding
- => Tutorial 14- Stochastic Gradient Descent with Momentum
- => Tutorial 15- Adagrad Optimizers in Neural Network
- => Tutorial 16- AdaDelta and RMSprop optimizer
- => Deep Learning-All Optimizers In One Video-SGD with Momentum,Adagrad,Adadelata,RMSprop,Adam Optimizers
- => Tutorial 17- Create Artificial Neural Network using Weight Initialization Tricks
- => Keras Tuner Hyperparameter Tuning-How To Select Hidden Layers And Number of Hidden Neurons In ANN
- => Tutorial 18- Hyper parameter Tuning To Decide Number of Hidden Layers in Neural Network

=> Tutorial 19- Training Artificial Neural Network using Google Colab GPU

=> Tutorial 20- Convolution Neural Network vs Human Brain

=> Tutorial 21- What is Convolution operation in CNN?

=> Tutorial 22- Padding in Convolutional Neural Network

=> Tutorial 23- Operation Of CNN(CNN vs ANN)

=> Tutorial 24- Max Pooling Layer In CNN

=> Tutorial 25- Data Augmentation In CNN-Deep Learning

=> Tutorial 26- Create Image Dataset using Data Augmentation using Keras-Deep Learning-Data Science

=> Tutorial 27- Create CNN Model and Optimize using Keras Tuner- Deep Learning

=> Tutorial 28- Create CNN Model Using Transfer Learning using Vgg 16, Resnet

=> Tutorial 29- Why Use Recurrent Neural Network and Its Application

=> Tutorial 30- Recurrent Neural Network Forward Propagation With Time

=> Tutorial 31- Back Propagation In Recurrent Neural Network

=> Tutorial 32- Problems In Simple Recurrent Neural Network

=> Tutorial 33- Installing Cuda Toolkit And cuDNN For Deep Learning

=> Tutorial 34- LSTM Recurrent Neural Network In Depth Intuition

=> Word Embedding - Natural Language Processing| Deep Learning

=> Implementing Word Embedding Using Keras- NLP | Deep Learning

=> Develop your Neural Network Like A Google Deep Learning Developer

=> Kaggle Fake News Classifier Using LSTM- Deep Learning| Natural Language Processing

=> Stock Price Prediction And Forecasting Using Stacked LSTM- Deep Learning

=> Bidirectional RNN In-depth Intuition- Deep Learning Tutorial

=> Implement Kaggle Fake News Classifier Using Bidirectional LSTM RNN

=> Sequence To Sequence Learning With Neural Networks| Encoder And Decoder In-depth Intuition

=> Develop Your First Deep Learning End To End Project As A Beginner In Data Science in 30 minutes

=> Encoder And Decoder- Neural Machine Learning Language Translation Tutorial With Keras- Deep Learning

=> Problems With Encoders And Decoders- In-depth Intuition

=> Live Session- Understanding Attention Models Architecture And Maths Intuition- Deep Learning

=> Live Session- Encoder Decoder, Attention Models, Transformers, Bert Part 1

=> Live- Attention Models, Transformers And Bert In depth Intuition Deep Learning- Part 2

=> Live -Transformers In-depth Architecture Understanding- Attention Is All You Need

=> How To Train Deep Learning Models In Google Colab- Must For Everyone

=> AlexNet Architecture In-depth-Discussion Along With Code-Deep Learning Advanced CNN

=> VGGNET Architecture In-depth Discussion Along With Code -Deep Learning Advanced CNN

=> Hummingbird-Run Traditional Machine Learning model on Deep Neural Network frameworks-Data Science

=> Lets Implement LSTM RNN Models For Univariate Time Series Forecasting- Deep Learning

=> TensorDash- How To Monitor Your Deep Learning Model Metrics, Loss, Accuracy Using Mobile App

=> Handling Imbalanced Dataset Using Cost Sensitive Neural Networks- Credit Card Fraud Detection

=> 500+ Machine Learning And Deep Learning Projects All At One Place

=> Google Colab Pro Vs Colab Free- Benefits Of Using Colab Pro- How To Access From India

# R Programming Job Preparation

---

Topic Name : PROGRAMMING

Sub-topic Name : R

Course link : <https://ineuron.ai/course/R-Programming-Job-Preparation>

## Course Description :-

R had over 2 million users globally in 2012, according to Oracle, reinforcing R's position as the premier programming language in statistics and data research. Every year, the number of R users increases by roughly 40%, and a rising number of businesses are incorporating it into their daily operations. This course is designed for individuals who are willing to start their journey with R programming to get into the field of Data Science or Machine Learning.

## Course Features :-

- => Roadmap
- => Resume preparation
- => Interview Questions
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Statistics overview
- => In depth discussion for various types of questions asked in interviews and how to approach them
- => Discussion on resume building.

## Requirements :-

- => Prior Understanding of R Programming.
- => A system with a decent internet connection
- => Dedication

## Instructors :-

- => Shlok Pandey :
  - ~ Content Developer (Deep Learning)

## Curriculum details :-

- => Course Introduction :
  - ~ Explain What is R? Preview
  - ~ What is GUI in R ?
  - ~ What is CLI in R ?
  - ~ What is Data Mining & what Data Miners do in R?
  - ~ Who & when was R discovered?
  - ~ Why should you adopt R Programming language ?
- => Introduction to basics of R :
  - ~ What are Programming Features of R? Preview
  - ~ What are the applications of R?
  - ~ Compare R with Other technologies.
  - ~ Why R is Important ?
  - ~ Is R a slow language?
  - ~ Explain main features to write R code that runs faster?
  - ~ What is SAS and SPSS in R?
  - ~ Why is R important for data science?
  - ~ Why is R Good for business?
  - ~ What is Visualization in R?
  - ~ What are R topical programming and statistical relevance?
  - ~ What are the statistical and programming features of R?
  - ~ What are the advantages of R?
  - ~ What are the disadvantages of R?
  - ~ Why R language?
- => R Analysis :
  - ~ What is Predictive Analysis in R?
  - ~ What is the Predictive analysis process in R?
  - ~ What is the need for Predictive Analysis in R?
  - ~ What is Descriptive analysis in R?
  - ~ What are Descriptive analysis methods in R?
- => R Datatypes :
  - ~ What is R studio and how to use it?

- ~ What are R data types?
- ~ How many types of data types are provided by R?

=> R Vector :

- ~ What is the main difference between an Array and a matrix?
- ~ What is R vector?
- ~ How many types of vectors are present in R?
- ~ What is an Atomic vector and how many types of atomic vectors are present in R?
- ~ What is recycling of elements in an R vector? Give an example.

=> R Lists :

- ~ What is R lists?
- ~ Explain how to create a list in R?
- ~ Explain how to access list elements in R?
- ~ Explain how to manipulate list elements in R?
- ~ Explain how to generate lists in R?
- ~ Explain how to operate on lists in R?
- ~ Can we update and delete any of the elements in a list?
- ~ How many types of object are present in R?

=> R Functions :

- ~ What are R Functions?
- ~ What are features of R functions?
- ~ What is function definition?
- ~ What are the components of R functions?



# Data Analytics Bootcamp

---

Topic Name : DATA ANALYTICS

Sub-topic Name : BUSINESS ANALYTICS MASTERS

Course link : <https://ineuron.ai/course/Data-Analytics-Bootcamp>

## Course Description :-

Data analytics is a field that combines information technology with business management to assist businesses in managing data, gaining business insights from it, and identifying expansion prospects. The course will provide you all of the skills you'll need to work as a data analyst.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Basics of MySQL
- => Playing with data
- => FOREIGN KEY and JOINS
- => Tables and Matrix in Power BI
- => Power BI
- => Tableau
- => Alteryx
- => Excel

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Pawan Lalwani :

~ Pawan is a highly skilled and self motivated trainer who has expertise in various business intelligence tools like Power BI, Tableau and Microsoft Excel. He comes with 10 years of experience in training individuals in different industry sectors like Banking, Finance, Healthcare, IT, Automobile, Manufacturing and Pharmaceutical.

## Curriculum details :-

=> Introduction and installation of MySQL :

- ~ Introduction to section 1
- ~ MySQL introduction - 5 points to know
- ~ MySQL Installation MAC

=> Basics of MySQL :

- ~ Introduction to section 2
- ~ Creating and dropping database - Startup
- ~ Resolving the issue for future
- ~ Creating your first table
- ~ Adding values to canon table
- ~ Answering customer question
- ~ Introduction to section 3

=> Playing with data :

- ~ Primary key, default, and NULL
- ~ Table with primary key and default values
- ~ Testing the new table
- ~ Adding new values and answering questions
- ~ Update in customers table
- ~ Delete from the customer's table

=> More on functions :

- ~ Introduction to section 4
- ~ Understand the new Ico user DB
- ~ Task for CONCAT

- ~ Task for REPLACE
- ~ task for SUBSTRING
- ~ Task for reverse and CHAR\_LENGTH
- ~ Task for case conversion and DOCS

=> Answering some DB questions :

- ~ Introduction to section 5
- ~ A task on DISTINCT
- ~ A task for ORDER BY
- ~ A task on LIMIT
- ~ Match the pattern
- ~ A task on COUNT
- ~ SQL MODES and GROUP BY
- ~ MIN MAX and SUBQUERIES
- ~ GROUP BY with MAX and MIN
- ~ SUM and AVERAGE with GROUP BY
- ~ A task on AND OR
- ~ A task in RANGE based selection
- ~ CASE THEN - multiple range selection

=> A pinch of theory :

- ~ Introduction to section 6
- ~ Data type for INTEGER and STRING
- ~ Data type for DATE, DATETIME and JSON
- ~ DATE TIME code Example
- ~ Get the date and time
- ~ Lets join tom and jerry tables
- ~ Types of JOIN

=> FOREIGN KEY and JOINS :

- ~ Introduction to section 7
- ~ Code talk over FOREIGN keys
- ~ Understand a new database
- ~ A task on INNER join
- ~ ONE to MANY and MANY TO MANY
- ~ Join more 3 or more tables
- ~ A task on LEFT JOIN
- ~ A task on RIGHT JOIN
- ~ FULL OUTER join and UNION tasks

=> A pinch of more theory :

- ~ Introduction to section 8
- ~ Database engines - INNODB and more
- ~ ACID in database

=> A 30 Task assignment for movie DB :

- ~ Introduction to section 9
- ~ How to practice database works - FILM

=> Final exam - single attempt :

- ~ MYSQL Outro and some free resources

=> Introduction :

- ~ Introduction to Power BI
- ~ Download, Install and Update Features in Power bi
- ~ Introduction to Alteryx
- ~ Download and Install Alteryx
- ~ User Interface of Alteryx

=> Basic Charts in Power BI :

- ~ Basic Charts in Power BI Desktop
- ~ Column Chart in Power BI
- ~ Stacked Column Chart in Power BI
- ~ Pie Chart in Power BI
- ~ Donut Chart in Power BI
- ~ Funnel Chart in Power BI
- ~ Ribbon Chart
- ~ Include and Exclude
- ~ Export data from Visual

=> Working with Maps :

- ~ Creating a Map in Power BI
- ~ Filled Map
- ~ Map with Pie Chart
- ~ Formatting in Map
- ~ Change Background in Map
- ~ Map of India in Power BI
- ~ Map of Australia in Power BI

=> Tables and Matrix in Power BI :

- ~ Table and Matrix in Power BI
- ~ Creating a Table in Power BI
- ~ Formatting a Table
- ~ Conditional Formatting in Table
- ~ Aggregation in Table
- ~ Matrix in Power BI
- ~ Conditional Formatting in Matrix
- ~ Hierarchies in Matrix
- ~ Sub-Total and Total in Matrix
- ~ Number Formatting in Table

=> Other Charts in Power BI :

- ~ *Other Charts in Power BI*
- ~ *Line Chart in Power BI*
- ~ *Drill Down in Line Chart*
- ~ *Area Chart in Power BI*
- ~ *Line vs Column Chart in Power BI*
- ~ *Scatter Plot in Power BI*
- ~ *Waterfall Chart in Power BI*
- ~ *TreeMap in Power BI*
- ~ *Gauge Chart in Power BI*

#### => Cards and Filters :

- ~ *Cards and Filters in Power BI*
- ~ *Number Card*
- ~ *Text Card*
- ~ *Formatting of Text Card*
- ~ *Date Card*
- ~ *Date Card (Relative Filtering)*
- ~ *Multi-Row Card*
- ~ *Filter on Visual*
- ~ *Filter on This Page*
- ~ *Filter on All Pages*
- ~ *Drillthrough in Power BI*

#### => Slicers in Power BI :

- ~ *Slicers in Power BI*
- ~ *Text Slicers in Power BI*
- ~ *Formatting a Text Slicer*
- ~ *Date Slicers in Power BI*
- ~ *Formatting a Date Slicer*
- ~ *Number Slicers in Power BI*

#### => Advanced Charts in Power BI :

- ~ *Advanced Charts in Power BI*
- ~ *Animated Bar Chart Race*
- ~ *Drill down donut Chart*
- ~ *Drill down Column chart*
- ~ *Word Cloud in Power BI*
- ~ *Sankey Chart in Power BI*
- ~ *Infographic in Power BI*
- ~ *Play Axis in Power BI*
- ~ *Scroller in Power BI*
- ~ *Sunburst Chart in Power BI*
- ~ *Histogram in Power BI*

#### => Objects in Power BI :

- ~ *Insert Image in Power BI*
- ~ *Insert Text in Power BI*
- ~ *Insert Shapes in Power BI*
- ~ *Insert Buttons in Power BI*
- ~ *Web URL Action in Power BI*
- ~ *Page Navigation Action in Power BI*
- ~ *Bookmark Action in Power BI*
- ~ *Drillthrough Action in Power BI*

#### => Power BI Service Introduction :

- ~ *Create a Superstore Report in Power BI*
- ~ *Create an Account on Power BI Service*
- ~ *Publish Report to Power BI Service Account*
- ~ *Export Power BI Report to PPT, PDF or PBIX*
- ~ *Comment, Share and Subscribe to Power BI Report*
- ~ *Create a Dashboard in Power BI Service*
- ~ *Problem in Power BI Dashboard and its solution*
- ~ *Automatic Refresh in Power BI using Gateway*

#### => Power Query - Text Functions :

- ~ *Text Functions in Power Query (Power BI)*
- ~ *Merge Columns in Power Query (Power BI)*
- ~ *Split and Trim in Power Query (Power BI)*
- ~ *Upper, Lower and ProperCase in Power Query (Power BI)*
- ~ *Prefix and Suffix in Power Query (Power BI)*
- ~ *Left, Right and Mid Functions in Power Query (Power BI)*
- ~ *Extract Text with Delimiters*

#### => Power Query - Date Functions :

- ~ *Date Functions in Power Query (Power BI)*
- ~ *Year, Quarter, Month and Day Functions in Power Query (Power BI)*
- ~ *Find Difference between Dates in Power Query (Power BI)*
- ~ *Month and Day Name in Power Query (Power BI)*
- ~ *Day, Week of Month, Year in Power Query (Power BI)*
- ~ *Extract Date, Time in Power Query (Power BI)*
- ~ *Calculate Age in Power Query (Power BI)*
- ~ *Day of Year, Quarter, Month in Power Query (Power BI)*

#### => Power Query - Number Functions :

- ~ *Number Functions in Power Query (Power BI)*
- ~ *Basic Number Functions in Power Query (Power BI)*
- ~ *Percentage, Percent Of, Module in Power Query (Power BI)*
- ~ *Round Functions in Power Query (Power BI)*
- ~ *IsEven, IsODD, Sign in Power Query (Power BI)*

#### => Power Query - Append Files :

- ~ Append multiple CSV files in a folder in Power Query (Power BI)
- ~ Append multiple excel sheets, Tables in Power Query (Power BI)
- ~ Append Excel sheets or Tables with different columns in Power BI
- ~ Append multiple Excel files from a folder in Power BI
- ~ Append different data sources in Power BI

#### => Power Query - Merge Files :

- ~ Merge Files and Tables in Power BI
- ~ Merge Sheets or Tables in Power Query (Power BI)
- ~ Merge Data from multiple Excel files or Workbooks in Power BI
- ~ Merge data from different data sources in Power Query (Power BI)
- ~ Merge data having multiple criteria in Power BI

#### => Power Query - Conditional Columns :

- ~ Conditional Column and Column from example in Power BI
- ~ Column from examples in Power BI - Split Text
- ~ Column from examples in Power BI - Merge Columns
- ~ Column from Examples in Power BI - Date
- ~ Column from Examples in Power BI - Alphanumeric
- ~ Conditional Column in Power BI - One Column
- ~ Conditional Column in Power BI - two columns
- ~ Conditional Column in Power BI - Compare two columns
- ~ Conditional Column in Power BI - on Dates

#### => Power Query - - Important Topics :

- ~ Very Important Topics in Power Query (Power BI)
- ~ Fill Down in Power BI
- ~ Grouping in Power Query (Power BI)
- ~ Transpose in Power Query (Power BI)
- ~ Unpivot In Power Query (Power BI)
- ~ Data Types in Power Query (Power BI)
- ~ Replace Errors and Values in Power Query (Power BI)
- ~ Keep and Remove Rows in Power Query (Power BI)
- ~ Add, Remove and Goto Columns in Power Query (Power BI)

#### => M Language Introduction :

- ~ M Language in Power Query
- ~ Introduction to M Language
- ~ IsIn Date Functions in M Language - Power BI
- ~ Add and Subtract Date M Functions in Power BI
- ~ Basic Date M Functions in Power BI
- ~ Basic Text M Functions in Power BI
- ~ Simple M Code in Power BI
- ~ Trick to get all 900+ M Functions in Power BI

#### => Introduction to tableau :

- ~ Tableau Introduction
- ~ Download and Install Tableau
- ~ Tableau Vs Excel

#### => Charts - 1 :

- ~ Column Chart
- ~ Horizontal Bar Chart
- ~ Stacked Column Chart
- ~ Stacked Bar Chart
- ~ Keep Only, Exclude
- ~ Keep Only, Exclude2, Normal
- ~ Publish to Tableau Public

#### => Charts - 2 :

- ~ Pie Chart
- ~ Multiple Pie Chart
- ~ TreeMap\_Editing
- ~ Packed Bubble Chart
- ~ Word Cloud OR Word Map
- ~ Formatting payal

#### => Charts - 3 :

- ~ Data Types in Tableau
- ~ Filled Map
- ~ Symbol Maps
- ~ India Map
- ~ Histogram

#### => Charts - 4 :

- ~ Text Table
- ~ Text Table with Multiple Measures
- ~ Measure Names and Measure Values
- ~ Line Chart
- ~ Line Chart with Multiple Measures
- ~ Discrete Vs Continuous Line Chart
- ~ Discrete Vs Continuous

#### => Charts - 5 :

- ~ Lollipop Chart
- ~ Line Vs Column Chart
- ~ Dual Axis Chart
- ~ Column vs Shapes
- ~ Bar in Bar Chart

#### => Charts - 6 :

- ~ Calculated fields

- ~ Conditional Column Chart
- ~ Column chart with Shapes based on condition
- ~ Conditional Maps

#### => Charts - 7 :

- ~ Map with Pie Chart
- ~ Map with WMS

#### => Charts - 8 :

- ~ Funnel Chart
- ~ Advanced Funnel Chart
- ~ Calendar
- ~ Dumbbell Chart
- ~ Donut Chart
- ~ Multiple Donut Chart

#### => Charts - 9 :

- ~ Bullet Chart 1
- ~ Bullet Chart 2
- ~ Table Calculations Part 1
- ~ Table Calculations - Compute Using - Part 2
- ~ Table Calculations - Relative - Part 3
- ~ Bump Chart
- ~ Bump Chart with Circle
- ~ 100 Percent Stacked Column Chart

#### => Charts - 10 :

- ~ Scatter Plot
- ~ Scatter Plot with Images OR Shapes
- ~ Bubble Chart
- ~ Animation - Column Chart
- ~ Animation - Line Chart
- ~ Animation - Column vs Line Chart

#### => Charts - 11 :

- ~ Heat Maps
- ~ Heat Map with Shapes
- ~ Heat Map with Conditional Formatting
- ~ Pareto Chart
- ~ Rounded Bar Chart

#### => IN/Out Tab :

- ~ ALTERYX - Get Data from Excel
- ~ ALTERYX - Get Data from CSV
- ~ ALTERYX - Append All CSV files
- ~ ALTERYX - Browse Tool
- ~ ALTERYX - Output Tool - Update Existing Data
- ~ ALTERYX - Directory Tool
- ~ ALTERYX - Directory Tool - Specific Files
- ~ ALTERYX - Text Input Tool
- ~ ALTERYX - Date and Time Tool

#### => Preparation Tab :

- ~ ALTERYX - Auto Field Tool
- ~ ALTERYX - Data Cleansing Tool Part 1
- ~ ALTERYX - Data Cleansing Tool - Part 2
- ~ ALTERYX - Filter Tool (Text Example)
- ~ ALTERYX - Filter Tool (Number Example)
- ~ ALTERYX - Filter Tool ( Date Example )
- ~ ALTERYX - FORMULA TOOL ( Basic Example )
- ~ ALTERYX - FORMULA TOOL - (Multiple Examples)
- ~ ALTERYX - GENERATE ROWS TOOL
- ~ ALTERYX - IMPUTATION TOOL
- ~ ALTERYX - MULTI-FIELD BINNING TOOL
- ~ ALTERYX - MULTI-FIELD FORMULA
- ~ ALTERYX - MULTI ROW FORMULA
- ~ ALTERYX - RANDOM % SAMPLE TOOL
- ~ ALTERYX - SAMPLE TOOL
- ~ ALTERYX - RECORD ID TOOL
- ~ ALTERYX - SELECT TOOL
- ~ ALTERYX - SORT
- ~ ALTERYX - CREATE SAMPLE TOOL
- ~ ALTERYX - TILE TOOL
- ~ ALTERYX - UNIQUE TOOL

#### => Join Tab :

- ~ ALTERYX - APPEND FIELDS TOOL
- ~ ALTERYX - FIND AND REPLACE TOOL
- ~ ALTERYX - FUZZY MATCH TOOL
- ~ ALTERYX - JOIN TOOL
- ~ ALTERYX - JOIN MULTIPLE TOOL
- ~ ALTERYX - UNION TOOL
- ~ REGEX TOOL
- ~ Text To Columns

#### => Transform Tab :

- ~ ALTERYX - CROSS TAB Tool
- ~ ALTERYX \_ TRANSPOSE Tool
- ~ ALTERYX - RUNNING TOTAL Tool
- ~ ALTERYX \_ SUMMARIZE TOOL

#### => Reporting Tab :

- ~ ALTERYX - TABLE TOOL
- ~ ALTERYX - INTERACTIVE CHART Tool
- ~ ALTERYX - JOIN TABLE AND CHART
- ~ ALTERYX - ADD ANNOTATION
- ~ ALTERYX - REPORT TEXT TOOL
- ~ ALTERYX - REPORT HEADER TOOL
- ~ ALTERYX - REPORT FOOTER TOOL
- ~ ALTERYX - REPORT LAYOUT TOOL

=> Documentation Tab :

- ~ ALTERYX - COMMENT TOOL
- ~ ALTERYX - EXPLORER TOOL
- ~ ALTERYX - CONTAINER TOOL

=> Case Studies :

- ~ Study 1
- ~ Study 2
- ~ Study 3
- ~ Study 4
- ~ Study 5

=> Microsoft Excel Fundamentals :

- ~ Launching Excel
- ~ Microsoft Excel Startup Screen
- ~ Introduction to the Excel Interface
- ~ Customizing the Excel Quick Access Toolbar
- ~ More on the Excel Interface
- ~ Understanding the Structure of an Excel Workbook
- ~ Saving an Excel Document
- ~ Opening an Existing Excel Document
- ~ Common Excel Shortcut Keys

=> Entering and editing text and formulas :

- ~ Entering Text to Create Spreadsheet Titles
- ~ Working with Numeric Data in Excel
- ~ Entering Date Values in Excel
- ~ Working with Cell References
- ~ Creating Basic Formulas in Excel
- ~ Relative Versus Absolute Cell References in Formulas
- ~ Understanding the Order of Operation

=> Working with basic excel functions :

- ~ The structure of an Excel Function
- ~ Working with the SUM() Function
- ~ Working with the MIN() and MAX() Functions
- ~ Working with the AVERAGE() Function
- ~ Working with the COUNT() Function
- ~ Adjacent Cells Error in Excel Calculations
- ~ Using the AutoSum Command
- ~ Excel's AutoSum Shortcut Key
- ~ Using the AutoFill Command to Copy Formulas

=> Modifying an excel worksheet :

- ~ Moving and Copying Data in an Excel Worksheet
- ~ Inserting and Deleting Rows and Columns
- ~ Changing the Width and Height of Cells
- ~ Hiding and Unhiding Excel Rows and Columns
- ~ Renaming an Excel Worksheet
- ~ Deleting an Excel Worksheet
- ~ Moving and Copying an Excel Worksheet

=> Formatting data in an excel worksheet :

- ~ Working with Font Formatting Commands
- ~ Changing the Background Color of a Cell
- ~ Adding Borders to Cells
- ~ Excel Cell Borders Continued
- ~ Formatting Data as Currency Values
- ~ Formatting Percentages
- ~ Using Excel's Format Painter
- ~ Creating Styles to Format Data
- ~ Merging and Centering Cells
- ~ Using Conditional Formatting
- ~ Editing Excel Conditional Formatting

# Snowflake Beginners

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Snowflake-Beginners>

## Course Description :-

Cloud Data Warehouse is the next big thing. Learn What is Snowflake Cloud Data Warehouse and its architecture. Build a highly scalable, high performance next-gen modern data warehouse for your company. The course is designed in beginner-friendly, helping you to understand the basics of cloud, SAAS and it all works together in the background.

## Course Features :-

- => Practical Implementation
- => Downloadable resources
- => Class Recordings
- => Quiz Questions
- => Completion Certificate

## What you will learn :-

- => Snowflake Architecture
- => Working with Snowflake UI
- => Features of Snowflake
- => Setup Connectors

## Requirements :-

- => Prior Knowledge of Databases Language
- => Interest to learn
- => Your dedication

## Instructors :-

=> MD Imran :

~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

=> Course Overview :

~ Course overview Preview

=> Master the basics :

- ~ What is data warehousing?
- ~ Why Snowflake?
- ~ What is Snowflake? Preview
- ~ Snowflake working architecture
- ~ Snowflake signup

=> Introduction to snowflake :

- ~ Key Concepts & Architecture
- ~ Supported Cloud Platforms
- ~ Supported Cloud Regions
- ~ Snowflake Classic Legacy WebUI
- ~ Snowflake Modern Web UI
- ~ Snowflake Unique Features

=> Snowflake Database SchemaTable & Container Hierarchy :

- ~ Create database and schema under snowflake account
- ~ Create a table with different data type
- ~ Create a table with text field
- ~ Create a table with date and timestamp field
- ~ Create a table with Upper, Lower & Mixed Case
- ~ Constraints in Snowflake

=> Partner connect :

- ~ Setup connectors part 1 Preview
- ~ Setup connectors part 2
- ~ Setup connectors part 3

# R Programming Projects

---

Topic Name : PROGRAMMING

Sub-topic Name : R

Course link : <https://ineuron.ai/course/R-Programming-Projects>

## Course Description :-

This course will help you gain extensive knowledge by building various projects on real-time datasets. This will not only clear all your concepts but also give you in-depth knowledge and hands-on experience of working with R programming on various industry level projects.

## Course Features :-

- => Roadmap
- => Quizzes
- => Complete project implementation
- => Assignments
- => Challenges
- => Downloadable resources

## What you will learn :-

- => Architecture design
- => Solution building
- => Building webapps
- => Machine learning uses
- => Working on real time problems
- => Project presentation skills

## Requirements :-

- => Prior understanding of R programming.
- => Basic knowledge of HTML and CSS
- => A System with internet connection.
- => Your dedication.

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Webscraping in R :
  - ~ Web scraping introduction part 1 Preview
  - ~ Web scraping load the data part 2 Preview
  - ~ Web scraping part 3
  - ~ Web scraping part 4
  - ~ Web scraping part 5



# Ansible

---

Topic Name : DEVOPS

Sub-topic Name : ANSIBLE

Course link : <https://ineuron.ai/course/Ansible>

## Course Description :-

Ansible is an automation engine that aid with infrastructure provisioning, operating system configuration, application deployment, and much more. Developers use Ansible to automate the creation of development environments that are identical to production settings.

The same playbooks can also be used to automate the construction of staging and production environments in operations.

## Course Features :-

- => Source code
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Ansible fundamentals
- => Ansible architecture
- => Installation
- => Inventory
- => Modules
- => Playbooks
- => Templates
- => Roles

## Requirements :-

- => Prior knowledge of Linux
- => A system with good internet connection
- => Your dedication

## Instructors :-

=> Ritesh Yadav :

*~ Ritesh is truly passionate about data science, machine learning and DevOps in general, he likes what he does, and is keen to learn. Currently, He is working as a Jr. Data Scientist at Ineuron.ai. He also loves to Contribute to Open Source Projects, which are mainly under CNCF Landscape. Ritesh loves to work in Cloud-Native technologies and Golang ( Go ). Apart from this, Ritesh has been actively involved in the open-source community for over a year, helping many open-source DevOps tools and CNCF Projects like Porter, Meshery, Keptn, TensorFlow, and Thanos through his contributions.*

## Curriculum details :-

=> Introduction :

- ~ What is Ansible? Preview
- ~ Getting started with Ansible
- ~ Ansible components

=> What is configuration management? :

- ~ What is infrastructure provisioning? Preview
- ~ What is IT automation?
- ~ Ansible use cases

=> How Ansible works? :

- ~ Ansible architecture
- ~ Ansible modules

=> Ansible concepts :

- ~ Controller in Ansible
- ~ Managed node in Ansible
- ~ Ansible modules

=> Ansible controller node setup :

- ~ Ansible SSH key generation
- ~ Ansible keygen
- ~ Ansible installation on Linux

=> Managing managed node via inventory Ansible modules :

- ~ Setting managed node in Ansible
- ~ Setting up Ansible modules
- ~ Using Ansible modules in AWS, Azure

=> Executing single tasks via ad-hoc commands :

- ~ Working with Ansible commands

=> Ansible playbook :

- ~ *Ansible playbook variables*
- ~ *Ansible playbook tasks*
- ~ *Ansible playbook handlers*
- ~ *Ansible playbook roles*
- ~ *Running Ansible playbook and target new EC2 instance host*

=> Ansible vault and Ansible galaxy

# Pro C++ Programming Language

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-C++-Programming-Language>

## Course Description :-

This course is designed mostly for C++ test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> C++ Theoretical Test

=> C++ Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> C++ Coding Test :

~ C++ Test 1

~ C++ Test 2

~ C++ Test 3

~ C++ Test 4

~ C++ Test 5

~ C++ Test 6

# Tableau Job Preparation

---

Topic Name : DATA ANALYTICS

Sub-topic Name : TABLEAU INTERVIEW QUESTIONS

Course link : <https://ineuron.ai/course/Tableau-Job-Preparation>

## Course Description :-

Tableau enables critical decision-makers to learn to display data and uncover data patterns such as customer purchase behaviour, sales trends, or production bottlenecks. We are presenting an end-to-end education for you to make a difference and join the industry. This course will cover all of the capabilities of Tableau that enable access to explore, experiment with, prepare, and present data quickly and beautifully throughout an organisation.

## Course Features :-

=> N

=> u

=> I

=> I

## What you will learn :-

=> N

=> u

=> I

=> I

## Requirements :-

=> N

=> u

=> I

=> I

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Tableau :

- ~ What is Tableau ? Preview
- ~ What are measures and dimensions ? Preview
- ~ What are the Different tableau Products ?
- ~ What are the Datatypes supported in Tableau ?
- ~ Define LOD Expression.
- ~ What is meant by "discrete" and "continuous" in Tableau ?
- ~ What is the difference between .twb and .twbx extension ?
- ~ What are the different Connections you can make with your dataset ?
- ~ What are filters in Tableau ?
- ~ Compare blending and joining in Tableau.
- ~ Can you explain the Tableau design flow ?
- ~ How is the Context Filter different from other Filters ?
- ~ What are Parameters ?
- ~ Give an Example of a story in Tableau.
- ~ What is aggregation and disaggregation of data ?
- ~ What is Calculated field and how you will create one ?
- ~ What is the difference between a Heatmap and Treemap ?
- ~ What is the use of user roles in Tableau ?
- ~ How do you generally perform load testing in Tableau ?
- ~ What is Tableau data engine ?
- ~ What are the components in a dashboard ?
- ~ How do you embed views into webpages ?
- ~ What is the difference between published data sources and embedded data sources in Tableau?
- ~ How to view underlying data sources in Tableau ?
- ~ State some ways to improve the performance in Tableau ?
- ~ Define dual-axis.
- ~ Define Gantt chart.
- ~ State some reasons for the low performance in Tableau.Explain in details.
- ~ State a few charts that we should not use with valid reasons.
- ~ Do we have any data limitations in Tableau public ?
- ~ Is it possible to view SQL commands generated by Tableau ?

- ~ What are the difference between RANK and INDEX ?
- ~ Can R and Tableau be used together ?
- ~ Can you get values from two different sources as a single input into parameter ?
- ~ Suppose without any line/bar chart I want to design a view to show the region-wise profit and sale. How should I approach it? Explain.
- ~ How to generate longitude and latitude in Tableau ?
- ~ How to increase size of pie in Tableau ?
- ~ Why use a hierchical field in Tableau ?
- ~ List the types of maps available in Tableau.

# Crack the Tech Interviews

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/Crack-the-Tech-Interviews>

## Course Description :-

Algorithmic programming techniques are a must-have skill. Learn Algorithms through programming and puzzle solving to advance your Software Engineering or Data Science career. Then, implement each algorithmic problem in this program to ace coding interviews.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Non Tech Round Preparation
- => Array interview problems and solutions
- => String interview problems and solutions
- => Recursion interview problems and solutions
- => Linked list interview problems and solutions
- => Math interview problems and solutions
- => Stack and Queue interview problems and solutions
- => Sorting interview problems and solutions
- => Trees interview problems and solutions
- => Graphs interview problems and solutions
- => Dynamic programming interview problems and solutions

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Preparing for the interview :

- ~ FAQ before taking this course
- ~ How to take this course

=> Non Tech Round Preparation :

- ~ Are you ready for interviews
- ~ Your resume needs more work
- ~ 8 point resume check list
- ~ Handle experience section
- ~ FAANG interview process
- ~ How to find jobs
- ~ 3 pillars of answers
- ~ Tell me about yourself
- ~ Why our company
- ~ Recent project problem
- ~ Tell me your weakness

=> Array interview problems and solutions :

- ~ Binary search - How to explain in interview
- ~ Binary search - recursion explanation
- ~ Binary search - iterative explanation
- ~ Rotation of array - expected explanation
- ~ Pivot problem code

- ~ Search in rotated array - Theory
- ~ Search in rotated array - Code
- ~ Find by comparison
- ~ Find by comparison - crafting code
- ~ Target Triplet
- ~ Target Triplet Code expected solution
- ~ Move to 1 side problems
- ~ Move to 1 side code
- ~ Sell at max profit problem
- ~ Sell at max profit code

=> String interview problems and solutions :

- ~ Word in a sentence problem
- ~ Word in a sentence problem Code
- ~ Inplace duplicates
- ~ Inplace duplicates code
- ~ Longest Substring
- ~ Longest Substring Code
- ~ Palindrome makes and breaks interviews
- ~ Palindrome makes and breaks interviews CODE

=> Recursion interview problems and solutions :

- ~ PreReq for recursion
- ~ Classic fibonacci problem but with diary
- ~ Classic fibonacci problem but with diary code
- ~ Popular subset problem
- ~ Popular subset problem CODE
- ~ Decimal to binary for Round 1
- ~ Decimal to binary for Round 1 Code
- ~ NearBy Duplicates
- ~ NearBy Duplicates Code
- ~ Pascal nth row
- ~ Pascal nth row Code

=> Linked list interview problems and solutions :

- ~ Approach for linked list and head
- ~ Approach for linked list and head Code
- ~ Insert in doubly linked list
- ~ Insert in doubly linked list Code
- ~ Tail insertion in doubly linked list
- ~ Tail insertion in doubly linked list Code
- ~ Deleting a val in doubly linked list
- ~ Deleting a val in doubly linked list Code
- ~ Reverse a doubly linkedlist with traveller
- ~ Reverse a doubly linkedlist with traveller Code
- ~ Floyds loop detection
- ~ Floyds loop detection Code
- ~ Merge 2 linked list
- ~ Merge 2 linked list code

=> Math interview problems and solutions :

- ~ Not counted in
- ~ Permutation explanation on White board
- ~ Permutation explanation code
- ~ kth Permutation theory explained
- ~ kth Permutation code
- ~ Bit manipulation
- ~ Bit manipulation Code

=> Stack and Queue interview problems and solutions :

- ~ Stack using queue
- ~ Stack using queue Code
- ~ Stack using queue - Approach 2
- ~ Stack using queue - Approach 2 Code
- ~ Queue using stack
- ~ Queue using stack Code
- ~ Queue using stack - approach 2
- ~ Queue using stack - approach 2 Code
- ~ Stock Spanning
- ~ Stock Spanning Code
- ~ Valid brackets
- ~ Valid brackets Code

=> Sorting interview problems and solutions :

- ~ Bubble Sort
- ~ Bubble Sort Code
- ~ Selection Sort
- ~ Selection Sort Code
- ~ Insertion sort
- ~ Insertion sort Code
- ~ Merge Sort
- ~ Merge Sort Code
- ~ Quick Sort
- ~ Quick Sort Code
- ~ Tea Coffee and Milk problem

=> Trees interview problems and solutions :

- ~ A quick word before problems
- ~ Same tree problem
- ~ Same tree problem Code
- ~ Killer pays road tax problem

- ~ In order iterator
- ~ In order iterator Code
- ~ Flip or Inverse a Binary tree
- ~ Flip or Inverse a Binary tree Code
- ~ Level order of tree
- ~ Level order of tree Code
- ~ Boundary of a tree
- ~ Boundary of a tree Code

=> Graphs interview problems and solutions :

- ~ Basics of graph theory
- ~ Clone a graph or copy
- ~ Clone a graph or copy Code
- ~ DFS and Cycle detection with University course problem
- ~ DFS and Cycle detection with University course problem CODE
- ~ Breadth first search for graphs
- ~ Breadth first search for graphs CODE
- ~ Island problem
- ~ Island problem CODE

=> Dynamic programming interview problems and solutions :

- ~ Foundataion of dynamic programming
- ~ 0Knapsack - Coke 2C pepsi 2C redbull
- ~ 0Knapsack - Coke 2C pepsi 2C redbull CODE
- ~ Largest sum of subset
- ~ Largest sum of subset Code
- ~ Largest sum - Difficult
- ~ Largest sum - Difficult CODE
- ~ Coin change problem
- ~ Coin change problem CODE
- ~ Min path to reach target
- ~ Min path to reach target CODE



# Angular Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : ANGULAR JS

Course link : <https://ineuron.ai/course/Angular-Crash-Course>

## Course Description :-

This course will help you to grab the fundamentals of Angular.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Angular crash course

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Angular crash course :

*~ Angular crash course*

=> NaN :

- ~ NaN*
- ~ NaN*
- ~ NaN*
- ~ NaN*
- ~ NaN*
- ~ NaN*

# Stripe with React and Node Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : WEB DEVELOPMENT PROJECT

Course link : <https://ineuron.ai/course/Stripe-with-React-and-Node-Crash-Course>

## Course Description :-

This course will help you to grab the implementation of Stripe with React and Node.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Stripe with React and Node

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Stripe with React and Node :

*~ Stripe with React and Node*

=> NaN :

- ~ NaN*
- ~ NaN*
- ~ NaN*
- ~ NaN*
- ~ NaN*
- ~ NaN*

# Mega Project Foundation

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/Mega-Project-Foundation>

## Course Description :-

iNeuron is known for conducting end to end community sessions where we discuss everything related to tech along with projects and many more things. The objective of conducting these community sessions is to learn and grow together.

## Course Features :-

=> Free for all

## What you will learn :-

=> Statistics

=> Blockchain

=> Android

=> DSA

=> DevOPS

=> Python

=> PowerBI

=> SQL

=> Projects

=> Drone

=> Robotics

=> AI in edge devices

## Requirements :-

=> Your dedication

=> Laptop

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Statistics :

~ Statistics Day 1 Preview

~ Statistics Day 2 Preview

~ Statistics Day 3

~ Statistics Day 5

~ Statistics Day 6

~ Statistics Day 7

# MySQL

---

Topic Name : DATABASE

Sub-topic Name : MYSQL

Course link : <https://ineuron.ai/course/MySQL>

## Course Description :-

Data practitioners must master SQL since it is the most essential query language you can learn. Many prominent relational database management systems such as MySQL employ it. However, data analysis and big data frameworks and tools such as Apache Spark also utilise it. As a result, learning MySQL offers up a plethora of prospects and occupations - whether you want to work with relational databases or become a data scientist, knowing Mysql is essential. Even if you have no previous experience of MySQL, this practical course will build the groundwork for SQL and structured database querying.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => MySQL tables
- => Primary keys and foreign keys
- => CRUD operations
- => SQL queries
- => Joins
- => ACID in database
- => Database engines

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction and installation of MySQL :

- ~ Introduction to section 1
- ~ MySQL introduction - 5 points to know
- ~ Mysql Installation MAC
- ~ MySQL installation for Windows

=> Basics of MySQL :

- ~ Introduction to section 2
- ~ Creating and dropping database - Startup
- ~ Resolving the issue for future
- ~ Creating your first table
- ~ Adding values to canon table
- ~ Answering customer question

=> Playing with data :

- ~ Introduction to section 3
- ~ Primary key, default and NULL
- ~ Table with primary key and default values
- ~ Testing the new table
- ~ Adding new values and answering questions
- ~ update in customers table
- ~ delete from the customers table

=> More on functions :

- ~ Introduction to section 4
- ~ Understand the new lco user DB
- ~ Task for CONCAT
- ~ Task for REPLACE
- ~ task for SUBSTRING
- ~ Task for reverse and CHAR\_LENGTH
- ~ Task for case conversion and DOCS

=> Answering some DB questions :

- ~ Introduction to section 5
- ~ A task on DISTINCT
- ~ A task for ORDER BY
- ~ A task on LIMIT
- ~ Match the pattern
- ~ A task on COUNT
- ~ SQL MODES and GROUP BY
- ~ MIN MAX and SUBQUERIES
- ~ GROUP BY with MAX and MIN
- ~ SUM and AVERAGE with GROUP BY
- ~ A task on AND OR
- ~ A task in RANGE based selection
- ~ CASE THEN - multiple range selection

=> A pinch of theory :

- ~ Introduction to section 6
- ~ Data type for INTEGER and STRING
- ~ Data type for DATE, DATETIME and JSON
- ~ DATE TIME code Example
- ~ Get the date and time
- ~ Lets join tom and jerry tables
- ~ Types of JOIN

=> FOREIGN KEY and JOINS :

- ~ Introduction to section 7
- ~ Code talk over FOREIGN keys
- ~ Understand a new database
- ~ A task on INNER join
- ~ ONE to MANY and MANY TO MANY
- ~ Join more 3 or more tables
- ~ A task on LEFT JOIN
- ~ A task on RIGHT JOIN
- ~ FULL OUTER join and UNION tasks

=> A pinch of more theory :

- ~ Introduction to section 8
- ~ Database engines - INNODB and more
- ~ ACID in database

=> A 30 Task assignment for movie DB :

- ~ Introduction to section 9
- ~ How to practice database works - FILM

=> Final exam - single attempt :

- ~ MYSQL Outro and some free resources

# Graph ML

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Graph-ML>

## Course Description :-

Graph Machine Learning provides a new set of tools for processing network data and leveraging the power of the relationship between entities that can be used for predictive, modelling, and analytics tasks. This course would cover theory related to GraphML with some implementation.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Embedding Methods
- => Supervised Graph learning
- => Unsupervised Graph learning

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Rishav Dash :

~ This is Rishav Dash. I am a Jr. Data Scientist and mentor at INeuron.ai with working experience in computer vision, natural language processing, Machine Learning, and Alops. Hands-on experience leveraging machine learning, deep learning, transfer learning models to challenging real-world problems, and building products to solve peoples problems.

## Curriculum details :-

=> Intro to Course :

~ Introduction to Graph Machine Learning Preview

=> Graphs :

~ Graph  
~ Graph:- Properties and Representation  
~ Hands on -: Graph:- Properties and Representation Preview

=> Graph Learning :

~ Graph -: Graph Representation Learning  
~ Unsupervised Graph Learning (UGL)  
~ UGL -: Shallow embedding methods  
~ Hands on UGL -: Shallow embedding methods  
~ UGL -: Autoencoders  
~ UGL -: Graph Neural Networks  
~ Supervised Graph Learning (SGL)  
~ SGL -: Feature-based methods  
~ SGL -: Shallow embedding methods  
~ SGL -: Graph regularization methods  
~ SGL -: Graph convolutional neural networks (CNNs)

# Raspberry Pi

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Raspberry-Pi>

## Course Description :-

In this course, you will learn the basics of Raspberry pi and difference between microprocessor and microcontroller and how to use raspberry pi from which you can build amazing IOT applications with Raspberry PI OS. You will also learn Python programming language. This course will introduce the basic of Python library GPIO which will help you to start your journey in the field of artificial intelligence.

## Course Features :-

- => Live instructor led classess
- => Completion certificate
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Get started with Raspberry Pi
- => Understanding Raspberry Pi
- => Understanding of Protocol used in Raspberry Pi
- => Basics of Electronic
- => OS Tour + Linux Fundamentals
- => Understanding Sensors and intergration with Raspberry Pi
- => Raspberry Programming
- => Raspberry Pi beginner projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

- => Get started with Raspberry Pi :
  - ~ Introduction to microcontroller and microprocessor
  - ~ Microcontroller vs Microprocessor
  - ~ Example of microcontroller and microprocessor
  - ~ introduction to raspberry pi
  - ~ Various models of Raspberry Pi
  - ~ Comparison among Raspberry pi, Arduino, Nvidia Jetson Nano, Google coral
  - ~ History of Raspberry Pi
  - ~ Real life use cases for Raspberry Pi
  - ~ Daily Computation
  - ~ Internet of things
  - ~ AI development
  - ~ Purchase Raspberry Pi
  - ~ Ineuron innovation lab (One Nueron)
  - ~ Installtion of Raspberry Pi OS
  - ~ Configure and initiate initial boot of Operating System
  - ~ Get started with programming (C++ and Python)

=> Understanding Raspberry Pi :

- ~ *Raspberry Pi Architecture*
- ~ *Raspberry Pi specification*
- ~ *Raspberry Pi (40 Pin)*
- ~ *Components of Raspberry Pi*

=> Understanding of Protocol used in Raspberry Pi :

- ~ *Introduction to Protocol*
- ~ *UART, SPI, I2C, I2S, Digital I/O, wifi, and bluetooth*

=> Basics of Electronic :

- ~ *Boards*

=> Basics of Electronic :

- ~ *Basic Components (Resistor, Led, Transistor, Capacitor, Diode)*
- ~ *Basic Concepts electricity (Current, Power, voltage etc)*

=> OS Tour + Linux Fundamentals :

- ~ *Desktop Personalization*
- ~ *Working with Terminal*
- ~ *Raspberry Pi Terminal commands*
- ~ *Connecting to a Network*
- ~ *Remote Desktop*

=> Understanding Sensors and integration with Raspberry Pi :

- ~ *Introduction to Sensor*
- ~ *Difference between analog and digital sensors*
- ~ *Sensor list with use case*

=> Raspberry Programming :

- ~ *Supporting Languages*
- ~ *I/O Programming*
- ~ *GPIO configuration*
- ~ *GPIO programming*
- ~ *Interfacing of raspberry pi with various sensors*
- ~ *Interfacing analog and digital sensors with Raspberry Pi*

=> Raspberry Pi beginner projects :

- ~ *Camera Interfacing in Raspberry Pi*
- ~ *Configuration of camera module in Raspberry Pi*
- ~ *Integration of multiple camera*
- ~ *Installation of OpenCV*
- ~ *Real-time video streaming using Camera*



# Pro Aptitude - Operating Systems

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Aptitude---Operating-Systems>

## Course Description :-

This course is designed mostly for computer science subject OPERATING SYSTEM test takers.

## Course Features :-

- => Quizzes
- => Course completion certificate

## What you will learn :-

- => OS Theoretical Test
- => OS Practical Test
- => OS Aptitude Test

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to solve

## Curriculum details :-

- => Operating System Test :
  - ~ Operating System Test 1
  - ~ Operating System Test 2
  - ~ Operating System Test 3
  - ~ Operating System Test 4

# Complete Front End Web Developer Bootcamp

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Complete-Front-End-Web-Developer-Bootcamp>

## Course Description :-

This course will help you to grab the fundamentals of Front End technologies used in Web development and implement them using various projects.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction, getting the dev tools and basics of HTML
- => Div, tables and Forms with Challenge to create FB page
- => HTML 5 semantics and adding audio, video and YouTube to web
- => CSS-Box model, color selection, Google Fonts
- => Web development projects - GYM and Sushi Restro Templates
- => CSS - classes, ID's, parallax and project to edit template
- => CSS - box sizing, gradients and TODO list project
- => Getting started with Bootstrap - Tour and creating landing page
- => Project - Pokemon Corporate site, yahoo selling page and Adm
- => Javascript projects to practice
- => Moving on to learn JQuery - Selectors and event
- => Actions in JQuery - fading, animations and callback function
- => Projects in JQuery and using JS plugins

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction, getting the dev tools and basics of HTML :
  - ~ Course introduction
  - ~ A personal Note
  - ~ Projects that we will build
  - ~ Collecting and installing developers tool
  - ~ Structuring the files and creating first file
  - ~ Text tags
  - ~ List items
  - ~ Ending with Challenge and solution
- => Div, tables and Forms with Challenge to create FB page :
  - ~ Divisions and Spans
  - ~ Images and links
  - ~ Challenge for links on images and solution
  - ~ Tables in HTML
  - ~ More about forms in HTML
  - ~ Challenge to create facebook page and solution
- => HTML 5 semantics and adding audio, video and YouTube to web :
  - ~ Introducing HTML 5

- ~ Comparing HTML 4 semantics with HTML 5
- ~ Adding video 2C audio and youtube videos

=> CSS-Box model, color selection, Google Fonts :

- ~ Introduction to css and where to write it
- ~ Solving the color selection problem
- ~ Comming soon template and backgrounds
- ~ Box model and centering text
- ~ Google fonts and font awesome

=> Web development projects - GYM and Sushi Restro Templates :

- ~ Project - GYM comming soon part 1
- ~ Project - GYM comming soon part 2
- ~ Project - Restro comming soon with video part 1
- ~ Project - Sushi Restro Comming soon part 2

=> CSS - classes, ID's, parallax and project to edit template :

- ~ Styling the links
- ~ Classes and ID in CSS
- ~ Designing a navigation bar from scratch
- ~ Color palletes and canva for design
- ~ Adding parallax part 1
- ~ Adding parallax part 2
- ~ Project - Learn to edit Templates part 1
- ~ Project - Learn to edit templates part 2
- ~ Project - Challenge to edit a Template

=> CSS - box sizing, gradients and TODO list project :

- ~ Gradients in css
- ~ Check through css
- ~ box sizing in css
- ~ Project - Todo list APP part 1
- ~ Project - Todo list APP part 2
- ~ Project - Todo list APP part 3

=> Getting started with Bootstrap - Tour and creating landing page :

- ~ Introduction to Bootstrap
- ~ Creating first Bootstrap file structure
- ~ Overview of bootstrap working
- ~ Understanding the Grid System
- ~ Tour to CSS part of Bootstrap
- ~ Tour to Components part of Bootstrap
- ~ javascript components and Layoutit website builder
- ~ Project - responsive landing Page part 1
- ~ Project - responsive landing Page part 2
- ~ Project - responsive landing Page and media queries part 3
- ~ Project - Adding facebook and twitter logins

=> Project - Pokemon Corporate site, yahoo selling page and Adm :

- ~ Project pokemon Company adding navigations
- ~ Project pokemon - adding slider
- ~ Project pokemon - adding team section
- ~ Project Pokemon - adding content section
- ~ Project Pokemon - footer and animations
- ~ Yahoo Selling page - navbars
- ~ Yahoo Selling page - complete
- ~ project Admin - adding navbar
- ~ project Admin - adding left dashboard
- ~ project Admin - adding top main content
- ~ project Admin - Complete

=> Grabbing the tools to learn and write Javascript :

- ~ important note before we move on to Javascript

=> JavaScript Refresher :

- ~ Welcome to JavaScript Course
- ~ What are JavaScript engines
- ~ What ES version of JavaScript is good for us
- ~ Variable and datatypes in JavaScript
- ~ Our first User Signup
- ~ Operators in JavaScript Calculate discount
- ~ Type and Operator precedence in JavaScript
- ~ What are conditionals in JavaScript
- ~ Logical conditional Login in JavaScript
- ~ Ternary operator in JavaScript
- ~ Switch for role-based access in JavaScript
- ~ Coercion and falsy values in JavaScript
- ~ Basics of functions in JavaScript
- ~ Functions in variable User Role in JavaScript
- ~ Understand the context in JavaScript
- ~ Code hoisting in JavaScript
- ~ Scope chaining in JavaScript
- ~ Light intro to THIS in JavaScript
- ~ Introduction to Array in JavaScript
- ~ Callback and arrow function introduction in array
- ~ Fill and Filter in Array in JavaScript
- ~ Slice and Splice in JavaScript
- ~ Objects in JavaScript
- ~ Methods and objects in JavaScript
- ~ For loop basics in JavaScript
- ~ While and do while loops in JavaScript

- ~ *For Each loop in JavaScript*
- ~ *For in and for of loop in JavaScript*
- ~ *Confusing part of THIS in JavaScript*
- ~ *What is DOM*
- ~ *How to grab web elements in JavaScript*
- ~ *A counter project in JavaScript*
- ~ *Get Computed properties in JavaScript*
- ~ *Event listener in JavaScript*
- ~ *New keyword in JavaScript*
- ~ *What is proto in JavaScript*
- ~ *Better code with object chain in JavaScript*
- ~ *Objects from MDN docs*
- ~ *Self-Executing Anonymous Function and functional programming*
- ~ *Lexical scoping in JavaScript*
- ~ *Closure in JavaScript*
- ~ *Borrow a method using bind*
- ~ *Get to know node Elements in JavaScript*
- ~ *Generating elements and Text node in DOM*
- ~ *Solution of Scope problem in JavaScript*
- ~ *Template literals in JavaScript*
- ~ *Maps in JavaScript*
- ~ *Destructure the data in JavaScript*
- ~ *Spread and REST operators in JavaScript*
- ~ *Classes and module exports in JavaScript*
- ~ *Private props getters and setters in JavaScript*
- ~ *Inheritance in JavaScript*
- ~ *Event loop Will JavaScript wait*
- ~ *Promise async and await in JavaScript*
- ~ *How to Handle API in JavaScript*
- ~ *Get to know game files*
- ~ *Logic of game JavaScript*
- ~ *Fixing the bug in game JavaScript*
- ~ *What is new in JavaScript 2021*
- ~ *Why iife appears in JavaScript interviews*
- ~ *Quirky Behavior of JavaScript*

=> Javascript projects to practice :

- ~ *Project - Random Password Gen*
- ~ *Project - Random Password Gen Styling*
- ~ *Project Ticky Clock styling*
- ~ *Project Ticky Clock Javascript*

=> Moving on to learn JQuery - Selectors and event :

- ~ *Are we on the same page to learn JQuery*
- ~ *Creating Sample exercise files*
- ~ *Your First JQuery code in action*
- ~ *Basics of Selectors in JQuery*
- ~ *Basics of events in JQuery*

=> Actions in JQuery - fading, animations and callback function :

- ~ *Hiding and showing the images*
- ~ *Fading out in jquery*
- ~ *Slides and Animations in jQuery*
- ~ *Callbacks and chaining in jquery*

=> projects in JQuery and using JS plugins :

- ~ *Getting the HTML 2C text and form values*
- ~ *Setting the HTML 2C text and form values*
- ~ *Toggle CSS classes with JQuery*
- ~ *Project - Image Slider part 1*
- ~ *Project - Image Slider part 2*
- ~ *Typeahead and CSS challenge*

=> Farewell and a gift :

- ~ *farewell and final gift*

=> Bonus: TODO project in JS and web hosting details :

- ~ *Bonus - Todo Project and hosting basics*
- ~ *Adding styles to TODO project*
- ~ *javascript for working of todo*
- ~ *Web hosting as quick as possible*

# Mulesoft

---

Topic Name : PROGRAMMING

Sub-topic Name : API

Course link : <https://ineuron.ai/course/Mulesoft>

## Course Description :-

This course will help you to grab the fundamentals of Mulesoft and its application in the industry.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What is Mulesoft
- => API and web services
- => API LED Connectivity
- => Introduction to Design center
- => Exploring Design Center
- => Designing First API using RAML
- => Introduction to Development
- => Downloading anypoint studio
- => Anypoint studio dashboard overview
- => Attributes
- => Configuration properties
- => Flow Reference
- => Variables
- => Importance of variable

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Manjunatha A :

~ Data Scientist with good experience in machine learning, deep learning, and Python programming. I was awarded the Gold medal in my Master's (MCA). I was also privileged to be honoured with the Ace of innovation award. I was also one of the Finalists of SIH -2020 the world's largest hackathon. In my spare time, I enjoy sharing my technological abilities and knowledge through classes. I supervised over 500 students and assisted them in establishing careers in their industries. I also travel frequently.

## Curriculum details :-

=> Introduction to Mulesoft :

- ~ Course curriculum overview
- ~ Who's this course for
- ~ What is Mulesoft
- ~ API and web services
- ~ API LED Connectivity
- ~ Introduction to anypoint studio and mulesoft community
- ~ Sign up and sign in to anypoint Studio
- ~ Practical demonstration of resource and query params

=> Design :

- ~ Introduction to Design center
- ~ Exploring Design Center
- ~ Designing First API using RAML
- ~ Designing API using POST method
- ~ Including both methods in a resource
- ~ Schemas folder creation using type

~ Example separate folder using RAML

## => Developing API's :

- ~ Introduction to Development
- ~ Downloading anypoint studio
- ~ Anypoint studio dashboard overview
- ~ Downloading and installation of Postman
- ~ Introduction to Postman contents
- ~ First Mule Project
- ~ Creating multiple flows
- ~ Importing RAML designing from designer
- ~ Cloud deployment

## => Connectors :

- ~ Attributes
- ~ Configuration properties
- ~ Flow Reference
- ~ Variables
- ~ Importance of variable
- ~ Logger
- ~ Transform message

## => DataWeave :

- ~ Introduction to DataWeave
- ~ Tooling and Preview
- ~ DataWeave basics
- ~ DataWeave passing data from payload and queryparams
- ~ Transforming types
- ~ Introduction to map operator and objects

## => DataWeave Operations and functions :

- ~ Dollar and Double dollar
- ~ Creating new fields using map
- ~ map using item and index
- ~ mapObject
- ~ JSON and XML using map
- ~ NULL check using default
- ~ reduce
- ~ flatten
- ~ pluck
- ~ groupBy
- ~ orderBy
- ~ Filter
- ~ if else
- ~ attributes
- ~ functions
- ~ choice
- ~ Choice -2
- ~ Scatter Gather
- ~ Validation

## => Exception handling :

- ~ Introduction to exception handling
- ~ Default exception handling
- ~ On-error propagate and On-continue error

# System Design Foundations

---

Topic Name : SYSTEM DESIGN

Sub-topic Name : SYSTEM DESIGN

Course link : <https://ineuron.ai/course/System-Design-Foundations>

## Course Description :-

This Course helps you to build the fundamentals of system designs to crack software engineer interviews.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What is System Design?
- => How it will help?
- => Where it is used?
- => Who should be benefited from system design?
- => What are design principles?
- => Different design principles
- => What are design patterns?
- => History of design patterns
- => Types of design patterns
- => Important oops concepts required
- => Abstract classes
- => Static classes
- => OOPS concepts
- => Interface
- => Virtual keyword
- => Override
- => System Design Basics and Key Characteristics of Distributed Systems
- => Load balacing, Caching, Sql vs NoSql

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Anjali Sheel :

~ Currently working for Microsoft as SDE2 in the windows team with more than 5 years of experience in software development. Mentoring students in the Microsoft's engage program. Have more than one year of teaching experience for competitive programming. Have given various webinars for learner community for interview preparation. Have done M.Tech from Delhi Technological university(formerly known as DCE). Have 4 years of experience working at Siemens healthineers, a german product based company.

## Curriculum details :-

=> Day 1 :

- ~ Intro to System Design
- ~ What is System Design?
- ~ How it will help?
- ~ Where it is used?
- ~ Who should be benefited from system design?

=> Day 2 :

- ~ What are design principles?
- ~ Different design principles

=> Day 3 :

- ~ *What are design patterns?*
- ~ *History of design patterns*
- ~ *Types of design patterns*

=> Day 4 :

- ~ *Important oops concepts required*
- ~ *Abstract classes*
- ~ *Static classes*
- ~ *OOPS concepts*
- ~ *Interface*
- ~ *Virtual keyword*
- ~ *Override*

=> Day 5 :

- ~ *System Design Basics and Key Characteristics of Distributed Systems*
- ~ *Load balancing, Caching, Sql vs NoSql*



# Python3 Scratch To Pro

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON

Course link : <https://ineuron.ai/course/Python3-Scratch-To-Pro>

## Course Description :-

This course will help you to grab the fundamentals of Python3 and implement in real life to solve problems.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction and installation of python
- => First interaction with python basics
- => Conditionals and loops
- => Detail analysis of data types
- => Functions Files and Exceptions
- => Python challenges for fun
- => Object Oriented programming in python
- => Database TODO App
- => Advance side of python
- => Handling API in Python

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Introduction and installation of python :

- ~ Introduction to python course
- ~ Python Installation
- ~ Pycharm Installation on Windows
- ~ Installation of python on MAC
- ~ Installing Pycharm in MAC
- ~ Using VSCode for python- optional

=> First interaction with python basics :

- ~ Indents and comments
- ~ take input from user and challenge
- ~ input challenge solution
- ~ getting started with variables in python
- ~ numbers and strings basics in python
- ~ Lists and tuples basics in python
- ~ Dictionary in python

=> Operations in Python :

- ~ Arithmetic and comparison operators in python
- ~ Logical operations in python
- ~ Membership and identity operations in python

=> Conditionals and loops :

- ~ Introduction to conditionals
- ~ Design a rating system in python
- ~ While - Getting started with loops in python
- ~ First step to read documentation

- ~ *For loop in python*
- ~ *Break keyword in python loops*
- ~ *continue and pass keywords in python*

=> Detail analysis of data types :

- ~ *Randomness in python*
- ~ *Using math library in python*
- ~ *String are powerful in python*
- ~ *Detail talk about lists in python*
- ~ *Tuples and dictionary talks in python*

=> Functions Files and Exceptions :

- ~ *getting started with functions in python*
- ~ *Multiple arguments in python*
- ~ *lambda in python*
- ~ *design custom modules in python*
- ~ *Find the day assignment in python*
- ~ *Main method and file handling in python*
- ~ *Exception handling*

=> Python challenges for fun :

- ~ *Prime number and challenges*
- ~ *range of prime numbers*
- ~ *finding factorials*
- ~ *Get matrix input and print it*

=> Object Oriented programming in python :

- ~ *Introduction to class*
- ~ *objects and constructor in python*
- ~ *Getters and setters in python*
- ~ *Inheritance from Samsung to iphone*
- ~ *Method overriding in python*

=> Database TODO App :

- ~ *Read sqlite3 documentation first*
- ~ *Database helper in sqlite3 part 1*
- ~ *database helper file part 2*
- ~ *Debugging and finishing the app*

=> Advance side of python :

- ~ *Iterator and generators in python*
- ~ *Maps and sets in python*
- ~ *All and any functions in python*
- ~ *Collections and deque*

=> Handling API in Python :

- ~ *Requests and JSON handling in python*
- ~ *Get a unique user every time - Project*

# Machine Learning Foundation

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-Foundation>

## Course Description :-

Machine learning is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Learn which Machine Learning model to choose for each type of problem and how to improve your Machine Learning Models. Become a Machine Learning Engineer and get hired.

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Certificate
- => Assignment
- => Quiz

## What you will learn :-

- => Master Machine Learning on Python
- => Make robust Machine Learning models
- => Use Machine Learning for personal purpose
- => Handle advanced techniques like Dimensionality Reduction
- => Classify data using K-Means clustering, Support Vector Machines (SVM), KNN, Decision Trees, Naive Bayes, and PCA
- => Design and evaluate A/B tests using T-Tests and P-Values
- => Data Visualization with Matplotlib and Seaborn
- => Clean your input data to remove outliers

## Requirements :-

- => Computer with Internet connectivity
- => Basic Programming understanding

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Complete Road Map To Be Expert In Python- Follow My Way :
  - ~ Introduction Preview
- => Complete Roadmap To Follow To Prepare Machine Learning With All Videos And Materials
- => Tutorial 1- Anaconda Installation and Python Basics
- => Why Python is the Best Programming Language For Machine Learning?
- => Tutorial 2 - Python List and Boolean Variables
- => Tutorial 3- Python Sets, Dictionaries and Tuples
- => Tutorial 4 - Numpy and Inbuilt Functions Tutorial
- => Tutorial 5- Pandas, Data Frame and Data Series Part-1
- => Tutorial 6- Pandas, Reading CSV files With Various Parameters- Part 2
- => Tutorial 7- Pandas-Reading JSON, Reading HTML, Read PICKLE, Read EXCEL Files- Part 3
- => Tutorial 8- Matplotlib (Simple Visualization Library)
- => Tutorial 9- Seaborn Tutorial- Distplot, Joinplot, Pairplot Part 1
- => Tutorial 10- Seaborn- Countplot(), Violinplot(), Boxplot()- Part2
- => How To Become Expertise in Exploratory Data Analysis
- => Tutorial 11- Exploratory Data Analysis(EDA) of Titanic dataset
- => Tutorial 12- Python Functions, Positional and Keywords Arguments
- => Tutorial 13- Python Lambda Functions
- => Tutorial 15- Map Functions using Python

=> Tutorial 16- Filter Functions In Python

=> Tutorial 17- Python List Comprehension

=> Tutorial 18- Python Advanced String Formatting

=> Tutorial 19- Python Iterables vs Iterators

=> Tutorial 20- How To Import All Important Python Data Science Libraries Using Pyforest

=> Tutorial 21- Python OOPS Tutorial- Classes, Variables, Methods and Objects

=> Advanced Python- Exception Handling Detailed Explanation In Python

=> Advanced Python Series- Custom Exception Handling In Python

=> Advance Python Series- Public Private And Protected Access Modifiers

=> Tutorial 22-Univariate, Bivariate and Multivariate Analysis- Part1 (EDA)-Data Science

=> Tutorial 23-Univariate, Bivariate and Multivariate Analysis- Part2 (EDA)-Data Science

=> Tutorial 24- Histogram in EDA- Data Science

=> Tutorial 24-Z Score Statistics Data Science

=> Tutorial 25- Probability Density function and CDF- EDA-Data Science

=> Tutorial 26- Linear Regression Indepth Maths Intuition- Data Science

=> Tutorial 27- Ridge and Lasso Regression Indepth Intuition- Data Science

=> Tutorial 28- Ridge and Lasso Regression using Python and Sklearn

=> Multiple Linear Regression using python and sklearn

=> Tutorial 28-MultiCollinearity In Linear Regression- Part 2

=> Machine Learning-Bias And Variance In Depth Intuition| Overfitting Underfitting

=> Tutorial 29-R square and Adjusted R square Clearly Explained| Machine Learning

=> Tutorial 31- Hypothesis Test, Type 1 Error, Type 2 Error

=> Tutorial 32- All About P Value,T test,Chi Square Test, Anova Test and When to Use What?

=> Tutorial 33- P Value,T test, Correlation Implementation with Python- Hypothesis Testing

=> Tutorial 33- Chi Square Test Implementation with Python- Hypothesis Testing- Part 2

=> Tutorial 34- Performance Metrics For Classification Problem In Machine Learning- Part1

=> Tutorial 35- Logistic Regression Indepth Intuition- Part 1| Data Science

=> Tutorial 36- Logistic Regression Indepth Intuition- Part 2| Data Science

=> Tutorial 36- Logistic Regression Mutliclass Classification(OneVsRest)- Part 3| Data Science

=> Tutorial 37: Entropy In Decision Tree Intuition

=> Tutorial 38- Decision Tree Information Gain

=> Tutorial 39- Gini Impurity Intuition In Depth In Decision Tree

=> Tutorial 40- Decision Tree Split For Numerical Feature

=> Advance House Price Prediction- Exploratory Data Analysis- Part 1

=> Advance House Price Prediction- Exploratory Data Analysis- Part 2

=> Advance House Price Prediction-Feature Engineering Part 1

=> Advance House Price Prediction-Feature Engineering Part 2

=> Advance House Price Prediction-Feature Selection

=> Tutorial 41-Performance Metrics(ROC,AUC Curve) For Classification Problem In Machine Learning Part 2

=> Performance Metrics On MultiClass Classification Problems

=> K Nearest Neighbor classification with Intuition and practical solution

=> K Nearest Neighbour Easily Explained with Implementation

=> Tutorial 42 - Ensemble: What is Bagging (Bootstrap Aggregation)?

=> Tutorial 43-Random Forest Classifier and Regressor

=> Tutorial 45-Handling imbalanced Dataset using python- Part 1

=> Tutorial 46-Handling imbalanced Dataset using python- Part 2

=> Hyperparameter Optimization for Xgboost

=> What is AdaBoost (BOOSTING TECHNIQUES)

=> Visibility Climate Prediction- You Can Add This In Your Resume

=> Euclidean Distance and Manhattan Distance

=> K Means Clustering Intuition

=> Hierarchical Clustering intuition

=> DBSCAN Clustering Easily Explained with Implementation

=> Silhouette (clustering)- Validating Clustering Models- Unsupervised Machine Learning

=> Curse of Dimensionality Easily explained| Machine Learning

=> Dimensional Reduction| Principal Component Analysis

=> Principle Component Analysis (PCA) using sklearn and python

=> What is Cross Validation and its types?

=> Tutorial 42-How To Find Optimal Threshold For Binary Classification - Data Science

=> Tutorial 47- Bayes' Theorem| Conditional Probability- Machine Learning

=> Tutorial 48- Naive Bayes' Classifier Indepth Intuition- Machine Learning

=> Tutorial 49- How To Apply Naive Bayes' Classifier On Text Data (NLP)- Machine Learning

=> Support Vector Machine (SVM) Basic Intuition- Part 1| Machine Learning

=> Maths Intuition Behind Support Vector Machine Part 2 | Machine Learning Data Science

=> Gradient Boosting In Depth Intuition- Part 1 Machine Learning

=> Gradient Boosting Complete Maths Indepth Intuition Explained| Machine Learning- Part2

=> Xgboost Classification Indepth Maths Intuition- Machine Learning Algorithms

=> Xgboost Regression In-Depth Intuition Explained- Machine Learning Algorithms

=> Data Science In Medical-Live Tracking Of CO--VID Cases In India using Python

=> Perform EDA In Seconds With Visualization Using SweetViz Library

=> 4 End To End Projects Till Deployment For Beginners In Data Science| All You Have To Do Is Learn

=> Deploy Machine Learning Models Using StreamLit Library- Data Science

=> Perform Exploratory Data Analysis In Minutes- Data Science| Machine Learning

=> Pandas Visual Analysis- Perform Exploratory Data Analysis In A Single Line Of Code

=> How To Read And Process Huge Datasets in Seconds Using Vaex Library| Data Science| Machine Learning

=> D-Tale The Best Library To Perform Exploratory Data Analysis Using Single Line Of Code

=> Interview Prep Day3-How To Prepare Support Vector Machines Important Questions In Interviews

=> Google Datasets Search Engine- Search All Datasets From One Place For Data Science,Machine Learning

=> How To Run Flask In Google Colab

=> Time Series Forecasting Using Facebook FbProphet

=> Performance Metrics Interview Questions- Data Science

=> How To Perform Post Pruning In Decision Tree? Prevent Overfitting- Data Science

=> How To Train Machine Learning Model Using CPU Multi Cores

=> Step By Step Process To Learn Machine Learning Algorithm Efficiently

=> Data Science Is Just Not About Model Building

=> How To Interpret The ML Model? Is Your Model Black Box? Lime Library

=> 6 Healthcare End To End Machine Learning Projects- Credits Devansh and Bedanta

=> Overfitting, Underfitting And Data Leakage Explanation With Simple Example

=> What Is API? Application Programming Interface And Why It Is Important-Data Science

=> 500+ Machine Learning And Deep Learning Projects All At One Place

=> Google Colab Pro Vs Colab Free- Benefits Of Using Colab Pro- How To Access From India

# Advanced Web Development

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Advanced-Web-Development>

## Course Description :-

In this course, you will learn how to make interactive web pages using Bootstrap, JavaScript, React, and Node. This course will focus on practical aspects of web page development with a focus on good programming habits. After knowing the fundamentals you can create a website layout according to your imagination with animations and effects to support your beautiful responsive website.

## Course Features :-

=> Live instructor lead classes

=> Completion certificate

## What you will learn :-

=> Web Development

=> Bootstrap

=> Javascript

=> React

=> Node

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

=> Introduction to Advanced Web Development :

- ~ Course Introduction
- ~ Who is this course for?
- ~ Course Overview
- ~ Course Outcome
- ~ Introduction to advanced web development

=> Bootstrap :

- ~ Introduction to Bootstrap
- ~ Bootstrap in Project
- ~ Containers
- ~ Buttons
- ~ Alerts
- ~ Badges
- ~ Button Groups
- ~ Cards in Bootstrap
- ~ Grids
- ~ Advance Column Properties
- ~ Image Slider
- ~ Dropdowns
- ~ Modal in Bootstrap
- ~ OffCanvas
- ~ Popovers
- ~ Spinners
- ~ Toast
- ~ Accordion
- ~ Bootstrap Navs
- ~ NavBars in Bootstrap
- ~ Forms
- ~ Helper Classes
- ~ Utilities Classes
- ~ Flex
- ~ Interactions
- ~ Utilities Properties
- ~ TypoGraphy in Bootstrap
- ~ Handling in Images & Tables
- ~ Building a Bootstrap Website

## => Assignment1 :

~ Design your own Bootstrap website.

## => Javascript :

- ~ Introduction to Javascript
- ~ Running Javascript in Browser
- ~ Console
- ~ Strings & Numbers
- ~ var, let & const
- ~ Data Types
- ~ Type Conversions
- ~ Arithmetic Operators
- ~ Assignment Operator
- ~ Comparison Operator
- ~ Logical Not, Or and And
- ~ Swap Numbers
- ~ String Handling
- ~ String Searching
- ~ Arrays
- ~ Objects
- ~ Dates
- ~ Maths
- ~ If & Else
- ~ Challenge - If & Else
- ~ Switch Case
- ~ Challenge - Switch Case
- ~ JS Loops
- ~ For Loops
- ~ Nested Loops
- ~ Break & Continue
- ~ Arrays, Strings & Objects
- ~ For-in
- ~ For-of
- ~ While Loops
- ~ Do while Loops
- ~ Loops Exercises
- ~ Functions
- ~ Variable Scopes in Functions
- ~ Nested Functions
- ~ Parameters & Arguments
- ~ How function is useful?
- ~ Return in Function
- ~ Anonymous Functions
- ~ Calculator Exercise
- ~ Arrow Functions
- ~ forEach
- ~ maps
- ~ String Literals
- ~ Filter, Reduce & Every
- ~ Spread Operator
- ~ Window & Document
- ~ Document Access
- ~ innerText & innerHTML
- ~ HTML Calculator
- ~ Query Selector
- ~ Styling in JS
- ~ Advance DOM Manipulation
- ~ Events
- ~ Basic Events
- ~ Time Events
- ~ Pop-up Boxes
- ~ Error Handling
- ~ Form Validation

## => Assignment2 :

~ Implement Stack data structure operations with Javascript

## => React :

- ~ Introduction
- ~ Folder Structure
- ~ JSX
- ~ Expressions and Literals
- ~ CSS in React
- ~ Nested Components
- ~ Greeting App
- ~ Props
- ~ Conditional Rendering
- ~ useState
- ~ Arrays & Objects in useState
- ~ Forms
- ~ ToDo app with useState
- ~ useEffect
- ~ useRef
- ~ React Router Dom
- ~ Context
- ~ fetch API
- ~ Axios
- ~ React-Hooks-Forms
- ~ Memo

- ~ *Callback*
- ~ *Sockets*
- ~ *Charts with Sockets*
- ~ *Custom Hooks*
- ~ *React Redux Introduction*
- ~ *Redux*
- ~ *Redux-thunk*
- ~ *Ecommerce with Redux*
- ~ *React Bootstrap Introduction*
- ~ *React Bootstrap*
- ~ *Material-ui Introduction*
- ~ *Material-ui Buttons*
- ~ *Material-ui Slider*
- ~ *Material-ui Typography*
- ~ *Material-ui Forms*
- ~ *Material-ui Grids*
- ~ *Material-ui Cards*

=> Assignment3 :

- ~ *Create youtube with react*

=> Nodejs :

- ~ *Introduction & Installation*
- ~ *Global Objects*
- ~ *Modules*
- ~ *OS Module*
- ~ *Path Module*
- ~ *Fs Module*
- ~ *Advance Fs*
- ~ *NPM*
- ~ *Http Server*
- ~ *Events*
- ~ *Streams*
- ~ *Express*
- ~ *Serving Files*
- ~ *Router*
- ~ *Post ,Query & Parameters*
- ~ *Adding routes & Validation*
- ~ *Middlewares*
- ~ *Controllers*
- ~ *Serving FTP & Compression*
- ~ *Async Express Route*
- ~ *Save API's from DDoS Attack*
- ~ *Uploading & Downloading*
- ~ *Nodemailer*
- ~ *Error Handling*
- ~ *Embedded Javascript Templates*

=> Assignment4 :

- ~ *Create youtube Api*

=> Course Summary :

- ~ *Course Outro*
- ~ *Future learning path*



# Spring Boot

---

Topic Name : PROGRAMMING

Sub-topic Name : SPRING BOOT

Course link : <https://ineuron.ai/course/Spring-Boot>

## Course Description :-

Spring Boot is an opinionated, simple-to-use extension to the Spring platform that is extremely beneficial for quickly developing stand-alone, production-ready apps. Spring Boot is a strong framework for quickly developing web apps with minimal code. The course will teach you how to use Spring Boot to create a variety of projects.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Annotations
- => IOC /DI
- => Constructor
- => Getter Setter
- => Spring Autowiring Part-1
- => Spring Autowiring Part-2
- => Spring JDBC Template

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Spring Boot :
  - ~ Introduction & Installation
  - ~ Annotations
  - ~ IOC /DI
  - ~ Constructor
  - ~ Getter Setter
  - ~ Spring Autowiring Part-1
  - ~ Spring Autowiring Part-2
  - ~ Spring JDBC Template

# Azure Luis

---

Topic Name : DATA SCIENCE

Sub-topic Name : CHATBOT

Course link : <https://ineuron.ai/course/Azure-Luis>

## Course Description :-

With a couple of coding sessions, this course is aimed to be more practical than theoretical. You'll have all the tools you need to build a fully working chatbot and integrate it with other platforms like Facebook and Skype by the conclusion of the course.

## Course Features :-

- => Source code
- => Challenges
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Get knowledge of various concepts for building chatbots using Bot Builder SDK and LUIS
- => Chatbot integration with Facebook Messenger, Skype, & Slack
- => Different cloud platforms like Heroku, AWS, Azure chatbot app deployment
- => LUIS: intents, entities, production, model training

## Requirements :-

- => No prior experience in any of the chatbots
- => Minimal knowledge of python language
- => Slack, facebook, telegram accounts
- => Interest to learn
- => A system with internet connection
- => Your dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Introduction :

- ~ Course introduction Preview
- ~ What is Chatbot?
- ~ Why Chatbot?

=> LUIS :

- ~ What is LUIS? Preview
- ~ Create account
- ~ Intent & utterances
- ~ Prebuilt domain intent
- ~ Using entities
- ~ Entity types
- ~ Utterances
- ~ Pattern
- ~ Machine learning features
- ~ Prediction score
- ~ Data management
- ~ LUIS and QnA maker
- ~ CI/CD with Luis

=> Build LUIS App :

- ~ Overview
- ~ Azure portal setup
- ~ Intent/entity
- ~ GetWeather API
- ~ Luis App credentials
- ~ Installation
- ~ Code walkthrough
- ~ Bot emulator
- ~ Test Chatbot

=> Course summary :  
~ *Summary*

# The Pro Backend Developer

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/The-Pro-Backend-Developer>

## Course Description :-

This course is titled pro for a reason. In this practical hands-on course, you will learn how to build complex backend applications that can be used for any web or mobile application. Your REST API will be in production with docs, social logins, images, authentications, mail and, much more. This is a true pro backend course.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => MongoDB
- => Heroku Cloud
- => Swagger
- => Authentication
- => File, image and form handling
- => MORGAN and razorpay
- => Configs and imports
- => Controllers and routes

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started :

- ~ Goal of this course and instructions
- ~ Tools for backend developer
- ~ MongoDB MAC install
- ~ MongoDB WIN install
- ~ MongoDB in cloud - Atlas
- ~ Mongo GUI - compass

=> Take it up to Heroku - Production :

- ~ Things you need to deploy on Heroku
- ~ Plan your application
- ~ Types of web request
- ~ Framework - Express, Koa, Hapi
- ~ Starting with package JSON file
- ~ Your first express app
- ~ Request Response and Status code
- ~ All social routes
- ~ Handle the date situation
- ~ Parameters and bugs in route
- ~ Pushing app to HEROKU
- ~ Debug social app in production

=> Swagger Docs :

- ~ What is swagger and api docs
- ~ Nodemon ext and YAML docs

- ~ Authentication token for swagger docs
- ~ Docs for HTTP methods swagger
- ~ A new documentation centric project
- ~ Setup information - swagger
- ~ Authentication and Authorization - swagger
- ~ String based GET request - swagger
- ~ handling objects - swagger
- ~ handling array in Swagger docs
- ~ Sending data in URL - swagger
- ~ managing request body in swagger
- ~ handle url query in swagger
- ~ handling images in swagger
- ~ handling header tokens in swagger

=> Authentication :

- ~ What we have done till section 3 - backend
- ~ Hiding secrets in backend
- ~ Picking up a database for backend
- ~ Why we need mongoose - ODM
- ~ Pro db modeling tools
- ~ Creating model for auth system
- ~ Creating basic structure for auth system
- ~ Creating user schema and dotenv
- ~ Registering a user in auth system
- ~ Database connection in auth system
- ~ What is a middleware
- ~ Handling password situation
- ~ What is JWT and creating token
- ~ Register route in auth app
- ~ Login flow for auth app
- ~ Web vs Mobile
- ~ Writing custom middleware
- ~ Setting up secure cookies

=> File, image and form handling :

- ~ Why people face issue in image upload
- ~ Cloudinary and EJS
- ~ How GET works and postman issues
- ~ Using template engines
- ~ Biggest confusion in front end forms
- ~ Handling images in forms
- ~ Handling images in forms part 2
- ~ upload image to cloudinary or other providers
- ~ Handling multiple files and uploading them

=> Theory and Razorpay :

- ~ File structure for production app
- ~ Getting a logger - MORGAN
- ~ Error handler and Promises
- ~ Sending emails using nodemailer
- ~ Why mongoose docs are important
- ~ Razorpay project
- ~ Razorpay front end integration

=> Big Ecommerce app starts :

- ~ Project requirement
- ~ User modeling and file structure
- ~ Product model discussion
- ~ Order Model discussion
- ~ How forgot password feature work
- ~ Functions in user model and hooks

=> Basic Config and imports :

- ~ Getting files and folders ready
- ~ Preparing basic express app
- ~ Routes and controllers in dummy
- ~ Injecting docs and middleware
- ~ Custom error handlers
- ~ The big Promise

=> User model and signup :

- ~ Creating a user model and validator
- ~ password encryption and mongoose prototypes
- ~ Validating the password
- ~ creating JWT tokens
- ~ forgot password and crypto hashing
- ~ User routes and postman
- ~ Signup a user and cookies
- ~ Database connection
- ~ Testing the user signup with postman
- ~ Handling image upload
- ~ Testing photo upload and user signup
- ~ yes, we know about postman files

=> User controllers and routes :

- ~ Login route and controller
- ~ logout controller and route
- ~ Send email from node
- ~ Forgot password controller
- ~ Reset password controller and routes
- ~ Middleware - injecting information

- ~ User dashboard controller and routes
- ~ Update the password for a user
- ~ Updating the user profile
- ~ User, admin, manager and more roles
- ~ Manager only routes
- ~ Admin get a single user
- ~ Admin can update any user
- ~ Admin can delete a user now

#### => Working on Product Model :

- ~ Product middleware setup for routes
- ~ Product Model and refs
- ~ A long talk on URL replace and mongo operators
- ~ Creating a product
- ~ Where clause in search
- ~ Where clause Pager
- ~ Aggregation filter in Where Clause
- ~ Get all products with WHERE and pager
- ~ Debugging and testing of product add and get

#### => More routes in Products :

- ~ Single product route
- ~ Update the product with photos
- ~ Delete a product and minor bug
- ~ Testing and debugging
- ~ Add a review
- ~ Delete a review and requested routes
- ~ Configure routes for reviews

#### => Razorpay and Stripe :

- ~ Stripe Docs
- ~ Stripe controllers
- ~ Razorpay payments and order
- ~ Setup payment routes

#### => Processing Orders :

- ~ Order model in action
- ~ Creating a order and BSON
- ~ Testing create order and routes
- ~ Populate fields in order
- ~ Order of routes is important
- ~ Updating the stock
- ~ Delete order and push to git
- ~ Pushing code to production server

#### => OAuth and Social Logins :

- ~ Social login foundation and demo app
- ~ Consent screen and API keys
- ~ Why passport.js
- ~ Package installation
- ~ Home routes and EJS
- ~ Preparing routes for login
- ~ Showing consent screen of google
- ~ Getting information and email from google
- ~ Moving google data to database
- ~ Serialize and deserialize user
- ~ Protect the Home

# Streamlit

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Streamlit>

## Course Description :-

In this course, you will learn about the fundamental concepts related to the Streamlit library. After completion of this course, you will be able to create Web applications for your Data science models with the help of Streamlit Library.

## Course Features :-

- => Practical Implementation
- => Downloadable resources
- => Class Recordings
- => Quiz Questions
- => Completion Certificate

## What you will learn :-

- => Streamlit
- => Web-apps
- => Building UI

## Requirements :-

- => Prior Knowledge of Python Programming Language
- => Interest to learn
- => Your dedication

## Instructors :-

=> Jaydeep Dixit :

~ Jaydeep Dixit is a data scientist and Blockchain Developer working at iNeuron having 1.5+ years of total experience. He specializes in Machine Learning and Blockchain. He has worked on various end-to-end projects in both machine learning and Blockchain. In addition to his primary job function, he has been recognized for his problem-solving skills.

## Curriculum details :-

=> Course Introduction :

- ~ Introduction to Streamlit Preview
- ~ Who is this course for?
- ~ Course overview Preview

=> Installation :

- ~ Installation of Streamlit

=> Main Concepts :

- ~ Basic concepts
- ~ apiReference Preview
- ~ Data display elements
- ~ Chart elements
- ~ Input widgets
- ~ Media elements
- ~ Layout container
- ~ Status elements
- ~ Control flow

=> Project :

- ~ Banknote Authentication

=> Course Summary :

- ~ Summary
- ~ Future learning

# JavaScript Fundamentals

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : JAVASCRIPT

Course link : <https://ineuron.ai/course/JavaScript-Fundamentals>

## Course Description :-

This course will help you to grab the fundamentals of JavaScript for web development.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Basics of JavaScript
- => What are JavaScript engines
- => Objects in JavaScript
- => Methods and objects in JavaScript
- => What is DOM?
- => New keyword in JavaScript
- => What is proto in JavaScript
- => How to Handle API in JavaScript

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

- => JavaScript Refresher :
  - ~ Welcome to JavaScript Course
  - ~ What are JavaScript engines
  - ~ What ES version of JavaScript is good for us
  - ~ Variable and datatypes in JavaScript
  - ~ Our first User Signup
  - ~ Operators in JavaScript Calculate discount
  - ~ Type and Operator precedence in JavaScript
  - ~ What are conditionals in JavaScript
  - ~ Logical conditional Login in JavaScript
  - ~ Ternary operator in JavaScript
  - ~ Switch for role-based access in JavaScript
  - ~ Coercion and falsy values in JavaScript
  - ~ Basics of functions in JavaScript
  - ~ Functions in variable User Role in JavaScript
  - ~ Understand the context in JavaScript
  - ~ Code hoisting in JavaScript
  - ~ Scope chaining in JavaScript
  - ~ Light intro to THIS in JavaScript
  - ~ Introduction to Array in JavaScript
  - ~ Callback and arrow function introduction in array
  - ~ Fill and Filter in Array in JavaScript
  - ~ Slice and Splice in JavaScript
  - ~ Objects in JavaScript
  - ~ Methods and objects in JavaScript
  - ~ For loop basics in JavaScript
  - ~ While and do while loops in JavaScript
  - ~ For Each loop in JavaScript
  - ~ For in and for of loop in JavaScript



- ~ Confusing part of *THIS* in JavaScript
- ~ What is DOM
- ~ How to grab web elements in JavaScript
- ~ A counter project in JavaScript
- ~ Get Computed properties in JavaScript
- ~ Event listener in JavaScript
- ~ New keyword in JavaScript
- ~ What is proto in JavaScript
- ~ Better code with object chain in JavaScript
- ~ Objects from MDN docs
- ~ Self-Executing Anonymous Function and functional programming
- ~ Lexical scoping in JavaScript
- ~ Closure in JavaScript
- ~ Borrow a method using bind
- ~ Get to know node Elements in JavaScript
- ~ Generating elements and Text node in DOM
- ~ Solution of Scope problem in JavaScript
- ~ Template literals in JavaScript
- ~ Maps in JavaScript
- ~ Destructure the data in JavaScript
- ~ Spread and REST operators in JavaScript
- ~ Classes and module exports in JavaScript
- ~ Private props getters and setters in JavaScript
- ~ Inheritance in JavaScript
- ~ Event loop Will JavaScript wait
- ~ Promise async and await in JavaScript
- ~ How to Handle API in JavaScript
- ~ Get to know game files
- ~ Logic of game JavaScript
- ~ Fixing the bug in game JavaScript
- ~ What is new in JavaScript 2021
- ~ Why iife appears in JavaScript interviews
- ~ Quirky Behavior of JavaScript

# Qlik Sense

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Qlik-Sense>

## Course Description :-

This program is specialized in providing theoretical as well as practical understanding of the Qlik Sense analytical platform which allows you to learn how to build analytical apps. Course curriculum includes concepts about Qlik sense features, different charts, hands-on and much more!

## Course Features :-

- => Learning of different concepts of Qlik sense analytical
- => Building Dashboards
- => Hands-on Project
- => Assignments
- => Practical Implementation
- => Downloadable Resources
- => Completion Certificate

## What you will learn :-

- => Qlik sense features
- => Basic concepts
- => Installation
- => Tutorial
- => Build Charts
- => Building Dashboards
- => Project

## Requirements :-

- => No prior knowledge in Analytics
- => System with Internet Connection
- => Interest to learn
- => Basic knowledge of BI
- => Dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Course Introduction :

- ~ Syllabus overview Preview
- ~ Introduction to Qlik Sense Preview
- ~ Qlik Sense features

=> Dashboard :

- ~ Overview
- ~ Qlik APP
- ~ Charts

=> Practical Implementation :

- ~ Load data
- ~ Creating charts
- ~ Scatter plot

# Golang

---

Topic Name : PROGRAMMING

Sub-topic Name : GOLANG

Course link : <https://ineuron.ai/course/Golang>

## Course Description :-

The following concepts will be covered in this course training: golang basics, features, environment setup, program structure, syntaxes, data types, type casting, operators, array, recursion, interfaces, error handling, and packages, among others.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Golang installation and hello world
- => GOPATH and reading go docs
- => Build for windows, linux and mac
- => Memory management in golang
- => Pointers in golang
- => Array in golang
- => Slices in golang
- => Functions in golang
- => Methods in golang
- => Defer in golang
- => Working with files in golang

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Golang :
  - ~ Welcome to series on GO programming language
  - ~ Before you start with golang
  - ~ Golang installation and hello world
  - ~ GOPATH and reading go docs
  - ~ Lexer in golang and Types
  - ~ Variables, types and constants
  - ~ Comma ok syntax and packages in golang
  - ~ Conversions in golang
  - ~ Handling time in golang
  - ~ Build for windows, linux and mac
  - ~ Memory management in golang
  - ~ Pointers in golang
  - ~ Array in golang
  - ~ Slices in golang
  - ~ How to remove a value from slice based on index in golang
  - ~ Maps in golang
  - ~ Structs in golang
  - ~ If else in golang
  - ~ Switch case in golang and online playground
  - ~ Loop break continue and goto in golang
  - ~ Functions in golang

- ~ *Methods in golang*
- ~ *Defer in golang*
- ~ *Working with files in golang*
- ~ *Handling web request in golang*
- ~ *Handling URL in golang*
- ~ *Creating server for golang frontend*
- ~ *How to make GET request in golang*
- ~ *How to make POST request with JSON data in golang*
- ~ *How to send form data in golang*
- ~ *How to create JSON data in golang*
- ~ *How to consume JSON data in golang*
- ~ *A long video on MOD in golang*
- ~ *Building API in golang - Models*
- ~ *Sending a API json response for all courses in golang*
- ~ *Get one course based on request id in golang*
- ~ *Add a course controller in golang*
- ~ *Update a course controller in golang*
- ~ *Delete a course controller in golang*
- ~ *Handling routes and testing routes in golang*
- ~ *MongoDB setup for API in golang*
- ~ *Defining models for netflix in golang*
- ~ *Making a connection to database in golang*
- ~ *Insert data in mongodb in golang*
- ~ *Update a record in mongodb in golang*
- ~ *Delete one and delete many in mongodb in golang*
- ~ *Get all collection in mongodb in golang*
- ~ *Get all movies from DB in golang*
- ~ *Mark movie as watched in golang*
- ~ *Delete 1 and all movie in golang*
- ~ *Creating routes and testing API in golang*
- ~ *Concurrency and goroutines in golang*
- ~ *Wait groups in golang*
- ~ *Mutex in golang*
- ~ *Race Condition in golang*
- ~ *Channels and Deadlock in golang*
- ~ *Math, crypto and random number in golang*

# Aptitude Test Series

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Aptitude-Test-Series>

## Course Description :-

This course is designed mostly for Aptitude test takers.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Aptitude
- => Time, Distance & Speed
- => Time & Work
- => Simple & Compound Interest
- => Profit, Loss & Discount
- => Problems on Ages
- => Permutation & Combination
- => Percentages
- => Partnership
- => Numbers, HCF & LCM
- => Numbers & Letter Series
- => Logical Deduction
- => Equation & Ratios
- => Direction Sense
- => Coding & Decoding
- => Clocks
- => Calendars
- => Blood Relations
- => Averages, Mixtures & Allegation
- => Analytical Reasoning

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Curriculum details :-

- => Time, Distance & Speed :
  - ~ Time, Distance & Speed Problems
- => Time & Work :
  - ~ Time & Work Problems
- => Simple & Compound Interest :
  - ~ Simple & Compound Interest Problems
- => Profit, Loss & Discount :
  - ~ Profit, Loss & Discount Problems
- => Problems on Ages :
  - ~ Problems on Ages Problems

=> Permutuation & Combination :  
~ *Permutuation & Combination Problems*

=> Percentages :  
~ *Percentages Problems*

=> Partnership :  
~ *Partnership Problems*

=> Numbers, HCF & LCM :  
~ *Numbers, HCF & LCM Problems*

=> Numbers & Letter Sries :  
~ *Numbers & Letter Series Problems*

=> Logical Deduction :  
~ *Logical Deduction Problems*

=> Equation & Ratios :  
~ *Equation & Ratios Problems*

=> Direction Sense :  
~ *Direction Sense Problems*

=> Coding & Decoding :  
~ *Coding & Decoding Problems*

=> Clocks :  
~ *Clocks Problems*

=> Calendars :  
~ *Calendars Problems*

=> Blood Relations :  
~ *Blood Relations*

=> Averages, Mixtures & Allegation :  
~ *Averages, Mixtures & Allegation*

=> Analytical Reasoning :  
~ *Analytical Reasoning Problems*

=> NaN :  
~ *NaN*

# Alteryx Course

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Alteryx-Course>

## Course Description :-

Data preparation, blending, sophisticated analytics, and sharing of findings are becoming more popular in Self-Service Data Analytics.

The worldwide operational analytics market is predicted to expand at an exponential scale by 2021, according to Research and Markets. They want data they can act on within a matter of hours, not weeks. Preparing, blending and analyzing data using a repeatable procedure with Alteryx Analytics is easy for analysts.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => ALTERYX - Directory Tool
- => ALTERYX - Browse Tool
- => ALTERYX - TABLE TOOL
- => ALTERYX - CREATE SAMPLE TOOL

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Pawan Lalwani :

~ Pawan is a highly skilled and self motivated trainer who has expertise in various business intelligence tools like Power BI, Tableau and Microsoft Excel. He comes with 10 years of experience in training individuals in different industry sectors like Banking, Finance, Healthcare, IT, Automobile, Manufacturing and Pharmaceutical.

## Curriculum details :-

=> Introduction :

- ~ Introduction to Alteryx
- ~ Download and Install Alteryx
- ~ User Interface of Alteryx

=> IN/Out Tab :

- ~ ALTERYX - Get Data from Excel
- ~ ALTERYX - Get Data from CSV
- ~ ALTERYX - Append All CSV files
- ~ ALTERYX - Browse Tool
- ~ ALTERYX - Output Tool - Update Existing Data
- ~ ALTERYX - Directory Tool
- ~ ALTERYX - Directory Tool - Specific Files
- ~ ALTERYX - Text Input Tool
- ~ ALTERYX - Date and Time Tool

=> Preparation Tab :

- ~ ALTERYX - Auto Field Tool
- ~ ALTERYX - Data Cleansing Tool Part 1
- ~ ALTERYX - Data Cleansing Tool - Part 2
- ~ ALTERYX - Filter Tool (Text Example)
- ~ ALTERYX - Filter Tool (Number Example)
- ~ ALTERYX - Filter Tool (Date Example)
- ~ ALTERYX - FORMULA TOOL (Basic Example)
- ~ ALTERYX - FORMULA TOOL - (Multiple Examples)
- ~ ALTERYX - GENERATE ROWS TOOL
- ~ ALTERYX - IMPUTATION TOOL
- ~ ALTERYX - MULTI-FIELD BINNING TOOL
- ~ ALTERYX - MULTI-FIELD FORMULA
- ~ ALTERYX - MULTI ROW FORMULA

- ~ ALTERYX - RANDOM % SAMPLE TOOL
- ~ ALTERYX - SAMPLE TOOL
- ~ ALTERYX - RECORD ID TOOL
- ~ ALTERYX - SELECT TOOL
- ~ ALTERYX - SORT
- ~ ALTERYX - CREATE SAMPLE TOOL
- ~ ALTERYX - TILE TOOL
- ~ ALTERYX - UNIQUE TOOL

=> Join Tab :

- ~ ALTERYX - APPEND FIELDS TOOL
- ~ ALTERYX - FIND AND REPLACE TOOL
- ~ ALTERYX - FUZZY MATCH TOOL
- ~ ALTERYX - JOIN TOOL
- ~ ALTERYX - JOIN MULTIPLE TOOL
- ~ ALTERYX - UNION TOOL
- ~ REGEX TOOL
- ~ Text To Columns

=> Transform Tab :

- ~ ALTERYX - CROSS TAB Tool
- ~ ALTERYX - TRANSPOSE Tool
- ~ ALTERYX - RUNNING TOTAL Tool
- ~ ALTERYX - SUMMARIZE TOOL

=> Reporting Tab :

- ~ ALTERYX - TABLE TOOL
- ~ ALTERYX - INTERACTIVE CHART Tool
- ~ ALTERYX - JOIN TABLE AND CHART
- ~ ALTERYX - ADD ANNOTATION
- ~ ALTERYX - REPORT TEXT TOOL
- ~ ALTERYX - REPORT HEADER TOOL
- ~ ALTERYX - REPORT FOOTER TOOL
- ~ ALTERYX - REPORT LAYOUT TOOL

=> Documentation Tab :

- ~ ALTERYX - COMMENT TOOL
- ~ ALTERYX - EXPLORER TOOL
- ~ ALTERYX - CONTAINER TOOL

=> Case Studies :

- ~ Study 1
- ~ Study 2
- ~ Study 3
- ~ Study 4
- ~ Study 5



# Class 7th Math

---

Topic Name : K12

Sub-topic Name : CLASS7

Course link : <https://ineuron.ai/course/Class-7th-Math>

## Course Description :-

This course is useful for Grade 7 students. In this course, entire NCERT will be covered, Various questions from NCERT, NCERT exemplar and previous year will also be discussed. Dedicated doubt clearing sessions will also be conducted for helping the students regularly throughout their learning journey.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Algebra
- => Geometry
- => Statistics
- => Numbers
- => Mensuration

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

=> Integers :

- ~ Lecture 1 : Introduction Preview
- ~ Lecture 3 : NCERT Solutions Ex1.1 Question 2,3,4,5
- ~ Lecture 4 : Part 1 NCERT Solutions Ex1.1 Question 5,6,7,8,9,10
- ~ Lecture 5 : Integer Properties
- ~ Lecture 6 : NCERT Solutions Ex 1.2
- ~ Lecture 7 : Multiplication Property
- ~ Lecture 8 : NCERT Solutions Ex1.3 Question 1,2,3
- ~ Lecture 9 : NCERT Solutions Ex1.3 Question 3,4,5,6
- ~ Lecture 10 : NCERT Solutions Ex1.3 Question 7,8,9
- ~ Lecture 12 : NCERT Solutions Ex1.4
- ~ Lecture 13 : NCERT Solutions Ex1.4 Question 5,6,7

=> Fractions & Decimals :

- ~ Lecture 1 : Introduction NCERT Solutions Ex2.1 Question 1 & 2 Preview
- ~ Lecture 2 : NCERT Solutions Ex2.1 Question 3,4,5,6,7,8 Preview
- ~ Lecture 3 : Introduction Multiplication of Fractions
- ~ Lecture 4 : NCERT Solutions Ex2.2
- ~ Lecture 5 : Introduction NCERT Solutions Ex2.3
- ~ Lecture 6 : NCERT Solutions Ex2.3
- ~ Lecture 7 : Introduction NCERT Solutions Ex2.5
- ~ Lecture 8 : NCERT Solutions Ex2.4
- ~ Lecture 10 : NCERT Solutions Ex2.5 Question 1,2
- ~ Lecture 11 : NCERT Solutions Ex2.5 Question 4,5,6,7,8,9
- ~ Lecture 13 : NCERT Solutions Ex2.6
- ~ Lecture 15 : NCERT Solutions Ex2.7

=> Data Handling :

- ~ Lecture 1 : Data Handling Introduction
- ~ Lecture 2 : NCERT Solutions Ex3.1 Question 1,2,3,4,5,6,7,8,9
- ~ Lecture 3 : NCERT Solutions Ex3.2
- ~ Lecture 4 : NCERT Solutions Ex3.3 Question 1,2
- ~ Lecture 6 : NCERT Solutions Ex3.3 Question 4
- ~ Lecture 9 : NCERT Solutions Ex3.4

=> Simple Equations :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : NCERT Solutions Ex4.1
- ~ Lecture 3 : Introduction Solving Equations
- ~ Lecture 4 : NCERT Solutions Ex 4.2

- ~ Lecture 5 : NCERT Solutions Ex4.3
- ~ Lecture 6 : Introduction NCERT Solutions Ex4.4
- ~ Lecture 7 : NCERT Solutions Ex 4.4

#### => Lines and Angles :

- ~ Lecture 1 : Some Important Definitions : Line segment, Line, Angles, Acute Angle, Obtuse Angle, Reflex Angle, Straight Angle, Complementary Angle, Supplementary Angle
- ~ Lecture 2 : Adjacent Angle, Linear Pair, Vertically Opposite Angle
- ~ Lecture 3 : Ex 5.1 Q 1 to 9
- ~ Lecture 4 : Ex 5.1 Q 9 to 14
- ~ Lecture 5 : Intersecting Lines, Transversal and Angles made by a Transversal, Transversal of Parallel Lines.
- ~ Lecture 6 : Checking for Parallel Lines, Ex 5.2

#### => The Triangle and its Properties :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Elements of Triangle, Vertex, Sides and Angles
- ~ Lecture 3 : Classification of Triangles on the basis of Sides and on the basis of Angles, Angle Sum Property of a Triangle
- ~ Lecture 4 : Median and Altitudes of a Triangle, Ex 6.1
- ~ Lecture 5 : Exterior Angle of a Triangle and its Properties, Ex 6.2
- ~ Lecture 6 : Ex 6.3
- ~ Lecture 7 : Two Special Triangles, Equilateral and Isosceles
- ~ Lecture 8 : Sum of Lengths of Two Sides of a Triangle
- ~ Lecture 9 : Ex 6.4, Q 1 to 5
- ~ Lecture 10 : Right-Angled Triangles And Pythagoras Property
- ~ Lecture 11 : EXERCISE 6.5 Q 3 to 8

#### => Congruence of Triangles :

- ~ Lecture 1 : Congruent Figures, Congruence of Triangles
- ~ Lecture 2 : Ex 7.1 Q 1 to 4
- ~ Lecture 3 : Criteria for Congruence of Triangles, SSS Criteria
- ~ Lecture 4 : SAS, ASA Congruence Condition
- ~ Lecture 5 : Congruence Among Right Angle Triangle (RHS Congruence Condition)
- ~ Lecture 6 : Ex 7.2 Q 1 to 5
- ~ Lecture 7 : Ex 7.2 Q 5 to 6

#### => Comparing Quantities :

- ~ Lecture 1 : Introduction, Understanding Ratio, Equivalent Ratio
- ~ Lecture 2 : Unitary Method, Ex 8.1, Proportion
- ~ Lecture 3 : Percentage, another way of Comparing Quantities
- ~ Lecture 4 : Converting Decimal to Percentage
- ~ Lecture 5 : NCERT Ex 8.2 Q 6 to 8
- ~ Lecture 6 : Profit and Loss
- ~ Lecture 7 : Question Practice on Profit and Loss
- ~ Lecture 8 : Charge Given on Borrowed money / Simple Interest
- ~ Lecture 9 : EX 8.3 Q 1 to 7
- ~ Lecture 10 : EX 8.3 Q 8 to 11

#### => Rational Numbers :

- ~ Lecture 1 : Introduction, Positive and Negative Rational number, Three Important Properties of Rational Numbers, Equivalent Number
- ~ Lecture 2 : Representing Rational Number on Number Lines
- ~ Lecture 3 : Rational Number in Standard Form
- ~ Lecture 4 : Comparison of Rational Numbers, Rational Numbers between two Rational Numbers
- ~ Lecture 5 : EX 9.1 Q 1 to 5
- ~ Lecture 6 : EX 9.1 Q 6 to 10

#### => Practical Geometry :

- ~ Lecture 1 : Construct line parallel to given line & triangle if 3 sides are given
- ~ Lecture 2 : Construction triangle SAS, ASA & RHS

#### => Perimeter and Area :

- ~ Lecture1\_Introduction\_&\_Course\_Content
- ~ Lecture2\_Square\_&\_Rectangle
- ~ Lecture3\_NCERT\_EX\_11.1\_PROBLEM\_DISCUSSION
- ~ Lecture4\_Triangle\_As\_Part\_Of\_Rectangle
- ~ Lecture5\_Area\_Of\_Parallelogram
- ~ Lecture6\_Area\_Of\_Triangle
- ~ Lecture7\_NCERT\_EX11.2\_PROBLEM\_DISCUSSION
- ~ Lecture8\_Circle\_Circumference\_&\_Area\_Part1
- ~ Lecture8\_Circumference\_&\_Area\_Circle\_Part2
- ~ Lecture8\_Circumference\_&\_Area\_Circle\_Part3
- ~ Lecture9\_NCERT\_EX11.3\_PROBLEM\_DISCUSSION
- ~ Lecture10\_CONVERSION\_OF\_UNITS
- ~ Lecture11\_Applications\_Perimeter\_&\_Area
- ~ Lecture12\_NCERT\_EX11.3\_PROBLEM\_DISCUSSIONS

#### => Algebraic Expressions :

- ~ Lecture1\_Introduction\_&\_Topics
- ~ Lecture2\_Algebraic\_Terminologies
- ~ Lecture3\_Like\_Vs\_Unlike\_Terms
- ~ Lecture4\_NCERT\_EX\_12.1\_PROBLEM\_DISCUSSION
- ~ Lecture5\_Addition\_&\_Subtraction\_Algebraic\_Expressions
- ~ Lecture6\_NCERT\_EX\_12.2\_PROBLEM\_DISCUSSION
- ~ Lecture7\_Finding\_Value\_Of\_Algebraic\_Expression
- ~ Lecture8\_NCERT\_EX\_12.3\_PROBLEM\_DISCUSSION
- ~ Lecture9\_Number\_Pattern\_Rules
- ~ Lecture10\_NCERT\_EX12.4\_PROBLEM\_DISCUSSIONS
- ~ Lecture11\_Chapter\_Summary\_The\_End

#### => Exponents and Powers :

- ~ Lecture1\_Course\_Content\_&\_Introduction

- ~ Lecture2\_Exponents\_&\_Powers
- ~ Lecture3\_NCERT\_EX\_13.1\_PROBLEM\_DISCUSSIONS
- ~ LECTURE4\_LAWS\_OF\_EXPONENTS
- ~ LECTURE5\_NCERT\_13.2\_PROBLEM\_DISCUSSION
- ~ LECTURE6\_STANDARD\_FORM\_NOTATION
- ~ LECTURE7\_NCERT\_EX13.3\_PROBLEM\_DISCUSSION

=> Symmetry :

- ~ Lecture1\_Introduction\_To\_Symmetry
- ~ Lecture2\_Line\_Of\_Symmetry\_For\_Regular\_Polygon
- ~ Lecture3\_NCERT\_EX\_14.1\_PROBLEM\_DISCUSSION
- ~ Lecture4\_All\_about\_Rotational\_Symmetry
- ~ Lecture5\_NCERT\_EX\_14.2\_PROBLEM\_DISCUSSION
- ~ Lecture6\_Line\_&\_Rotational\_Symmetry
- ~ Lecture7\_NCERT\_EX\_14.3\_PROBLEM\_DISCUSSION

=> Visualising Solid Shapes :

- ~ Lecture1\_Introduction\_&\_Course\_Walkthrough
- ~ Lecture2\_All\_About\_Nets
- ~ Lecture3\_NCERT\_EX15.1\_PROBLEMS\_DISCUSSION
- ~ Lecture4\_Oblique\_Vs\_Isometric\_Sketches
- ~ Lecture5\_NCERT\_EX15.2\_PROBLEM\_DISCUSSION
- ~ Lecture6\_Visualizing\_Solid\_Objects
- ~ Lecture7\_Viewing\_Different\_Sections\_Of\_Solid

# AIOps Interview Questions

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS INTERVIEW PREPARATION

Course link : <https://ineuron.ai/course/AIOps-Interview-Questions>

## Course Description :-

Artificial Intelligence Operations is the most in-demand technical skill (AIOps). It facilitates the use of DevOps techniques in the creation of AI products. This course will cover a variety of approaches to implementing AIOps methodology in machine learning and deep learning projects, including implementation on AWS, Azure, Google Cloud Platform, and DigitalOcean.

## Course Features :-

=> Challenges

=> Quizzes

=> Assignments

## What you will learn :-

=> N

=> u

=> l

=> l

## Requirements :-

=> A system with stable internet connection

=> Your dedication

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Interview Questions :

~ AI Ops Question Discussion Part 1 Preview

~ AI Ops Question Discussion part 2

# Deep Learning Foundations

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/Deep-Learning-Foundations>

## Course Description :-

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. It is a function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Learn Deep Learning, Transfer Learning and Neural Networks using the latest frameworks. Become a Deep Learning Guru!

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Foundations of Deep Learning
- => Artificial Neural Networks
- => Convolution Neural networks
- => Natural Language Processing
- => RNN
- => LSTM

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Foundations of Deep Learning :

- ~ Introduction to Deep Learning
- ~ Why Deep Learning?
- ~ Difference between Machine learning & Deep Learning
- ~ Basic's of Deep learning

=> Artificial Neural Networks :

- ~ Neural Network Foundations
- ~ Forward propagation
- ~ Backward Propagation
- ~ Weight Initialization
- ~ Loss Function and Gradient Descent
- ~ Activation Function
- ~ Optimizers
- ~ Artificial Neural networks

=> Convolution Neural networks :

- ~ CNN vs ANN
- ~ Convolutional Neural Network
- ~ Filters
- ~ Channels/Feature Maps
- ~ Padding
- ~ Receptive Fields
- ~ Practical demonstration

# Mega Community

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Mega-Community>

## Course Description :-

This is a free community live class where multiple projects have been discussed by various mentors in the field of data science.

## Course Features :-

=> Everything will be discussed with python

## What you will learn :-

=> End to End data Science project

=> Tech used in industry

## Requirements :-

=> Your Dedication

=> Laptop

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Brand Measures

=> Automatic conversation AI from scratch without any framework

=> Automated machine learning +dev ops end to end with state of the art :

~ MLops Preview

=> The real machine learning project end to end

=> SQL interview question in detail

=> Mail Automation with power BI

=> Design pattern and full end to end application development

# VueJS Crash Course

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : VUE JS

Course link : <https://ineuron.ai/course/VueJS-Crash-Course>

## Course Description :-

This course will help you to grab the fundamentals of VueJs.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What is VueJS
- => VueJS project structure
- => Adding data and methods
- => Passing data to prop in Vue
- => Adding editable form in todo
- => Passing methods in Vue
- => Adding info to list

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> VueJS :

- ~ What is VueJS
- ~ VueJS project structure
- ~ Adding data and methods
- ~ Passing data to prop in Vue
- ~ Adding editable form in todo
- ~ Passing methods in Vue
- ~ Adding info to list

# Docker

---

Topic Name : DEVOPS

Sub-topic Name : DOCKER

Course link : <https://ineuron.ai/course/Docker>

## Course Description :-

Docker makes it easier to create, share, and operate contemporary programmes. Docker is a programme that uses containers to make it easier to construct, deploy, and manage applications.

## Course Features :-

- => Source code
- => Downloadable resources
- => Quizzes
- => Completion certificate

## What you will learn :-

- => Docker & its architecture
- => Docker as a service
- => Docker CLI
- => Docker Volumes
- => Port Mapping
- => Dockerizing a web application

## Requirements :-

- => Prior knowledge of linux
- => A System with good internet connection
- => How the bash works
- => Interest to learn

=> Your dedication

## Instructors :-

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Docker Introduction :

- ~ Introduction
- ~ What is Docker?
- ~ Why Docker?
- ~ Benefits of Docker
- ~ What is Container?
- ~ Containers vs VM
- ~ Containers vs Image
- ~ Docker Editions
- ~ What Docker is not?
- ~ Important Terminologies
- ~ Docker Setup in Windows
- ~ Docker Setup in Linux
- ~ Docker Setup in Mac

=> Basic Usage :

- ~ Docker Basic Commands part 1
- ~ Docker Basic Commands part 2

=> Docker Run :

- ~ Docker Run Part 1
- ~ Docker Run Part 2

=> Docker Images :

- ~ Docker Images
- ~ Creating a new image
- ~ Environment variables
- ~ Commands & Entry Points



=> Docker Compose :

- ~ *Docker Compose*
- ~ *Voting Application Understanding*
- ~ *Docker Compose Versions*
- ~ *Docker Compose Networks*
- ~ *Voting Application with Docker Run*
- ~ *Voting Application with Docker Compose*

=> Docker Concepts :

- ~ *Docker Engine*
- ~ *Docker Storage*
- ~ *Docker Networking*
- ~ *Docker Registry*

# Fundamentals of Web Designing

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Fundamentals-of-Web-Designing>

## Course Description :-

This course will teach about web development by making real websites from scratch with HTML and CSS. You can start by creating a website and that's one of the best ways to learn to code. This Skill Path will help you build strong fundamentals of HTML and CSS to define the structure and beauty of your website.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => One to One Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to the course
- => HTML Basic formatting tags
- => HTML Basic Formatting Tags
- => Grouping Tags
- => HTML Hyperlink
- => HTML Headers
- => CSS
- => CSS Selectors
- => CSS Background Cursor
- => CSS Text font
- => CSS List Style
- => CSS Box Models
- => CSS Display Positioning
- => CSS Advanced
- => Hands-on Projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Monal Kumar :

~ Monal Kumar is a data scientist and instructor working at iNeuron having 2+ years of total experience in both service and product-based organisations. He is specialised in Deep Learning, Computer vision and Image processing. Previously, he held positions as a support configurator at Wipro Technologies and as a Deep Learning researcher at Harptec Research. Offering the finest possible services to his clients. In addition to his primary job function, he is recognised for his creativity and ideas that change the nature of the existing problem.

## Curriculum details :-

- => Introduction to the course :
  - ~ Course Introduction
  - ~ Who this course for?
  - ~ Course Outcome
  - ~ Basics of Internet
  - ~ What is Web Development?
  - ~ What is Web Designing?

- ~ *What is Frontend?*
- ~ *What is Backend?*
- ~ *What is Website?*
- ~ *Differernt technologies in making the website*

#### => Assignment1 :

- ~ *What are the technologies used for Front End Development?*

#### => HTML Introduction :

- ~ *History of HTML*
- ~ *What are HTML Tags and Attributes?*
- ~ *HTML Tags vs. Element*
- ~ *Anatomy of an HTML element*
- ~ *Nesting elements*
- ~ *Images*
- ~ *Headings*
- ~ *Paragraphs*
- ~ *Lists*
- ~ *Links*
- ~ *Forms*

#### => HTML Basic Formatting Tags :

- ~ *HTML Headings*
- ~ *HTML basic tags*
- ~ *HTML Fomattting tags*
- ~ *HTML Colour coding*

#### => Grouping Tags :

- ~ *Div and Span Tags for grouping*

#### => HTML Lists :

- ~ *Ordered list*
- ~ *Unordered list*

#### => HTML Images :

- ~ *Image and Image Mapping*

#### => HTML Hyperlink :

- ~ *Uniform Resource Locator*
- ~ *What is a Uniform resource locator?*
- ~ *URL Encoding*

#### => HTML Table :

- ~ *What is Table in HTML?*
- ~ *Different types of HTML Tags used in Table*

#### => HTML Forms :

- ~ *What are Forms in HTML?*
- ~ *Different types of HTML tags used in Forms*

#### => HTML Headers :

- ~ *Title*
- ~ *Base*
- ~ *Link*
- ~ *Styles*
- ~ *Script*
- ~ *Meta*

#### => Assignment2 :

- ~ *Create a Portfolio webpage with your photo, bio, and personal details with your Family members*

#### => HTML 5 :

- ~ *What exactly Is HTML5?*
- ~ *Block vs. Inline Elements - Divs and Spans*
- ~ *An Odd Assortment of Elements: HR, BR, Sup and Sub*
- ~ *Entity Codes*
- ~ *Intro to Semantic Markup*
- ~ *Playing with Semantic Elements*
- ~ *Screen Reader Demonstration*
- ~ *Introducing HTML Tables*
- ~ *Tables: TR, TD, and TH Elements*
- ~ *Tables: Thead, Tbody, and Tfoot Elements*
- ~ *Tables: Colspan & Rowspan*
- ~ *The Form Element*
- ~ *Common Input Types*
- ~ *HTML Buttons*
- ~ *The Name Attribute*
- ~ *Radio Buttons, Checkboxes, & Selects*
- ~ *Range & Text Area*
- ~ *HTML5 Form Validations*

#### => Assignment3 :

- ~ *Create a webpage and display images of all the visiting places in your city and add their descriptions.*
- ~ *Favorite trip: If you could take the perfect trip, where would you go?*

*What animals would you see? What food would you eat? Add images*

*to this webpage after each paragraph tag, to share your perfect trip with the world.*

~ *Wishlist: What do you wish for? In this, you'll make a Wishlist, a list of activities you want to do or things you want to have.*

#### => CSS :

- ~ *What is CSS?*
- ~ *Benefits of CSS*

- ~ CSS versions history
- ~ CSS syntax
- ~ External style sheet using `<link>`
- ~ Multiple stylesheet

#### => CSS Selectors :

- ~ Id Selectors
- ~ Class Selectors
- ~ Grouping Selectors
- ~ Universal Selectors
- ~ Descendant Selectors
- ~ Attribute Selectors

#### => CSS Background Cursor :

- ~ Background-Image
- ~ Background-Repeat
- ~ Background-Position
- ~ CSS cursor

#### => CSS Text font :

- ~ Colour
- ~ Background colour
- ~ Text-decoration
- ~ Text-align
- ~ Text-transform
- ~ White space
- ~ Letter-spacing
- ~ Word-spacing
- ~ Line-height
- ~ Font family
- ~ Font-style
- ~ Font-variant

#### => CSS List Style :

- ~ List style type
- ~ List style position
- ~ List style image

#### => CSS Box Models :

- ~ Borders and Outline
- ~ Margin and Padding
- ~ Height and Width
- ~ CSS Dimensions

#### => CSS Display Positioning :

- ~ CSS Visibility
- ~ CSS Scrollbar
- ~ CSS Display
- ~ Static Positioning
- ~ Fixed Positioning
- ~ Relative Positioning
- ~ Absolute Positioning

#### => CSS float :

- ~ The float property
- ~ The clear property

#### => Assignment4 :

- ~ Modify the personal portfolio website build using HTML by adding different styles using CSS.
- ~ Create a Blog web page on your favorite topic ( Example: History, Biology, Physics, Science, etc.)

#### => Project1 :

~ Create an interactive website for your school and include different web pages like homepage, about page, facilities page, and fee structure page.

#### => Project2 :

~ Create and design a documentation website that will have subjects, important subtopics, and notes on that topics. Also, have an MCQs page that will provide the score after completing the test.

# Language Identification

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP PROJECT

Course link : <https://ineuron.ai/course/Language-Identification>

## Course Description :-

This is an audio classification project in which we will use Pytorch for audio processing and CNN for audio classification. We will use Indian language audio data from four classes, Hindi, Tamil, Telugu, and Kannada, and predict the language spoken in the audio.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => Language Identification
- => Audio preprocessing steps to build
- => Train and evaluate Deep learning models in PyTorch
- => Creating custom PyTorch dataset and dataloader
- => Use convolution neural network for audio classification.
- => Modular coding approach for training and prediction pipeline
- => Building Flask App
- => Learn about GCP basics
- => CICD tool like Github Actions for deployment

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Aravind S :

~ Data scientist with over a year of experience in developing advanced deep learning projects, core expertise in machine learning and NLP, proficient in data preprocessing and model building, and has closely mentored over 100 students from various domains.

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Project :- Language Identification :

- ~ Introduction of Instructor
- ~ Project Overview
- ~ End Notes
- ~ Problem Description
- ~ Understand the application scope
- ~ End Notes
- ~ Solution Description
- ~ Notebook Walkthrough
- ~ Tour to Architecture diagram
- ~ Cost involved
- ~ End Notes
- ~ Structure overview
- ~ Data Ingestion
- ~ Data Validation
- ~ Data Transformation
- ~ Model Training and Tuning
- ~ Model Evaluation
- ~ Model Pusher
- ~ Training Pipeline
- ~ Frontend app design
- ~ Tour to the cloud and Service Overview

- ~ IAM setup
- ~ GCP setup
- ~ Workflow
- ~ Adding Self hosted runner
- ~ Conclude the project
- ~ Points to improve from current project
- ~ Assignments & External Resources

# Digital Marketing Bootcamp in Hindi Tech Neuron

---

Topic Name : DIGITAL MARKETING

Sub-topic Name : DIGITAL MARKETING MASTERS

Course link : <https://ineuron.ai/course/Digital-Marketing-Bootcamp-in-Hindi-Tech-Neuron>

## Course Description :-

Digital Marketing Course Will Help You in businesses of all sizes by giving access to the mass market at an affordable price. Unlike TV or print advertising, it allows truly personalised marketing. Digital marketing also comes with a number of challenges you should be aware of. The main advantage of digital marketing is that a targeted audience can be reached in a cost-effective and measurable way. Other digital marketing advantages include increasing brand loyalty and driving online sales.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Digital Marketing benefit over Traditional Marketing
- => Social Media Branding
- => Website Planning and Building
- => Social Media Marketing
- => Type of SEO
- => Facebook Ads Manager
- => Facebook Pixel Integration
- => Twitter Marketing
- => Instagram Marketing
- => Linkedin Marketing
- => Content Marketing
- => Important Websites & Tools for Youtube
- => Different Types of Channels
- => Youtube Short Videos
- => Email Marketing
- => Google Ads
- => Affiliate Marketing
- => Influencer marketing

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Orientation Courses :
  - ~ Overview of Course
  - ~ Digital Marketing Definition
  - ~ Digital Marketing and Traditional Marketing Relationship
  - ~ Digital Marketing benefit over Traditional Marketing
  - ~ 4 P's of Marketers
  - ~ Future perspective of Digital Marketing
  - ~ Social Platforms for Digital Marketing
- => Social Media Branding :
  - ~ Bloggers Introduction
  - ~ Installation and customisation of Bloggers theme and plugins
  - ~ Domain Connection to Blogger

- ~ How to connect website with Google Analytics
- ~ How to submit website with Google Console
- ~ Learn to create meta tags, description and sitemap
- ~ Seo based content writing
- ~ How to avoid plagiarism
- ~ Monetization your writing
- ~ Learn Backlinks strategies
- ~ Content Marketing Tool
- ~ Selection of topics regarding Article Writing
- ~ Keyword Research and its KD
- ~ Examine Competitor Content
- ~ Assignment on Social Media Branding
- ~ Creating a Quiz and Test Module.

#### => Website Planning and Building :

- ~ Choose your Domain and its necessity.
- ~ Choosing the best and affordable hosting service.
- ~ How to install Wordpress and it's plugin
- ~ Important elements to set up in wordpress blogs.
- ~ Seo plugins to increase seo ranking.
- ~ Elementor Overview in Five Parts
- ~ Elementor Templates
- ~ Elementor Site Settings
- ~ WooCommerce and it's installation
- ~ Product Tab in WooCommerce
- ~ Tax Rates in WooCommerce
- ~ Shipping and Payment in WooCommerce
- ~ Linked and Group Products in WooCommerce
- ~ Technique to install best wordpress theme
- ~ News, Blogging, Professional Website in wordpress
- ~ Assignment on Website Planning and Building
- ~ Creating a Quiz and Test Module.
- ~ Search Engine Optimization
- ~ Learn Seo Fundamentals
- ~ History and Growth of Seo
- ~ Beginner to Advance Seo

#### => Type of SEO :

- ~ On Page Seo
- ~ What is on page seo, how to use it
- ~ Learn Seo in Blogger and Wordpress
- ~ How to add Meta Tag and it's description
- ~ Add Description in Blogger
- ~ Off Page Seo
- ~ Technical Seo
- ~ Local Seo
- ~ Keywords
- ~ Google Adwords
- ~ Assignment on SEO
- ~ Creating a Quiz and Test Module.

#### => Social Media Marketing :

- ~ Overview of Social Media Marketing
- ~ Understanding Social Media
- ~ Necessity of Social Media in Business
- ~ Creating your business brand using Social Marketing

#### => Facebook Marketing Overview :

- ~ What You'll Learn In This Course
- ~ Why Facebook Ads Are Important

#### => FB Page Setup :

- ~ How To Create & Optimize A Facebook Page
- ~ What Kind of Facebook Page Do You Want To Create
- ~ Top 5 SEO Tips For Your Facebook Page
- ~ What Makes Your Facebook Page Awesome
- ~ How To Create A Facebook Group For Your Business
- ~ How To Access The Facebook Ads Manager
- ~ Facebook Ads Manager vs Business Manager: What's the Difference?

#### => Facebook Ads Manager :

- ~ How to Build an Audience from Scratch!
- ~ How to Set Up an Advertising Account
- ~ Key Ad Policies (Facebook & Instagram)

#### => Ads Structure And Types :

- ~ Facebook Ad Structure
- ~ Create Your First Ad - Choose a Campaign Objective
- ~ What Is Brand Awareness Ads
- ~ What Is Traffic Ads
- ~ What Is App installs Ads
- ~ What Is Lead Generation Ads
- ~ Creating A Video Ads
- ~ Creating A Photo Ads
- ~ Creating A Carousel Ads
- ~ Targeting by Location & Demographics
- ~ Targeting By Interests
- ~ Targeting By Behaviors & Connection
- ~ Ad Placements

#### => Billing & Payment :



- ~ Budgets
- ~ Place Ad Order
- ~ Running Our First Ad
- ~ Ad Reporting
- ~ How to View Your Billing Summary

#### => Facebook Pixel Integration :

- ~ How to Install the Facebook Pixel
- ~ How to track your facebook ads
- ~ Conclusion

#### => (Scaling & Optimizing Ads) :

- ~ How To Optimize And Split Test FB Ads
- ~ How To Duplicate Ad Sets Correctly
- ~ How To Create a Lookalike Audience
- ~ What is Remarketing & How To Do It
- ~ Campaign Budget Optimization (CBO)
- ~ What To Do If FB Ad Account Disabled
- ~ Advance FB Ad Hacks

#### => (Sales Funnel & Website) :

- ~ What is a sales funnel & why do we need it
- ~ How to create a sales funnel (FREE & PAID METHOD)
- ~ How to insert pixel code in your funnel
- ~ How to comply your funnel with FB policies

#### => (Ads Creation & Designing) :

- ~ Which type of ad converts most?
- ~ How to create high converting ads
- ~ Don't Sell Like This (How To Use Storytelling For Selling)
- ~ Create video ads with voice over only
- ~ The right strategy for remarketing ads

#### => Twitter Marketing :

- ~ Overview of Twitter Marketing
- ~ Creating a brand name over Twitter
- ~ Twitter Marketing Strategy
- ~ Twitter Tools to Manage Twitter
- ~ Achieve Your Business Goals with Twitter
- ~ How to manage twitter story
- ~ Assignment on Twitter Marketing
- ~ Creating a Quiz and Test Module

#### => Instagram Marketing :

- ~ How to find niche for Instagram Account
- ~ How to make money through Instagram
- ~ How To Get 1000 Followers In 50 Days
- ~ Which Niche Has More Money
- ~ How to create Instagram Account
- ~ Types of Account on Instagram
- ~ How To Make Business Page Post Through Your Mobile
- ~ How To Make fashionable Product Page Post
- ~ How To Make Informational Page Post
- ~ How To Make Motivational Quotes Page Post
- ~ How to Create Instagram account
- ~ How To Make Business Page Post Through Your Mobile
- ~ How To Make fashionable Product Page Post
- ~ How To Make Informational Page Post
- ~ How To Make Motivational Quotes Page Post
- ~ Product Promotional Post for Instagram
- ~ Personal Branding Instagram Page
- ~ How to create & edit Instagram story
- ~ Perfect time to posting Video and Post
- ~ How to set up your Instagram profile Basic To Advance
- ~ How to create & edit Instagram story
- ~ How to convert your personal profile to business creator one
- ~ How to find Instagram
- ~ How to use Hashtag on Instagram Post
- ~ How to use Hashtag on Instagram Reels
- ~ How to use Hashtag on Instagram Story
- ~ How to create & edit Igtv videos
- ~ How to upload Igtv videos properly
- ~ How to create & edit Instagram reels
- ~ How to create & edit Instagram reels
- ~ How to upload Instagram reel properly
- ~ How to grow quote page on Instagram
- ~ How to increase Instagram story views hacks tricks
- ~ Instagram algorithm explained!
- ~ How to promote your post, story, reels (step by step guide)
- ~ Instagram insights fully explain!
- ~ Top 3 applications for Instagram Post Editing
- ~ Don't do these mistakes on Instagram
- ~ Fast Instagram growth tricks
- ~ How to download Instagram reels\_ 3 Method
- ~ Assignment on Instagram Marketing
- ~ Creating a Quiz and Test Module

#### => LinkedIn Marketing :

- ~ Overview of LinkedIn Marketing
- ~ Establishing Connection with LinkedIn
- ~ How LinkedIn help to Grow your Contacts over LinkedIn

- ~ How to Create groups in LinkedIn
- ~ How to Create Events in LinkedIn
- ~ Assignment on LinkedIn Marketing
- ~ Creating a Quiz and Test Module

#### => Content Marketing :

- ~ What is Content Writing and Making Effective Content for Website
- ~ Learn Seo Based Content Writing
- ~ Content Research and Planning
- ~ How to use Content marketing tools
- ~ What is Plagiarism and how to avoid it
- ~ What is Keyword and Keyword Description
- ~ Using right keywords
- ~ What is Blog and how to monetise it
- ~ Assignment on Content Marketing
- ~ Creating a Quiz and Test Module

#### => Why YouTube As a platform & How Youtube Works :

- ~ What is YouTube and how to earn through it
- ~ What is Amazon Affiliate Through Youtube
- ~ Youtube as a profession and business
- ~ Sponsorship through Youtube & another Multiple way of learning

#### => Channel Creation and Learn Seo Identify your Target Audience :

- ~ Creating Youtube Channel
- ~ How to Verify your Youtube Channel and apply custom Thumbnail
- ~ Decide Your Unique name of YouTube Channel
- ~ Making Professional Channel Art
- ~ How To Make Your Channel Brand New Logo
- ~ Complete Youtube Creators Studio
- ~ Youtube A to Z Complete Step by Step Settings
- ~ How To Upload Youtube Video With 100% SEO
- ~ Best Mic For Your Video & How To Use it
- ~ Best Camera for Video
- ~ How To Decide Your Profitable Niche

#### => Creating Your Videos Content :

- ~ How To Write an Effective Script for your videos
- ~ How do the best creators produce content consistently
- ~ well-researched and steadily consistent videos
- ~ Learn the secrets of high-quality channel

#### => Important Websites & Tools for Youtube :

- ~ How to get Copyright free videos and IMAGES
- ~ Learning to use Google Fonts
- ~ Using Google Ads to promote Youtube Videos
- ~ Complete tutorial of Social Blade, TubeBuddy tutorial, VidIQ tutorial
- ~ Creative Common Licence

#### => Useful Apps for Youtubers :

- ~ Best Video Editing App
- ~ Best Audio Editing App
- ~ Best Screen Recorder App
- ~ Youtube Video Analytics
- ~ Best Tag Finder For Youtube Video
- ~ Lower Third For Youtube Video
- ~ How To Make Professional Thumbnail for Your Youtube Video
- ~ How To Make Professional Intro or Outro For YOUTUBE Video
- ~ TubeBuddy For Tags

#### => Free Youtube Video, Audio etc. :

- ~ Download Audacity software ,install and use it
- ~ Type of Mic
- ~ You Tube Equipment Setup
- ~ How to set Chroma Properly

#### => 50 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Elementary class,

#### => 51 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Middle class

#### => 52 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ High School

#### => 53 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Professional Skills-

#### => 54 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Cooking,

#### => 55 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Yoga

#### => 56 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Sales

#### => 57 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Marketing

#### => 58 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

- ~ Startups

#### => 59 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

~ *Digital Marketing,*

=> 60 Niche for Youtube Channel One To One Session With Youtube Growth Expert :

~ *Affiliate Marketing, etc*

=> Fitness- Fitness Channels :

~ *Product Fit Without Gym*

=> Comedy Channels :

~ *Vines*

~ *Group*

=> Unboxing Videos :

~ *Unboxing Videos*

=> Bollywood Gossip Video :

~ *Bollywood Gossip Video*

=> Sports Video :

~ *Sports Video*

=> Fashion :

~ *Mens*

~ *Womens*

=> Different Types of Channels :

~ *Music and music tutorial channel*

~ *Food Challenges*

~ *Vlogs channel*

~ *Timelapse channel*

~ *Copy Paste channel*

~ *Interviews channel*

~ *Property channel*

~ *Facts Video- Day to Day*

~ *Life Hacks*

~ *Beauty product video*

~ *Product Comparison*

~ *Video Editing channel*

~ *Graphic Designing channel*

~ *Film Your Podcast*

~ *Give Business Advice*

~ *Hair Tutorial channel*

~ *Clothes Tips channel*

~ *Health Fruits channel*

~ *New Channel-Shot Film*

~ *Food Vlogging channel*

~ *Startup news-biopic*

~ *Case Studies*

~ *Charts/Graphs*

~ *Ebooks Reader*

~ *Cartoons/Illustrations*

~ *Book Summaries*

~ *Tool Reviews*

~ *Online earning application review channel*

~ *Share market*

~ *Personal finance*

~ *Infographics*

~ *Mind Maps*

~ *Online Game*

~ *Podcasts*

~ *Quotes*

~ *Quizzes*

~ *Cricket News Channel*

~ *Cricket Prediction*

~ *Baby Care Channel*

=> Youtube Short Videos :

~ *Understand Youtube Shorts Algorithm*

~ *Youtube Shorts Video Uploading*

~ *How To Shoot & Edit Youtube Shorts Videos*

~ *How To Upload Youtube Shorts Video*

~ *Youtube Shorts Video Complete 100% Seo*

=> Get 1000 subscribers in 90 days and many more tips :

~ *How to get 1000 subscribers and 4000 hours watch time in 2022*

~ *Rank as no. 1 Youtuber- Youtube seo step by step tutorial*

~ *Making videos trending in less views*

~ *Becoming digital business growth consultant*

~ *Engaging people in youtube channels*

~ *How to Increase watch time*

~ *Starting from 0 subscribers*

~ *Why youtubers fail to accomplish*

=> Pre Production Work Editing From Mobile Apps Power Director :

~ *Power Director Tutorial*

~ *Power Director Tutorial part 1*

~ *Power Director Tutorial part 2*

=> Kinemaster Expert :

~ *Basic of Kinemaster Expert*

~ *Important Setting of Kinemaster Expert*

~ *Learning Text Tool in Kinemaster Expert*

- ~ Use transition Tool in Kinemaster Expert
- ~ Learning to use Pan and Zoom in Kinemaster Expert
- ~ Remove Green Screen in Kinemaster Expert
- ~ Learning to Save Video in High Quality
- ~ How to add Subscribers button in Channel
- ~ How to use colour filters and adjustment tools
- ~ How to make 3D Mockup book
- ~ Kinemaster Latest update explained 720p
- ~ Editing Transitional type video 1080 p
- ~ Tutorial of Biography Channel
- ~ Kinemaster Latest update 5
- ~ Learning to animated videos like GIGL and Seeken
- ~ Learning to make pdf and slideshow in mobile
- ~ Recording and editing educational videos
- ~ Learning to import and export Kinemaster Video
- ~ Learning to edit Chroma Video in mobile phone
- ~ Audio Settings in Kinemaster
- ~ Learn to change video speed in Kinemaster
- ~ How to do Voiceover in Kinemaster
- ~ Learn to make scrolling text
- ~ How to use Keyframe Animation Tool and Handwriting Tool in Kinemaster
- ~ How to use Sticker Tool in Kinemaster

#### => Editing Through Laptop Filmora Tutorial Overview :

- ~ Begin your Video Editing Journey with Filmora
- ~ How to download and install Filmora X
- ~ Selection of Aspect Ratio
- ~ Overview of Filmora X Interface
- ~ Add and Adjust Background Music
- ~ Speed up and Slow down Video
- ~ Adding multiple video at single screen

#### => Functions of Filmora X :

- ~ How to add text and filters
- ~ How add filters
- ~ Adding Transitions between videos
- ~ Adding own logo
- ~ How to crop video
- ~ How to use pan and zoom tool
- ~ Removing green screen
- ~ How to use Keyframing

#### => Advance Lust pack and colour grading :

- ~ How to use colour matching
- ~ Learn to colour grade Video

#### => Advance Tutorials :

- ~ How to add scrolling text
- ~ Learn to blur face in video

#### => Learning to screen record in Filmora X :

- ~ Learning to screen record in Filmora X

#### => FilmoraGo App Tutorials :

- ~ 3 FilmoraGo App Tutorials

#### => Whiteboard Animation Video Tutorial :

- ~ How to make Animated Video on Android
- ~ Make Animation Video through Mobile Phone
- ~ Learn to script for Animation video
- ~ Learn to script for Animation video
- ~ How to make Animation Video Full Tutorial
- ~ How to make Animation Video Full Tutorial
- ~ How to make cartoon tutorial

#### => Make Thumbnail Like Successful Youtubers :

- ~ How To Make Professional Thumbnail
- ~ How To Make Attractive & Eye Catchy Thumbnail

#### => Canva Complete Tutorial :

- ~ How To make Clickbait Thumbnails

#### => Youtube Advance Class Session { Live Sessions } With Famous Youtuber :

- ~ How To Complete 1000 Subscribers & 4000 Hour Watch Time In 90 Days
- ~ How To Apply For Monetization Step By Step Process
- ~ Google Adsense Complete Process Setup & Detailed Class
- ~ How To Grow 10x Your Youtube Channel
- ~ Always Rank On Top Your Video
- ~ How To Compete With Your Competitor

#### => What is Quora Marketing? :

- ~ Quora Marketing Guide

#### => How to Use Canva :

- ~ Using Canva like a Pro

#### => Email Marketing :

- ~ Learn what is email marketing
- ~ Making Strategy for Email marketing
- ~ Email Marketing tools and analyze it
- ~ How to setting Email Marketing effectively
- ~ Learn to set up e-books in less than 15 minutes
- ~ Assignment on Email Marketing

~ *Creating a Quiz and Test Module*

#### **=> DROPSHIPPING :**

- ~ *Dropshipping Introduction*
- ~ *Shopify Store setup video*
- ~ *Product Research Product*
- ~ *Shopify product add with oberlo*
- ~ *Shopify Theme and Setting Final*
- ~ *Final Shopify Strategy*
- ~ *Shopify Marketing Strategy*
- ~ *Final how to design product in page in Shopify*
- ~ *Assignment on Dropshipping*
- ~ *Creating a Quiz and Test Module*

#### **=> Google Ads :**

- ~ *Introduction on Google Ads*
- ~ *Types of Google Ads*
- ~ *Display Ads*
- ~ *What is Google Ads?*
- ~ *Types of Audience*
- ~ *Google Ads Certification*
- ~ *Google Ads Strategy*
- ~ *Google Sequence*
- ~ *Google Shopping*
- ~ *Keyword Research*
- ~ *Location Targeting*
- ~ *Negative Keyword Research*
- ~ *How to increase Adsense CPC*
- ~ *Assignment on Google Ads*
- ~ *Creating a Quiz and Test Module*

#### **=> Google Tag Manager :**

- ~ *Google Tag Manager*
- ~ *How to install Google Tag Manager*
- ~ *Assignment on Google Tag Manager*
- ~ *Quiz on Google Tag Manager*

#### **=> Start Your Own Website :**

- ~ *How to become entrepreneur*

#### **=> Become a Amazon Seller :**

- ~ *How to Create Amazon Seller Account*
- ~ *Product listing on amazon*
- ~ *Single product add in amazon*
- ~ *GST registration for amazon seller*
- ~ *How to get gtin exemption on amazon*
- ~ *Live Classes on Amazon Seller*

#### **=> Affiliate Marketing :**

- ~ *Introduction*
- ~ *Welcome to the Course*
- ~ *Overview*
- ~ *What is Affiliate Marketing*

#### **=> Best Affiliate Marketing Platforms :**

- ~ *Gearbest.com*
- ~ *CJ.com (Communication Junction)*
- ~ *Affiliate Program Amazon.com*
- ~ *Jvzoo.com*
- ~ *Clickbank.com*

#### **=> Affiliate Marketing with Clickbank :**

- ~ *Setup clickbank account*
- ~ *Choosing Product*
- ~ *5 Best selling niches on Clickbank*
- ~ *Assignment on Affiliate Marketing*
- ~ *Quiz on Affiliate Marketing*

#### **=> Landing Page Building :**

- ~ *What is Landing page and why do we need it?*
- ~ *Tips for building a great landing page*
- ~ *Best tips for landing page building*
- ~ *Build a landing page using Mailchimp*
- ~ *Assignment on Landing Page Building*
- ~ *Quiz on Landing Page Building*

#### **=> Getting unlimited Traffic on Landing Page :**

- ~ *Generating traffic using facebook*
- ~ *Generating traffic using youtube*
- ~ *Generating traffic using instagram*
- ~ *Generating traffic using quora*
- ~ *Generating traffic using linkedin*
- ~ *Live Session on Traffic Landing Page*

#### **=> Influencer marketing :**

- ~ *Influencer marketing strategy*
- ~ *Content Marketing Strategy*
- ~ *How to find Influencers*
- ~ *How to manage to Engage your influencer*
- ~ *How to measure influencer marketing*
- ~ *Influencer Campaign*
- ~ *Assignment on Influencer Marketing*

~ Creating a Quiz and Test Module

#### => Ebooks for Marketing :

~ Video Marketing Ebook

~ Amazon Marketing Ebook

~ Quora Marketing Ebook

~ Ebook on DM career Guide

~ Final Millionaire Mindset book

#### => Automation / Sales Funnel :

~ The real meaning of the sales funnel & why we need it

~ Why funnels are the key to your success

~ Identify Your Ideal Customer Avatars

~ Paid Funnel Builder Vs Free Builder

~ How To Set Up ClickFunnels Account

~ What Type Of Funnel Is Right For You

~ Build Your Funnel Overview

~ Set Up Your Tracking

~ How I've Structured My Offer

~ Landing Page Examples

~ Write Your Killer Landing Page

~ Create Your Killer Landing Page

~ Optimize for Mobile View

~ Create a Privacy Policy & Disclaimer

~ Sales Page Examples

~ Build Your Sales Page

~ Launch and Test Overview

~ Complete Your Launch Checklist

~ Measure Your Success

#### => How To Make Money Online :

~ What is Digital Marketing

~ Ways Of Earning Through Digital Marketing

~ What is Affiliate Marketing

~ Best Affiliate Product Category

~ How To Make Money Through Instagram

~ How To Make Money Through Youtube

~ How To Make Money Through Facebook

~ How To Make Money Through Blogging

~ How To Make Money Through Whatsapp

~ How to Make Money Through E-Commerce

#### => Youtuber Success Story :

~ Amit Bhadana

~ Carryminati

~ Bhuvaneshwar Bam

~ Carryminati

~ Saurav Joshi Vlogs

~ Cooking Shooking Hindi

~ Nisha Madhulika

~ Round To Hell

~ Ashish Chanchlani

~ Round To Hell

~ Sandeep Maheshwari

~ Dr. Vivek Bindra

~ Mahatmaji Technical

~ CA Rachna Nagar

~ Technical Gururji

~ Flying Beast

~ Tech Burner

~ Tracking Tech

~ Sejal Kumar

~ Shruti Arjun Anand

~ Kabita Singh

~ Prajakta kohli

~ Khan Sir

~ Total Gaming

~ Dynamo Gaming

~ Ranveer Allahbadia (BeerBiceps)

~ Sanam Puri & Partners

~ Tanmay Bhatt

~ Puskar Raj Thakur

~ Satish Khuswah

~ Pranjal Kamra

~ Labour Law Adviser

~ UTKARSH CLASSES JODHPUR

~ Kumar Gaurav Sir

~ Hitesh Choudhary

~ Krish Naik

~ Sudhanshu Kumar

~ Code With Harry

~ Ankur Wariko

~ Teachnical Yogi

~ Manoj Dey

~ Aman Dhatarwal

~ Technical Gyan

~ Physics wallah

~ Jatt Prabhjot

~ Mr. Indian Hacker

~ Abhi & Niyu

- ~ Foodie We
- ~ Himesh Madaan
- ~ Mumbaiker Nikhil
- ~ Quiz and Test Module with Youtuber Success Story

=> 50 Digital Marketing Tools :

- ~ Mixpanel
- ~ Hotjar
- ~ HubSpot Forms
- ~ Proof
- ~ PersistIQ
- ~ Marketo
- ~ Google Analytics
- ~ Mailerlite
- ~ Zoho SalesIQ
- ~ MailChimp
- ~ Leadfeeder
- ~ ContactOut
- ~ Evernote
- ~ Pardot
- ~ Buffer
- ~ Mention
- ~ Feedly
- ~ REP
- ~ Pilai
- ~ Nimble
- ~ Hootsuite
- ~ Hubspot
- ~ Slack
- ~ Bitly
- ~ TailWind
- ~ Ahrefs
- ~ Unbounce
- ~ Screaming Frog
- ~ Keywords Everywhere
- ~ Keywords Everywhere
- ~ Trello
- ~ Builtwith
- ~ Yesware
- ~ Wordable
- ~ Builtwith
- ~ Hunter.io
- ~ Ontraport
- ~ Facebook Manager Ads
- ~ AnswerThePublic.com
- ~ Nightwatch
- ~ Mangools
- ~ SE Ranking
- ~ Screaming Frog
- ~ SEMRush
- ~ Moz Open Site Explorer
- ~ Moz Pro
- ~ Google Alerts
- ~ Google Trends
- ~ Google Forms
- ~ Quiz and Test Module of 50 Digital Marketing Tools

# Machine Learning Bootcamp Tech Neuron

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-Bootcamp-Tech-Neuron>

## Course Description :-

In this Machine Learning Bootcamp you will learn technologies like Python, API, database, statistics, ML algorithms, deployment of ML models in various cloud platforms, and all machine learning algorithms. You will also learn about chatbots like Dialogflow, Amazon Lex, Azure Luis & RASA NLU. 15+ live projects are included to make your journey interesting from Zero to ML Engineer.

## Course Features :-

- => Machine Learning in Depth
- => CI/CD pipeline for ML
- => End to End Model Deployment in Azure, GCP & AWS
- => Time Series end-to-end implementation in ML
- => 20 + hands-on industry real-time projects
- => Power BI and Tableau self-placed course
- => 150+ hours live interactive classes
- => Doubt clearing session after the live classes
- => Doubt clearing one-to-one
- => Doubt clearing through mail and support team
- => Assignment in all the modules
- => 20+ use cases of Machine learning
- => Live project with real-time implementation
- => Online Instructor-led learning

## What you will learn :-

- => Python
- => APIs
- => Databases
- => Python projects
- => Numpy
- => Pandas
- => Visualizations
- => Stats
- => Supervised Machine learning Algorithms
- => Unsupervised Machine learning Algorithms
- => Dimensionality Reduction
- => Machine Learning Projects
- => Deep learning
- => PowerBI
- => Tableau
- => Chatbots

## Requirements :-

- => Dedication
- => Laptop with internet connectivity

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sunny Savita :

~ I'm an AI enthusiast, graduate in Computer science and engineering. Currently working with iNeuron.ai as a Data Scientist and having 2+ years of experience. I have skills in big data, machine learning, computer vision, Natural language processing. My expertise also includes project design development and implementation with AIOps tools.



## Curriculum details :-

### => Course Introduction :

- ~ *Introduction of Data Science, AI, ML, DL and its application in Day to Day life*
- ~ *Course overview and Dashboard description*

### => Installation and setup of the required software :

- ~ *Installation and setup of Anaconda Distribution*
- ~ *Installation and setup of Pycharm and VScode*
- ~ *Complete walk-through of Jupyter Notebook in local*
- ~ *Setup of Google Colab with GPU*
- ~ *Create a virtual environment through anaconda and project setup*

### => Introduction of Python :

- ~ *Python Introduction and comparison with other Programming language*
- ~ *Important Features of python*
- ~ *Testing Python Installation with hello world*
- ~ *Introduction To Predefined Functions And Modules*
- ~ *How print() function works ?*
- ~ *How To Remove Newline From print( ) ?*
- ~ *Rules For Identifiers, Python Reserved Words, Data Types In Python*
- ~ *Operators Arithmetic, Bitwise, Comparison and Assignment operators, Operators Precedence and associativity*
- ~ *Compound Operators, Identity Operators, Membership Operators*

### => String :

- ~ *What Is A String ?*
- ~ *Creating A String*
- ~ *Different Ways Of Accessing Strings*
- ~ *Operators Which Work On Strings*
- ~ *Built In String Functions*
- ~ *Printing string using f-string*
- ~ *Modifying Strings*
- ~ *String conversion methods*
- ~ *String comparison methods*
- ~ *String searching methods*
- ~ *String replace methods*

### => List :

- ~ *What Is A List ?*
- ~ *Creating A List*
- ~ *Accessing The List Elements*
- ~ *Adding New Data In The List*
- ~ *The Slice Operator With List*
- ~ *Modifying A List*
- ~ *Deletion In A List*
- ~ *Appending / Prepending Items In A List*
- ~ *Multiplying A List*
- ~ *Membership Operators On List*
- ~ *Built In Functions For List*
- ~ *Methods Of List*
- ~ *List Comprehension*

### => Tuples :

- ~ *What Is A Tuple and how to create Tuple*
- ~ *Differences between List and Tuples*
- ~ *Benefits Of Tuple*
- ~ *Packing / Unpacking A Tuple*
- ~ *Accessing A Tuple*
- ~ *Changing The Tuple*
- ~ *Deleting The Tuple*
- ~ *Functions Used With Tuple*
- ~ *Methods Used With Tuple*
- ~ *Operations Allowed On Tuple*

### => Dictionaries and set :

- ~ *What Is A Dictionary ?*
- ~ *What Is Key-Value Pair ?*
- ~ *Creating A Dictionary*
- ~ *Important Characteristics Of A Dictionary*
- ~ *Different Ways To Access A Dictionary*
- ~ *Updating Elements In Dictionary*
- ~ *Removing Elements From Dictionary*
- ~ *Functions Used In Dictionary*
- ~ *Dictionary Methods*
- ~ *Set introduction*
- ~ *Set methods*

### => Decision Control Statements and loops in python :

- ~ *if Statement*
- ~ *Concept of Indentation*
- ~ *if-else Statement*
- ~ *if-elif-else Statement*
- ~ *Types of loop supported by Python*
- ~ *while loop*
- ~ *while-else loop*
- ~ *Break, continue and pass Statement*
- ~ *for Loop*
- ~ *for Loop In Python*
- ~ *Differences with other languages*
- ~ *range( ) Function*

~ Using for with range( )

## => Python Functions :

- ~ What Is A Function ?
- ~ Function V/s Method
- ~ Steps Required For Developing User-Defined Function
- ~ Calling A Function
- ~ Returning Values From Function
- ~ Arguments V/s Parameters
- ~ Types Of Arguments
- ~ Variable Scope
- ~ Local Scope
- ~ Global Scope
- ~ Argument Passing
- ~ Anonymous Functions OR Lambda Function
- ~ The map( ) Function
- ~ The filter( ) Function
- ~ Using map( ) and filter( ) with Lambda Expressions
- ~ Iterators Generator functions

## => OOPS Concepts :

- ~ Procedure Oriented Programming vs Object Oriented Programming
- ~ What Is A Classes and Object ?
- ~ \_\_init\_\_() Method
- ~ Types Of variable in class
- ~ Types Of Methods in class
- ~ Difference Between local variable, class variable and Instance variable
- ~ Difference Between Instance Method, Class Method and Static Methods
- ~ concept of Encapsulation
- ~ How To Declare Private Members In Python ?
- ~ The setattr( ) And getattr( ) Functions
- ~ object Class, \_\_repr\_\_() and \_\_str\_\_() methods
- ~ concept of Inheritance
- ~ Types Of Inheritance
- ~ Single Inheritance
- ~ Using super( )
- ~ Method Overriding
- ~ MultiLevel Inheritance
- ~ Hierarchical Inheritance
- ~ Multiple Inheritance
- ~ The MRO Algorithm
- ~ Hybrid Inheritance
- ~ The Diamond Problem
- ~ Operator Overloading

## => Exception Handling :

- ~ Introduction To Exception Handling
- ~ Exception Handling Keywords
- ~ Exception Handling Syntax
- ~ Handling Multiple Exceptions
- ~ Handling All Exceptions

## => Python logging :

- ~ What is logging?
- ~ When to use logging?
- ~ Logging to a file
- ~ Different level of logging
- ~ Logging from multiple module
- ~ Logging variable data
- ~ Display Date&Time in logging file

## => Working With Files :

- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other File methods

## => Database :

- ~ What Is A Database ?
- ~ Steps Needed For Connecting To mysql From Python
- ~ Exploring Connection And Cursor Objects
- ~ Executing The SQL Queries
- ~ Different Ways Of Fetching The Data
- ~ Executing INSERT Command
- ~ Executing Update Command
- ~ Executing Delete Command
- ~ Introduction MongoDB
- ~ What is Apache Atlas and features of Apache Atlas
- ~ MongoDB atlas setup
- ~ Querying the documents
- ~ Finding, Inserting, Deleting & Updating elements
- ~ Bulk insert operations
- ~ Updating multiple document
- ~ Understanding insertOne vs insertMany()
- ~ Updateone() vs updateMany()
- ~ Understanding find() & fetchall()
- ~ Understanding "deleteOne()" & "deleteMany()"
- ~ Filtering documents

## => API :

- ~ Flask Introduction

- ~ Flask variable rules
- ~ Flask templates and static files
- ~ App Routing Flask
- ~ URL Building Flask
- ~ HTTP Methods Flask
- ~ Flask requesting object
- ~ Flask sending Form data to Template

#### => Python Pandas Modules :

- ~ Pandas Series
- ~ Pandas DataFrame
- ~ Pandas Panel
- ~ Pandas Basic functionality
- ~ Pandas read csv
- ~ Pandasread json
- ~ Pandas reading data from mysql
- ~ Pandas aggregations
- ~ Pandas group by
- ~ Pandas merging and joining
- ~ Pandas concatenation operation
- ~ Pandas date functionality
- ~ Pandas .loc() and .iloc() function
- ~ Pandas windows functions
- ~ Pandas indexing and selecting data
- ~ Cleaning data with pandas
- ~ Working with missing data
- ~ Working with categorical data

#### => Python Numpy Modules :

- ~ NumPy Narray Object
- ~ NumPy Data Types
- ~ NumPy Array Attributes
- ~ NumPy Array Creation Routines
- ~ NumPy Array from Existing
- ~ Data Array From Numerical Ranges
- ~ NumPy Indexing & Slicing
- ~ NumPy Advanced Indexing
- ~ NumPy Broadcasting
- ~ NumPy Iterating Over Array
- ~ NumPy Array Manipulation
- ~ NumPy Binary Operators
- ~ NumPy String Functions
- ~ NumPy Mathematical Functions
- ~ NumPy Arithmetic Operations
- ~ NumPy Statistical Functions
- ~ Sort , Search & Counting Functions
- ~ NumPy Byte Swapping
- ~ NumPy Copies Views
- ~ NumPy Matrix Library
- ~ NumPy Linear Algebra

#### => Python Visualization Modules :

- ~ Matplotlib Pyplot
- ~ Matplotlib Plotting
- ~ Matplotlib Subplot
- ~ Matplotlib Line Chart
- ~ Matplotlib Bar Chart
- ~ Matplotlib Histogram Chart
- ~ Matplotlib Pie chart
- ~ Seaborn Histogram
- ~ Seaborn Kernel density estimates
- ~ Seaborn Facet grid
- ~ Seaborn Pairgrid
- ~ Seaborn Boxplot, violin plot and contour plot
- ~ Seaborn Countplot
- ~ Seaborn Heatmap
- ~ Plotly Barchart histogram and pie chart
- ~ Plotly scatter plot and Bubble chart
- ~ Plotly distplot, density plot, and error bar plot
- ~ Plotly Heatmap
- ~ Plotly 3-D scatter plot and surface plot
- ~ Plotly with pandas and cufflinks
- ~ Plotly with matplotlib and chartstudio
- ~ Visualizing pairwise relationship
- ~ Finding statical estimation
- ~ Finding linear relationship
- ~ Finding correlation between variable

#### => Statistics :

- ~ Introduction
- ~ Different types of Statistics
- ~ Population vs Sample
- ~ Mean, Median and Mode
- ~ Variance, Standard Deviation
- ~ Sample Variance why n-1
- ~ Standard Deviation
- ~ Variables
- ~ Random Variables
- ~ Percentiles & quartiles
- ~ 5 number summary

- ~ Histograms
- ~ Gaussian - Normal distribution
- ~ Standard Normal distribution
- ~ Application Of Zscore
- ~ Basics Of Probability
- ~ Addition Rule In Probability
- ~ Multiplication rule in probability
- ~ Permutation
- ~ Combination
- ~ Log Normal Distribution
- ~ Central Limit theorem
- ~ Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median And Mode
- ~ Covariance
- ~ Pearson And Spearman Rank Correlation
- ~ What is P Value
- ~ What is Confidence Intervals
- ~ How To Perform Hypothesis Testing - Confidence Interval Z Test Statistics Derive Conclusion
- ~ Hypothesis testing part 2
- ~ Hypothesis testing part 3
- ~ Finalizing statistics

#### => Exploratory Data Analysis :

- ~ Feature Engineering and Selection
- ~ Create a profile of the data
- ~ Perform statical analysis
- ~ Building Tuning and Deploying Models
- ~ Perform EDA with automated library
- ~ Analyzing Bike Sharing Trends
- ~ Analyzing Movie Reviews Sentiment
- ~ Customer Segmentation and Effective Cross Selling
- ~ Analyzing Wine Types and Quality
- ~ Analyzing Music Trends and Recommendations
- ~ Forecasting Stock and Commodity Prices

#### => Machine Learning Module 1 :

- ~ Introduction of machine learning
- ~ Difference between Supervised, Unsupervised & Semi-supervised
- ~ Linear Regression Mathematical Institution
- ~ Linear Regression assumption.
- ~ OLS
- ~ Different Training methodology
- ~ Train, Test, Validation Split
- ~ Hands-on linear regression in python from scratch
- ~ Complete hands-on with scikit learn
- ~ Overfitting & Underfitting
- ~ Ridge Regression
- ~ Lasso Regression
- ~ Elastic Net Regression
- ~ Polynomial Regression
- ~ Logistics regression
- ~ Difference between Linear Regression and Logistic Regression
- ~ Performance matrix
- ~ Confusion matrix
- ~ Precision, Recall, ROC, AUC Curve
- ~ F-beta Score

#### => Machine Learning Module 2 :

- ~ SVR(support vector regressor)
- ~ SVC(support vector classifier)
- ~ SVM(Support vector machine)
- ~ KNN Classifier
- ~ KNN Regressor
- ~ K Nearest Neighbour
- ~ Lazy learners
- ~ KNN Issues
- ~ Performance measurement of KNN

#### => Machine Learning Module 3 :

- ~ Decision Tree Classifier
- ~ Decision tree Regressor
- ~ Cross Validation
- ~ Bias vs Variance
- ~ Ensemble approach
- ~ Bagging
- ~ Boosting
- ~ Stacking
- ~ Random Forest

#### => Machine Learning Module 4 :

- ~ Ada boosting
- ~ Gradient boosting
- ~ XGBoosting
- ~ Hands-on XgBoost

#### => Unsupervised Machine Learning :

- ~ Introduction to K-Means Clustering
- ~ Hard K-Means clustering
- ~ Soft K-Means clustering
- ~ Visualizing Each Step of K-Means
- ~ How to Choose K value

- ~ Advantages and Disadvantages of K-Means Clustering
- ~ Examples of where K-Means can fail
- ~ How to Evaluate a Clustering algorithm
- ~ Silhouette Coefficient
- ~ Dunn's Index
- ~ Python implementation using K-Means on Real Data
- ~ Real-time Clustering Application
- ~ Visual Walkthrough of Agglomerative Hierarchical Clustering
- ~ Using Hierarchical Clustering in Python and Interpreting the Dendrogram
- ~ python implementation of Agglomerative Clustering
- ~ DBSCAN: A Density-Based Clustering Algorithm
- ~ How to use DBSCAN: A Density-Based Clustering Algorithm for outlier detection
- ~ Python implementation of DBSCAN

#### => Dimension Reduction Techniques :

- ~ Principal Component Analysis (PCA)
- ~ T-distributed Stochastic Neighbor Embedding(t-SNE)
- ~ Curse of Dimensionality

#### => Natural Language Processing :

- ~ Text Analytics
- ~ Tokenizing, Chunking
- ~ Document term
- ~ Matrix TFIDF
- ~ Sentiment analysis hands-on
- ~ Naive Bayes classifier

#### => Deep Learning :

- ~ Deep Learning Introduction.
- ~ Neural Network Architecture.
- ~ Loss Function.
- ~ Cost Function.
- ~ Optimizers.
- ~ CNN architecture.
- ~ Build First Classifier in CNN.
- ~ Deploy Classifier over cloud.
- ~ RNN overview.
- ~ GRU.
- ~ LSTM.
- ~ Time Series using RNN LSTM.
- ~ Customer Feedback analysis using RNN LSTM.

#### => Time series :

- ~ Arima
- ~ Sarima .
- ~ Auto Arima
- ~ Time series using RNN LSTM .
- ~ Prediction of NIFTY stock price.

#### => Machine Learning Deployment :

- ~ Deployment of all the project in Cloudfoundary, AWS, AZURE & Google Cloud Platform
- ~ Expose api to web browser and mobile application retraining approach of Machine learning model
- ~ Devops infrastructure for machine learning model
- ~ Database integration and scheduling of machine learning model and retraining custom machine learning training approach.
- ~ AUTO ML
- ~ Discussion on infra cost and data volume
- ~ Prediction based on streaming data

#### => Machine Learning Extra Sessions :

- ~ Discussion on project explanation in interview
- ~ Data scientist roles and responsibilities
- ~ Data scientist day to day work
- ~ Companies which hire a data scientist
- ~ Resume discussion with our team one to one

# App Building Using Android

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : ANDROID

Course link : <https://ineuron.ai/course/App-Building-Using-Android>

## Course Description :-

This course will teach students how to use core Java programming to create mobile apps. Create your first Android app with Android Studio, learn how to run it on a device or emulator, and add interactive widgets and objects to your app. You will learn how to use Android Studio to get started with Android development and how to publish your own apps on the Google Play store.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => One to One Doubt Clearing
- => Assignment in all the Module
- => Quizzes
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to JAVA
- => Activity in android
- => View in android
- => Widgets, Pixel, and Layout
- => Adaptors in android
- => Intents in android
- => Gestures and Fragments
- => Data Storage
- => Database storage
- => Creating android projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ What is Android?
  - ~ Android platform architecture
  - ~ Basic requirements to create your own app
  - ~ What is Java?
  - ~ Role of Java in Android development
  - ~ Android Versions
  - ~ Challenges of Android development
  - ~ Downloading Android Studio
  - ~ What is Android studio?
- => Assignment 1: :
  - ~ Name the other mobile operating systems?
  - ~ Write down the market share ratio's of mobile operating systems?
- => Android studio installation :
  - ~ Installing Android studio
  - ~ Overview of the interface
  - ~ Basic App development workflow with Android studio
  - ~ What is Android virtual device?
  - ~ What is Android physical device?
- => Assignment 2: :
  - ~ Create your own 'Hello world!' app in Android studio

=> Introduction to JAVA :

- ~ What is Java programming language?
- ~ What is Inheritance?
- ~ What is class and methods?
- ~ What is loop?
- ~ What is variable and objects?

=> Assignment 3: :

- ~ Why is Java not a pure object oriented language?

=> Activity in Android :

- ~ What is Activity?
- ~ Activity life-cycle
- ~ Methods of activity life-cycles
- ~ What is Manifest files?
- ~ Configuring manifest files
- ~ What is Toast in Android?
- ~ How to create a Toast message?

=> Assignment 4: :

- ~ Create a simple app and add a toast message into it.

=> View :

- ~ Introduction to View group class
- ~ Introduction to View class
- ~ What is Linear View?
- ~ What is Relative View?
- ~ What is Absolute view?
- ~ View attributes
- ~ What is TextView?
- ~ What is EditText?
- ~ What is Button?
- ~ Working with images
- ~ What is radio button?
- ~ what is check boxes?

=> Widgets, Pixels and Layout :

- ~ What are Widgets?
- ~ What are Properties?
- ~ How to add Properties to widgets?
- ~ How to add Properties to widgets using Java?
- ~ What is multiple widgets?
- ~ What is Pixels?
- ~ What are Density Pixels?
- ~ What are Density Independent pixels?
- ~ What is Layout?
- ~ How to add Layout to activity?
- ~ What is Grid Layout?

=> Adaptors in Android :

- ~ What are Adaptors?
- ~ Types of Adaptors
- ~ What is List view?
- ~ Steps to implement List view

=> Intents in Android :

- ~ What is Intent?
- ~ What are the different types of Intent?
- ~ Fundamental use cases of Intents
- ~ Passing data
- ~ Types of Data

=> Gestures and Fragments :

- ~ What are Gestures?
- ~ How to use Gestures?
- ~ What are Fragments?
- ~ How to use Fragments?

=> Data Storage :

- ~ Introduction to data storage
- ~ Shared preferences
- ~ Modes and Creation of shared preferences
- ~ What is Internal storage?
- ~ What is External storage?

=> Database storage :

- ~ What is Database?
- ~ What is SQLite Database?
- ~ Database creation and version management

=> Publishing App :

- ~ What is Google app store?
- ~ How to publish apps to Google app store?

=> Project 1 :

- ~ Create your own scientific calculator app

=> Project 2 :

- ~ Create your own app which includes clock and stopwatch

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Alteryx>

## Course Description :-

Upskill your analytics skills through the power of Alteryx Platform. Alteryx is an analytical platform enabling us to explore and visualize new discoveries in data through immersive dashboards and advanced analytics. You will be able to enhance your skill by exploring this Alteryx Analytical platform.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Alteryx Features
- => Analytics
- => Alteryx visualization
- => Dashboard

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Course Introduction :

- ~ Syllabus overview Preview
- ~ Alteryx installation Preview
- ~ Alteryx features

=> Dashboard :

- ~ Alteryx dashboard
- ~ Dashboard features
- ~ Loading data
- ~ Interactive graphs



# GCP Machine Learning Certification

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/GCP-Machine-Learning-Certification>

## Course Description :-

This course will give you the skills you need to enhance your career, as well as instruction to help you prepare for the Google Cloud Professional Machine Learning Engineer certification.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Big Data & ML Fundamentals
- => Recommendation System
- => BigQuery ML
- => BigQuery Keypoints
- => BigQuery Features
- => PubSub Features
- => AI Platform
- => DataPrep Tool
- => Understanding ML Workflow
- => What is vertex AI
- => Pricing
- => Managing ML Datasets with vertex AI
- => Managed dataset handson
- => Speech-to-Text using UI Console
- => Speech-to-text using gcloud
- => Speech-to-text using python library

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Khushali Shah :

*~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.*

## Curriculum details :-

- => Introduction :
  - ~ Syllabus Overview
  - ~ Things to learn
- => Big Data & ML Fundamentals :
  - ~ Necessity of Compute Power for ML Workloads
  - ~ Storage, networking & security importance
  - ~ Big Data & Big query tools
- => Recommendation System :
  - ~ Recommendation System Fundamentals
  - ~ Diff products of GCP for recommendation system
  - ~ Practical:- Query Public Dataset in bigquery

~ Practical:- Loading and querying data

=> BigQuery ML :

- ~ BigQuery ML
- ~ BigQuery Keypoints
- ~ BigQuery Features
- ~ Bigquery GIS Example
- ~ ML Flow
- ~ ML Process
- ~ BigQueryML create Model
- ~ BigqueryML evaluate model
- ~ BigqueryML predict model

=> PubSub/Dataflow/Pipeline :

- ~ Overview of pipeline
- ~ PubSub Features
- ~ Core concepts
- ~ Handson send/receive messages with gcloud
- ~ Dataflow

=> ML :

- ~ Overview
- ~ AI Platform
- ~ DataPrep Tool
- ~ Understanding ML Workflow
- ~ What is vertex AI
- ~ Pricing
- ~ Managing ML Datasets with vertex AI
- ~ Managed dataset handson
- ~ Build & Train ML model with vertex AI
- ~ Training a model using AutoML Vertex AI
- ~ AutoML training & deployment
- ~ AutoML prediction
- ~ Clean up resources
- ~ Overview of custom model
- ~ Enable API
- ~ Create Notebook instance
- ~ Create container image
- ~ custom ML model training with vertex AI
- ~ Deployment
- ~ Prediction
- ~ Image classification application overview
- ~ Collection of yoga pose image dataset
- ~ Build image classification model with vertex AI
- ~ Deployment & prediction
- ~ Vertex AI SDK
- ~ Vertex AI SDK code walkthrough
- ~ Vertex AI SDK code walkthrough -2
- ~ Vertex AI SDK handson-1
- ~ Vertex AI SDK handson-2
- ~ Vertex AI SDK handson-3
- ~ Vertex AI SDK handson-4
- ~ Vertex AI SDK handson-5
- ~ Vertex AI SDK handson-6
- ~ What is Hyperparameter tuning
- ~ Hyperparameter tuning
- ~ Hyperparameter tuning mechanism
- ~ Hyperparameter tuning handson-1
- ~ Hyperparameter tuning handson-2
- ~ Hyperparameter tuning handson-3
- ~ Hyperparameter tuning handson-4

=> Conversational AI :

- ~ Overview
- ~ Speech-to-Text using UI Console
- ~ Speech-to-text using gcloud
- ~ Speech-to-text using python library
- ~ Different types of Speech recognition
- ~ Asynchronous speech recognition demo
- ~ Assignment Speech-text for streaming data
- ~ Cloud Text-to-Speech overview
- ~ TTS using python handson
- ~ TTS list down voices available
- ~ TTS basics
- ~ TTS SSML to audio handson
- ~ Assignment Translating text from a photo
- ~ Cloud Natural Language AI Overview
- ~ Analyze sentiment
- ~ Sentiment\_analysis\_in\_different\_language
- ~ Analyze entities in a string
- ~ Classify text
- ~ Analyzing syntax
- ~ Analyzing entity sentiment
- ~ Video Intelligence

=> Designing AI systems :

- ~ Problem overview
- ~ Change in Data
- ~ System Failure
- ~ Concept of drift

- ~ *Risks in ML system*
- ~ *Rules of ML*
- ~ *Best Practices of data*
- ~ *Hybrid ML Model*
- ~ *Kubeflow*

=> MLOps Fundamentals :

- ~ *Basics*
- ~ *Implementation:- Creating GKE Cluster with kubeflow*
- ~ *Implementation:- defining custom component using KFP SDK*
- ~ *Implementation:- Running/deploying pipeline*
- ~ *CI/Cd pipeline*
- ~ *Cloud Build trigger*

=> Explainable AI :

- ~ *Overview of explainable AI*
- ~ *Vertex Explainable AI*

=> GCP ML Certification sample questions :

- ~ *QnA part-1*
- ~ *QnA part-2*
- ~ *QnA part-3*

# Machine Learning Live Class

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-Live-Class>

## Course Description :-

This is Machine Learning course, where you will learn various things from the beginning like python, API, deployment in AWS, Azure, GCP, Heroku, Database, various modules in statistics, all machine learning algorithms.

## Course Features :-

- => Online Classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Master machine learning on python
- => Make robust machine learning models
- => Use machine learning for personal purpose
- => Handle advanced techniques like dimensionality reduction
- => Classify data using K-Means clustering, Support Vector Machines (SVM), KNN, Decision Trees, Naive Bayes, and PCA

## Requirements :-

- => Basic knowledge of python programming
- => A system with a stable internet connection
- => Your dedication

## Instructors :-

=> Sagar Kandpal :

~ Sagar Kandpal completed M.Tech. in Computer Integrated Manufacturing from NIT Warangal in 2019. He also worked as a teaching assistant in NIT Warangal. He was also working as a PhD research fellow at IIT Gandhinagar before joining iNeuron as a Data Scientist. His main research is focused on the Application of Machine Learning and Deep Learning algorithms to real-world use cases. Research Interests: Mathematical Modelling and Optimization, Deep Reinforcement Learning, Geometric Deep Learning, Robotics and Computer Vision.

## Curriculum details :-

=> Machine Learning Module 1 :

- ~ Introduction machine learning module 1
- ~ Supervised, unsupervised, semi-supervised, reinforcement
- ~ Train, test, validation split
- ~ Performance
- ~ Overfitting, underfitting
- ~ OLS
- ~ Linear regression
- ~ polynomial regression
- ~ Assumptions R-square adjusted, R-square intro to Scikit-learn, training methodology, hands-on linear regression, ridge regression, logistics regression, precision-recall

=> Machine Learning Module 2 :

- ~ Decision tree, decision tree regressor, cross-validation
- ~ Bias vs variance, ensemble approach, Bagging, boosting
- ~ Random forest, stacking, variable importance
- ~ XGBoost, hands-on XGBoost, gradient boost, ada boost

=> Machine Learning Module 3 :

- ~ K Nearest Neighbour, k-NN regressor, lazy learners, the curse of dimensionality, k-NN issues

=> Machine Learning Module 4 :

- ~ K-means, hierarchical clustering, DBSCAN
- ~ Performance measurement, principal component analysis, dimensionality reduction

=> Machine Learning Module 5 :

- ~ Naive Bayes SVM
- ~ Anomaly detection

=> Time series :

- ~ Arima, Sarima, Auto Arima
- ~ Time series using RNN LSTM, prediction of NIFTY stock price

# Machine Learning

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning>

## Course Description :-

This is Machine Learning masters, where you will learn various things from the beginning like python, API, deployment in AWS, Azure, GCP, Heroku, Database, various modules in statistics, all machine learning algorithms.

## Course Features :-

- => Quizzes
- => Assignments
- => Hands-on practical's
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Master machine learning on python
- => Make robust machine learning models
- => Use machine learning for personal purpose
- => Handle advanced techniques like dimensionality reduction
- => Classify data using K-Means clustering, Support Vector Machines (SVM), KNN, Decision Trees, Naive Bayes, and PCA

## Requirements :-

- => Basic knowledge of python programming
- => A system with a stable internet connection
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Machine Learning Module 1 :

- ~ Introduction machine learning module 1 Preview
- ~ Supervised, unsupervised, semi-supervised, reinforcement
- ~ Train, test, validation split
- ~ Performance
- ~ Overfitting, underfitting
- ~ OLS
- ~ Linear regression
- ~ polynomial regression
- ~ Assumptions R-square adjusted, R-square intro to Scikit-learn, training methodology, hands-on linear regression, ridge regression, logistics regression, precision-recall

=> Machine Learning Module 2 :

- ~ Decision tree, decision tree regressor, cross-validation Preview
- ~ Bias vs variance, ensemble approach, Bagging, boosting
- ~ Random forest, stacking, variable importance
- ~ XGBoost, hands-on XGBoost, gradient boost, ada boost

=> Machine Learning Module 3 :

- ~ K Nearest Neighbour, k-NN regressor, lazy learners, the curse of dimensionality, k-NN issues

=> Machine Learning Module 4 :

- ~ K-means, hierarchical clustering, DBSCAN
- ~ Performance measurement, principal component analysis, dimensionality reduction

=> Machine Learning Module 5 :

- ~ Naive Bayes SVM
- ~ Anomaly detection

=> Time series :

- ~ Arima, Sarima, Auto Arima
- ~ Time series using RNN LSTM, prediction of NIFTY stock price

# Interview ready DSA course in Python Level 2

---

Topic Name : DATA STRUCTURE

Sub-topic Name : MAANG INTERVIEW PREPARATION

Course link : <https://ineuron.ai/course/Interview-ready-DSA-course-in-Python-Level-2>

## Course Description :-

A comprehensive chase to excel any interview for the Data Structures and Algorithms. The course comprises 250 curated data structures problems grounded on personal interview experiences of students of preceding batches. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

## Course Features :-

- => Downloadable resources
- => Roadmap
- => Quizzes
- => Interview Questions
- => Resume Preparation
- => Completion Certificate
- => Various Assignments

## What you will learn :-

- => Basic and Complex data structures and algorithm concepts starting from linked list, array continuing to graph, trees, and ending on advanced programming concepts like dynamic programming, greedy algorithm, etc.
- => An optimised approach to solving complex problems.
- => Competitive programming from basic to advanced level.
- => Deep understanding of the implemented approach.
- => What data structure to use for what questions?
- => During the course journey, you can see yourself evolving and overcoming the usual mistakes you might have previously committed.
- => This course will help you initiate and work in the right direction for different projects that you can mention in your resume.
- => End to end understanding of the solved questions.
- => Solve questions with the least time complexity.
- => Most importantly, you will learn to face the most challenging problem-solving interviews.

## Requirements :-

- => Understanding of python programming language
- => A system with a decent internet connection

## Instructors :-

=> Akarsh Jaiswal :

~ A meticulous software engineer and impeccable educator, Akarsh Jaiswal has been supporting the community in various forms. He is currently working as a Full Stack Developer in a MNC. Being a software developer and data structures scholar himself, he has supported multiple startups and trained innumerable students, helping them shape their careers in IT industry. His main forte lies in competitive programming which has led him to win number of prestigious hackathons worldwide. Talking of the tech, he is well versed with Python & Java and has many running projects. He is a highly motivated individual and believes that every learning in Computer Science has the potential to open infinite avenues.

## Curriculum details :-

- => Recursion :
  - ~ Why do we need recursion?
  - ~ What is a recursive tree?
  - ~ Basic algorithms for recursion, how should we break the input? What is a base condition?
  - ~ Fibonacci series and difference between recursive and iterative method
  - ~ Print numbers from 1 to n using recursion
  - ~ Print numbers from n to 1 using recursion
  - ~ Factorial of a number
  - ~ Reverse an array, string using recursion
  - ~ Reverse a stack using recursion
  - ~ Sort an array using recursion
  - ~ Tower of Hanoi problem
  - ~ Generating all subsets/powersets
  - ~ Generating all unique subsets/powersets
  - ~ Generating all permutation with spaces
  - ~ Generating all permutation with case change
  - ~ Power of Two (Recursion/Iterative)
  - ~ Reverse a given number for eg. 634 should be 436

- ~ Letter case Permutation
- ~ Check if number is palindrome or not ?
- ~ Check if a number is Armstrong or not?

=> Linked List :

- ~ What is a Linked List?
- ~ Advantages/Disadvantages
- ~ Properties
- ~ What is a node, its structure?
- ~ Types of Linked List
- ~ Making a linked list
- ~ Insertion Singly Linked List at start
- ~ Deletion Singly Linked List from start
- ~ Insertion Singly Linked List at middle
- ~ Deletion Singly Linked List from middle
- ~ Insertion Singly Linked List at end
- ~ Deletion Singly Linked List from end
- ~ Print a Linked List
- ~ Length of Linked List iterative
- ~ Search in a linked list
- ~ Delete a Linked List
- ~ Length of a linked list recursive
- ~ Insertion doubly Linked List at start
- ~ Deletion doubly Linked List from start
- ~ Insertion doubly Linked List at middle
- ~ Deletion doubly Linked List from middle
- ~ Insertion doubly Linked List at end
- ~ Deletion doubly Linked List from end
- ~ Insertion Circular Linked List at start
- ~ Deletion Circular Linked List from start
- ~ Insertion Circular Linked List at middle
- ~ Deletion Circular Linked List from middle
- ~ Insertion Circular Linked List at end
- ~ Deletion Circular Linked List from end
- ~ Merge two sorted Linked List
- ~ Reverse a Linked List iterative
- ~ Reverse a Linked List recursive
- ~ Loop in linked list using hashing
- ~ Loop in linked list using tortoise and hare algorithm
- ~ Start of the loop
- ~ Length of loop
- ~ Middle of Linked List
- ~ Check whether a linked list has repeated elements or not?
- ~ Remove duplicates from a sorted linked list
- ~ Remove duplicates from an unsorted linked list
- ~ Delete a Linked List node without head pointer
- ~ Count of every element in a linked list
- ~ Check the length of the Linked List is even or odd without counting nodes.
- ~ The intersection of Two Linked List using stack
- ~ The intersection of Two Linked List without extra memory
- ~ Find Nth Node from End of Linked list
- ~ Reverse a doubly linked list recursive
- ~ Reverse a doubly linked list iterative

=> Trees :

- ~ What is a tree?
- ~ What are a node and its structure?
- ~ What is a Binary Tree
- ~ What is a Binary Search Tree
- ~ Types of trees?
- ~ Construction of a Tree?
- ~ Insertion of a node
- ~ Inorder traversal iterative
- ~ Preorder traversal iterative
- ~ Postorder traversal iterative
- ~ Preorder traversal recursive
- ~ Inorder traversal recursive
- ~ Postorder traversal recursive
- ~ Level order traversal
- ~ Level order traversal line by line
- ~ Vertical order traversal
- ~ Search in a tree
- ~ Height of tree
- ~ Number of nodes
- ~ Number of leaf nodes
- ~ Sum of all nodes
- ~ Inorder successor
- ~ Inorder Predecessor
- ~ Delete a Binary Tree
- ~ Number of nodes at nth level
- ~ Number of leaf nodes, internal nodes, total node if height is h

=> Sorting :

- ~ Bubble
- ~ Selection
- ~ Insertion
- ~ Merge
- ~ Quick
- ~ Counting



=> Searching :

- ~ Linear search
- ~ Binary search
- ~ Order not known Search
- ~ Ternary Search
- ~ First and last occurrence of an element
- ~ Count of element in a sorted array
- ~ Floor/ceil of an element in array
- ~ Next permutation
- ~ Searching in infinite sorted array
- ~ Index of first one in a binary sorted array
- ~ Find Pivot in rotated and sorted array
- ~ Search in rotated and sorted array
- ~ Find the number of 1s in a sorted binary array

=> Heap :

- ~ What is a heap
- ~ Types of Heap
- ~ Implementing Heap
- ~ Heap operations
- ~ Heap Sort
- ~ MinHeap Implementation
- ~ Time & space complexity
- ~ Kth smallest element
- ~ Kth largest element
- ~ K largest and smallest elements
- ~ K closest numbers map
- ~ Implementing Max & Min Heap which is not atomic
- ~ K closest numbers Heap without map
- ~ Top k frequent numbers
- ~ Frequency Sort
- ~ K closest points to origin

=> Stack & Queue :

- ~ What is a stack?
- ~ Implementing stack and its operations
- ~ Implementing queue and its operations
- ~ Types of queue
- ~ Parenthesis checker
- ~ Next greater to right
- ~ Next greater to left
- ~ Next smaller to right
- ~ Next smaller to left
- ~ Stack using queue
- ~ Queue using stack
- ~ Implementing Stack using linked list
- ~ Implementing Queue using linked list

=> Greedy Approach :

- ~ What is a greedy approach?
- ~ N meeting in one room
- ~ Activity Selection
- ~ Greedy algorithm to find the minimum number of coins
- ~ Fractional Knapsack Problem
- ~ Minimum number of platforms required for a railway
- ~ Job sequencing Problem

=> Array :

- ~ What is an array
- ~ Find the minimum and maximum element in an array
- ~ Segregate 0's and 1's in an array
- ~ Sort the array of 0s, 1s, and 2s
- ~ Reverse the given input array
- ~ Find a most frequent element in an array
- ~ Find non repeated elements in an array of integers
- ~ Find duplicate elements in an array
- ~ Find the first repeating element in an array of integers
- ~ Find the first non-repeating element in an array of integers
- ~ Find unique features in an array
- ~ Check whether the given array is sorted or not.
- ~ Move all the negative elements to one side of the array
- ~ Merge two sorted arrays to form a single array
- ~ Find the missing number in an integer array of 1 to 100
- ~ Remove duplicates from an array of integer
- ~ Move all 0s to the end of the array
- ~ What is a two-dimensional array
- ~ nth row/col of a matrix
- ~ Diagonal of a matrix
- ~ Upper lower triangular matrix
- ~ Transpose of a Matrix

=> String :

- ~ What is a string
- ~ Reverse a string
- ~ Find a maximum occurring character in a given string
- ~ Remove a given character from the string
- ~ Print duplicate characters of the given string.
- ~ Remove all duplicates from the given string.
- ~ Check if a string is a substring of another or not.

- ~ Check if two strings are rotation of each other or not
- ~ check if two given string is an anagram of each other
- ~ Find first non-repeating character of the given string
- ~ Generate all substrings of a string
- ~ Check if the given string is palindrome or not
- ~ Check if the given string has all unique characters
- ~ Check if one string is a permutation of the other or not
- ~ Write a function to perform basic string compression, e.g. aabcccccaaa would become a2b1c5a3
- ~ Write a function to perform basic string expansion, e.g. a2b1c5a3 would become aabcccccaaa

=> Graphs :

- ~ What is a graph data structure and its examples ?
- ~ Order and degree of a graph
- ~ Types of graph
- ~ Classes of graph
- ~ Representation of graph
- ~ Implementation of graph
- ~ BFS Traversal
- ~ BFS Traversal for disconnected graph
- ~ DFS Traversal
- ~ DFS Traversal for disconnected graph

=> Time Complexity :

- ~ What is time complexity?
- ~ Sample problem 1
- ~ Sample problem 2
- ~ Sample problem 3
- ~ Sample problem 4
- ~ Sample problem 5
- ~ Sample problem 6
- ~ Time complexity of recursive functions

=> Space Complexity :

- ~ What is space complexity?
- ~ Sample problem 1
- ~ Sample problem 2
- ~ Sample problem 3
- ~ Sample problem 4
- ~ Sample problem 5
- ~ Space complexity of fibonacci series problem

=> Hashing :

- ~ What is hashing?
- ~ Two sum problem
- ~ Find all symmetric pairs.

=> Maths :

- ~ Prime number
- ~ Factors of a number
- ~ GCD
- ~ LCM
- ~ Trailing zeroes in a factorial
- ~ Search in a sorted 2D matrix
- ~ Power function
- ~ Majority element
- ~ Grid Unique Paths

=> Dynamic Programming :

- ~ What is DP?
- ~ Fibonacci series
- ~ 0-1 Knapsack Recursive
- ~ 0-1 Knapsack Memorized
- ~ 0-1 Knapsack Tabular
- ~ Subset sum problem
- ~ Equal sum partition problem
- ~ Count subset with given sum
- ~ Unbounded Knapsack
- ~ Coin change max ways
- ~ Coin change min ways
- ~ Longest common subsequence Recursive
- ~ Longest common subsequence Memorized
- ~ Longest common subsequence Tabular
- ~ Printing LCS
- ~ Longest common substring  $O(n^2)$
- ~ Shortest common subsequence
- ~ Kadane's algorithm

=> Bit Manipulation :

- ~ Introduction to Bit Manipulation
- ~ Bitwise operators
- ~ Bitwise operators Implementation
- ~ Power of Two
- ~ Number of 1 Bits or Hamming Weight
- ~ Missing Number in array
- ~ Find element which appears once in an array while all other appears twice

# SQL Server Integration Services

---

Topic Name : DATABASE

Sub-topic Name : MSSQL

Course link : <https://ineuron.ai/course/SQL-Server-Integration-Services>

## Course Description :-

SSIS is an enterprise-level Extract, Transform and Load (ETL) Development tool, SSIS is one of the most powerful application for moving data in and out of various databases and files. This course provides developers with a thorough knowledge in developing SSIS Packages with SQL Server 2019.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Work with various ControlFlow Activities
- => Data Migration Techniques
- => Make Dynamic SSIS Packages
- => Work with various DataFlow Transformations

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Introduction :
  - ~ Introduction to SSIS Preview
- => Installation :
  - ~ Downloading sql server
  - ~ Installing visual studio
- => Creating simple Extract and load :
  - ~ Running ssms and ssis
  - ~ Looking around ssis
  - ~ Data flow task Preview
  - ~ Connection manager
  - ~ Ole db source task
  - ~ Ole db destination
  - ~ Data viewer
- => Adding a bit of Transforming :
  - ~ Implicit conversions
  - ~ Data conversion
  - ~ Derived column task
  - ~ Aggregation Preview
  - ~ MultiCast
  - ~ Conditional Split
  - ~ Error Handling
  - ~ Row Sampling and Percentage Sampling
  - ~ Sort
  - ~ Adding an Excel Source
  - ~ Add Text Source and Excel Destination part 1
  - ~ Add Text Source and Excel Destination part 2
  - ~ Union ALL
  - ~ Merge and solving Mapping Problems
  - ~ Add different Destinations

=> Adding control Flow Tasks :

- ~ Adding Variables and Row count
- ~ Script component part 1
- ~ Script component part 2

# Automatic Number Plate Recognition

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/Automatic-Number-Plate-Recognition>

## Course Description :-

In the following project, we will understand how to recognize License number plates using Python. We will utilize OpenCV for this project in order to identify the license number plates and the Paddle OCR for the characters and digits extraction from the plate. We will create a web app with a Flask framework that automatically recognizes the License Number Plate .

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access
- => Course Materials
- => Assignments

## What you will learn :-

- => Real Time Projects
- => Automatic Number Plate Recognition
- => Object detection using tensorflow
- => Paddle OCR
- => Modular Coding Techniques
- => Learn about AWS basics along with CICD tools like Github actions for production-grade deployment
- => Flask web framework

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- Automatic Number Plate Recognition :
  - ~ Introduction of Instructor
  - ~ Project Overview
  - ~ Application Tour
  - ~ Jupyter Notebook Walkthrough
  - ~ Tour to Architecture diagram
  - ~ Folder Structure overview
  - ~ Environment and Project Setup
  - ~ Data Ingestion
  - ~ Data Transformation
  - ~ Prepare Base Model
  - ~ Model Training
  - ~ Model Pusher
  - ~ Training Pipeline
  - ~ Prediction Pipeline
  - ~ Frontend app development
  - ~ Running project locally
  - ~ Running project using Docker
  - ~ Tour to the cloud and Service Overview (AWS)
  - ~ IAM setup
  - ~ ECR setup
  - ~ EC2 setup

- ~ *Self hosted runner*
- ~ *Assignments Discussion*
- ~ *End Notes*

# Python advance with projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON PROJECT

Course link : <https://ineuron.ai/course/Python-advance-with-projects>

## Course Description :-

Throughout this course, you will learn everything you need to know about Python, from the basics to advanced topics. Python applications like download manager will be developed using advanced principles to help you become a professional programmer capable of landing well-paying employment.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => N
- => u
- => l
- => l

## Requirements :-

- => Basic knowledge of Python programming
- => A system with stable internet connection
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Introduction :

- ~ Programming language overview Preview
- ~ Installation (tools: sublime, vscode, pycharm, anaconda, atom, jupyter notebook, kite) Preview
- ~ Virtual environment
- ~ Why python

=> Python Basic :

- ~ Introduction of python and comparison with other programming language
- ~ Installation of anaconda distribution and other python ide
- ~ Python objects, number & Booleans, strings.
- ~ Container objects, mutability of objects
- ~ Operators - arithmetic, bitwise, comparison and assignment operators, operators precedence and associativity
- ~ Conditions (if else, if-elif-else), loops (while, for)
- ~ Break and continue statement and range function

=> String Objects :

- ~ Basic data structure in python
- ~ String object basics
- ~ String inbuilt methods
- ~ Splitting and joining strings
- ~ String format functions

=> List Object Basics :

- ~ List methods
- ~ List as stack and queues
- ~ List comprehensions

=> Tuples, Set, Dictionaries & Its Function :

- ~ Dictionary object methods
- ~ Dictionary comprehensions
- ~ Dictionary view objects
- ~ Functions basics, parameter passing, iterators.
- ~ Generator functions
- ~ Lambda functions
- ~ Map, reduce, filter functions.

# Enterprise Java with Spring Boot

---

Topic Name : PROGRAMMING

Sub-topic Name : JAVA

Course link : <https://ineuron.ai/course/Enterprise-Java-with-Spring-Boot>

## Course Description :-

Java is one of the most widely used programming languages, owing to its versatility and compatibility. Java can be used for various purposes, including software development, mobile application development, and large-scale system development. This Java course will teach you all you need to know to get started with Java.

## Course Features :-

- => Early Launch Offer
- => 6 months programme
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate
- => Course resources

## What you will learn :-

- => Java Basics
- => Loops
- => OOP
- => Java Projects
- => Exception Handling
- => Blockchain Project
- => MultiThreading
- => Collection Framework
- => Junit
- => MySQL
- => NoSQL
- => JDBC
- => Hibernate

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hyder Abbas :

~ Corporate Software Development Trainer with a demonstrated track record of success in the IT and Ed-tech industries. I started my career as a software developer and have since taught Java, Python, Javascript to hundreds of IT enthusiasts, including corporate professionals, throughout the years. I have been developing software for over 6 years.

=> Navin Reddy :

~ I am Corporate Java trainer. Since past few years successfully trained many professionals at JP Morgan, Accenture, Polaris and L&T infotech. My youtube channel "Telusko" presently has 1.7 million subscribers. Passionate about Java Technology for over a decade and moved on as a corporate trainer. I am certified blockchain developer and Currently, building Applications running on Blockchain (dapps).

=> Nitin M :

~ I began working for a multinational corporation as a developer, but teaching has always been my passion. I shifted to education technology and have five years of expertise instructing both college freshmen and Corporate Employees. My interests include Java, JEE, and frameworks, and I have developed numerous applications using SpringBoot and microservices. Last but not least, I want to code as well as teach and continue to teach forever.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Introduction
- => Java Basics :
  - ~ Introduction to Java
  - ~ Setup



- ~ *Getting Started*
- ~ *How Java Works*
- ~ *Variables*
- ~ *Data Types*
- ~ *Naming Conventions*
- ~ *Operators*
- ~ *Conditional Statements*
- ~ *If Else*
- ~ *Ternary*
- ~ *Switch*

=> **Loops :**

- ~ *While*
- ~ *For*
- ~ *Nested Loops*
- ~ *Break and Continue*

=> **OOP :**

- ~ *Introduction to Object Oriented Programming*
- ~ *Class and Object*
- ~ *Constructor*
- ~ *This Keyword*
- ~ *Method and Constructor Overloading*
- ~ *Static Keyword*
- ~ *Inner Class*
- ~ *Arrays*
- ~ *Enhanced for loop*
- ~ *Varargs*
- ~ *Inheritance*
- ~ *Super Method*
- ~ *Multiple Inheritance*
- ~ *Method Overriding*
- ~ *Super Keyword*
- ~ *Multiple Inheritance*
- ~ *Encapsulation*
- ~ *Wrapper class | Autoboxing*
- ~ *Abstract Keyword*
- ~ *Final Keyword*
- ~ *Interface*
- ~ *Anonymous Inner class*
- ~ *Functional Interface*
- ~ *Lambda Expression*
- ~ *Package*
- ~ *Access Modifier*

=> **Java Projects :**

- ~ *Number Guessing Game*
- ~ *Java Object to JSON Converter*

=> **Exception Handling :**

- ~ *Exceptions*
- ~ *Try Catch*
- ~ *Finally*
- ~ *Multiple Catch Blocks*
- ~ *Checked Exceptions*
- ~ *User Defined Exception*

=> **Blockchain Project :**

- ~ *Building Blockchain using Java*

=> **MultiThreading :**

- ~ *Introduction*
- ~ *Thread Class*
- ~ *Runnable*
- ~ *Lambda Expression*
- ~ *Thread Priority*
- ~ *Synchronized*

=> **Project :**

- ~ *Chatting Application*

=> **Collection Framework :**

- ~ *Introduction*
- ~ *Iterator and List Interface*
- ~ *Set*
- ~ *Map*
- ~ *Comparator, Comparable*
- ~ *Generics*

=> **Projects :**

- ~ *Online Voting System*
- ~ *Work Scheduler*

=> **Junit :**

- ~ *Introduction*
- ~ *Unit Testing*
- ~ *Test Exceptions*
- ~ *Multiple Assertions*
- ~ *Annotations*

=> **Junit Project :**

- ~ *Test Cases for Online Voting System*

## => MySQL :

- ~ *Introduction to SQL*
- ~ *Setup*
- ~ *What is Database*
- ~ *Creating, Dropping Databases*
- ~ *Introduction to Tables*
- ~ *Creating, Dropping, Altering the Tables*
- ~ *CRUD*
- ~ *Select, Insert, Update, delete Queries*
- ~ *Where, groupby, having*
- ~ *Aggregate Functions*
- ~ *One to many, many to one, many to many relationships*

## => Sql with Java Projects :

- ~ *Indexing for Blockchain Project*
- ~ *Online Course Management*

## => NoSQL :

- ~ *Introduction to NoSQL*
- ~ *Categories*
- ~ *NoSQL vs RDBMS*
- ~ *Couch DB*
- ~ *Mongo DB*
- ~ *Cassandra*
- ~ *Redis*

## => NoSql with Java Projects :

- ~ *IPL Stats*

## => JDBC :

- ~ *Introduction*
- ~ *CRUD Operations*
- ~ *ResultSet*
- ~ *Connection Pooling*

## => JDBC Project :

- ~ *Book My Calendar*

## => Hibernate :

- ~ *Introduction to Hibernate*
- ~ *Setup*
- ~ *Configuration File*
- ~ *SQL Property*
- ~ *Annotation*
- ~ *CRUD*
- ~ *Embeddable Object*
- ~ *Mapping Relations*
- ~ *EAGER LAZY*
- ~ *Caching*
- ~ *HQL*
- ~ *Object States Persistence Life Cycle*
- ~ *Get vs Load*
- ~ *JPA*

## => Hibernate Project :

- ~ *Hall of Fame*

## => Servlets and JSP :

- ~ *Introduction*
- ~ *Setup*
- ~ *Creating First project*
- ~ *Creating Servlet and XML*
- ~ *Get vs Post*
- ~ *RequestDispatcher*
- ~ *HttpServletRequest and HttpServletResponse*
- ~ *RequestDispatcher and sendRedirect*
- ~ *HttpSession and Cookie*
- ~ *ServletConfig and ServletContext*
- ~ *Servlet Annotation Configuration*
- ~ *JSP*
- ~ *JSP to Servlet*
- ~ *Tags, Scriptlet, Declaration, Directive, Expression*
- ~ *Implicit Objects*
- ~ *Exception handling in JSP*
- ~ *JDBC in JSP*
- ~ *Servlet Filters*

## => JSP Project :

- ~ *Stock Broker*

## => Spring :

- ~ *Introduction*
- ~ *Dependency Injection*
- ~ *BeanFactory*
- ~ *ApplicationContext*
- ~ *Spring Container*
- ~ *Singleton vs Prototype*
- ~ *Setter Injection*
- ~ *Ref Attribute*
- ~ *Constructor Injection*
- ~ *Autowire*

- ~ *Primary Bean*
- => Spring Boot :
  - ~ *Introduction*
- => Spring MVC :
  - ~ *Introduction*
  - ~ *Creating Controller*
  - ~ *Accepting User Input*
  - ~ *@RequestParam*
  - ~ *@ModelAttribute*
  - ~ *Prefix and Suffix*
  - ~ *Model and ModelAndView*
  - ~ *ModelAttribute*
  - ~ *GetMapping and PostMapping*
- => Spring Project :
  - ~ *Quiz Application*
- => Spring ORM :
  - ~ *Introduction*
  - ~ *Spring Hibernate Config*
  - ~ *MySQL and DAO*
  - ~ *DAO Creation*
  - ~ *Add and Fetch*
- => Spring Data JPA :
  - ~ *Spring Data JPA Configuration*
  - ~ *JpaRepository*
  - ~ *JpaRepository Add and Fetch*
  - ~ *Query DSL*
  - ~ *Query Annotation*
- => Spring REST :
  - ~ *Introduction*
  - ~ *REST GetMapping*
  - ~ *Jackson*
  - ~ *PathVariable*
  - ~ *RestController*
  - ~ *PostMapping*
  - ~ *Jackson XML*
  - ~ *Produces Attribute*
  - ~ *RequestBody and Consumes Attribute*
- => Spring AOP :
  - ~ *Why AOP*
  - ~ *AOP Terms*
  - ~ *Aspect and Before Annotation*
  - ~ *Logger*
  - ~ *After Finally*
  - ~ *AfterReturning and Throwing*
- => Spring Security :
  - ~ *Introduction*
  - ~ *Implementation*
  - ~ *Managing Users*
  - ~ *Passwords*
  - ~ *Authentication*
  - ~ *Authorization*
  - ~ *CSRF and CORS*
  - ~ *OAuth2*
  - ~ *JWT*
- => Spring Mega Project :
  - ~ *Secure Stock Broker App*
- => Agile and Scrum (Optional) :
  - ~ *Agile Values*
  - ~ *12 Agile Principles*
  - ~ *Scrum Overview*
  - ~ *Scrum Values*
  - ~ *Concept of Sprints*
  - ~ *Scrum Roles*
  - ~ *Role of Scrum Master*
  - ~ *Role of Product Owner*
  - ~ *Role of Development Team*
  - ~ *Daily Stand-up*
  - ~ *Sprint Planning*
  - ~ *Sprint Review*
  - ~ *Sprint Retrospective*
  - ~ *Backlog Refinement*
  - ~ *User Stories*
  - ~ *Product Backlog and Sprint Backlog*
  - ~ *Working Agreements*
  - ~ *Definition of Ready, Done*
  - ~ *Team Velocity*
  - ~ *Burndown Chart*
- => Docker Installation Basics :
  - ~ *What is Docker?*
  - ~ *How to install Docker and Hello World*
  - ~ *What is container in Docker*

- ~ Docker vs Virtual Machine
- ~ First interaction with busy box image

=> Fundamentals of docker :

- ~ Docker lifecycle and PS
- ~ Start and delete a container
- ~ Getting a mongodb container for fun
- ~ Exploring exec command
- ~ Multiple ways to get inside a container

=> Custom Docker images :

- ~ Analogy for custom docker image
- ~ Our first base image and custom image
- ~ Behind the scene for custom image
- ~ Creating a custom mongodb image
- ~ Concept of caching in docker
- ~ Provide a custom name for your image

=> Project and Docker :

- ~ Introduction to node project for docker
- ~ Introduction to node project for docker part 2
- ~ Containerize a node application
- ~ Performance upgrade in node project container

=> Multi container setup :

- ~ Introduction to multi docker container
- ~ A mini mongo connector project
- ~ Put your node code in a container
- ~ Introduction to docker compose
- ~ Connect 2 compose images in docker
- ~ Access the compose container app with browser

=> Ngnix - production grade deployment :

- ~ Ngnix A production grade docker
- ~ Attaching volumes in Docker
- ~ Types of docker files
- ~ Dev test and production stages
- ~ Understand react project for docker deployment
- ~ Docker for development
- ~ Docker for testing
- ~ Docker for production

=> Docker AWS and Travis CI :

- ~ Docker CI and AWS
- ~ What is CI-CD Jenkins vs Travis CI
- ~ Moving to AWS Elastic Beanstalk
- ~ Moving project to GitHub repo
- ~ Reading Travis CI documentation
- ~ Writing our 1st Travis CI config file
- ~ AWS IAM user generation
- ~ Elastic Beanstalk and S3 bucket
- ~ Finally hosting app on AWS with CI integrated with docker
- ~ TURN OFF those AWS apps

=> Microservices :

- ~ Introduction to Microservices
- ~ Microservice Architecture
- ~ Spring Cloud

# Full Stack Web Development with Python in Hindi Tech Neuron

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON

Course link : <https://ineuron.ai/course/Full-Stack-Web-Development-with-Python-in-Hindi-Tech-Neuron>

## Course Description :-

Full Stack Python with Django is specially created to fulfil the standards set by the industry. You may learn a lot about Python, the Django REST framework, Django Models, React, and other topics in this lengthy online interactive course. With the Python Django program, you will also gain proficiency with the web framework while working on practical use cases.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate
- => 18 hr live support all seven day
- => Extra doubt clearing as per demand
- => hindi
- => Doubt clearing
- => 300+ practice problems

## What you will learn :-

- => Python basic
- => List object basics
- => String objects
- => Tuples
- => Functions
- => Memory management
- => OOps concepts
- => Exception Handling
- => Files
- => DATABASE
- => Web Development using Python on Django
- => React
- => Web API

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Saurabh Shukla :

~ Saurabh Shukla has been educating children with this credo, and he does so for free. MySirG.com, the educator's YouTube channel, features video lessons on programming languages. Saurabh has successfully reached thousands of students around the country by making it his aim to produce subject-related videos on a daily basis.

## Curriculum details :-

- => Course introduction :
  - ~ course overview and dashboard description
  - ~ Programming language overview
  - ~ History of Python
  - ~ Version History of Python
  - ~ Installation (tools:vscode, IDLE)
  - ~ Learning Path through Python

## => Python basic :

- ~ *Developing and executing Python program*
- ~ *Python Virtual Machine*
- ~ *Comment, data, variables, types*
- ~ *heap space and namespace*
- ~ *id, keywords, import, module, help*
- ~ *print, input*
- ~ *number system*
- ~ *Conversion functions*
- ~ *Operators - arithmetic, bitwise, comparison and assignment operators, operators precedence and associativity*
- ~ *Decision Control Statements*
- ~ *Iterative control Statements - while loop*
- ~ *transfer control statements- break, continue, pass*
- ~ *for loop*
- ~ *range*

## => List object basics :

- ~ *list*
- ~ *list methods*
- ~ *List comprehensions*
- ~ *packing unpacking*
- ~ *built-in methods*

## => String objects :

- ~ *String object basics*
- ~ *String methods*
- ~ *Splitting and joining strings*
- ~ *String format functions*

## => Tuples :

- ~ *tuple basics*
- ~ *tuple methods*
- ~ *list vs tuple*

## => set :

- ~ *set basics*
- ~ *set comprehensions*
- ~ *set methods*

## => dict :

- ~ *dictionary basics*
- ~ *dict methods*
- ~ *dict comprehensions*

## => Functions :

- ~ *Function Basics*
- ~ *Ways to define a function*
- ~ *function returning None*
- ~ *default arguments*
- ~ *positional vs keyword arguments*
- ~ *variable number of arguments*
- ~ *Recursion*
- ~ *Lambda functions*
- ~ *Iterators*
- ~ *Generator functions*
- ~ *Decorators*
- ~ *Map, reduce, filter functions.*

## => Memory management :

- ~ *Multithreading*
- ~ *Multiprocessing*

## => OOps concepts :

- ~ *oops basic concepts.*
- ~ *Main aspects of OOP*
- ~ *Encapsulation and Abstraction*
- ~ *Classes and Objects*
- ~ *init method*
- ~ *Types of variables*
- ~ *Types of functions*
- ~ *Inheritance*
- ~ *Name conflict issues*
- ~ *Polymorphism*
- ~ *Overriding*
- ~ *Operator Overloading*

## => Exception Handling :

- ~ *Introduction to Exceptions*
- ~ *Exceptions handling with try-except*
- ~ *use of else and finally*
- ~ *Defining exception*

## => Files :

- ~ *Working with files*
- ~ *Reading and writing files*
- ~ *Buffered read and write*
- ~ *Other file methods.*
- ~ *Renaming a file*
- ~ *Removing a file*

## => DATABASE :

~ *Postgres Basics*

=> **Project :**

~ *Core Python + DB*

~ *Web Project-1*

~ *Web Project-2*

=> **Web Development using Python on Django :**

~ *Client Server Architecture*

~ *HTML*

~ *CSS*

~ *JS*

~ *Introduction to Django*

~ *Creating first Django Project*

~ *Creating web application*

~ *Defining views in an application*

~ *Multiple views*

~ *Multiple Application*

~ *Application level url configuration*

~ *Template*

~ *Template variables*

~ *Template Tags*

~ *Template Filters*

~ *Custom Template Tags and Filters*

~ *Database Introduction*

~ *SQLite DB*

~ *Models*

~ *makemigrations*

~ *migrate*

~ *Admin app*

~ *CRUD operations*

~ *session tracking*

~ *login functionality*

=> **React :**

~ *React Essentials*

=> **Web API :**

~ *What is web API*

~ *Difference b/w API and web API*

~ *Restful services*

~ *POSTMAN*

# Python Coding Interview Preparation

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Python-Coding-Interview-Preparation>

## Course Description :-

This course is designed mostly for Python Coding test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Python Theoretical Test

=> Python Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Python Coding Questions :

- ~ Python Test 1
- ~ Python Test 2
- ~ Python Test 3
- ~ Python Test 4
- ~ Python Test 5
- ~ Python Test 6
- ~ Python Test 7
- ~ Python Test 8
- ~ Python Test 9
- ~ Python Test 10
- ~ Python Test 11
- ~ Python Test 12
- ~ Python Test 13
- ~ Python Test 14
- ~ Python Test 15
- ~ Python Test 16
- ~ Python Test 17
- ~ Python Test 18
- ~ Python Test 19
- ~ Python Test 20



# MLflow

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : <https://ineuron.ai/course/MLflow>

## Course Description :-

This program is meant to give you the basics till an advanced understanding of one of the most popular ML cycle management frameworks - MLflow. Course curriculum includes concepts about the MLflow framework, tutorials, and much more!

## Course Features :-

- => Learning of different concepts of MLflow to manage ML lifecycle
- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Basic concepts
- => Installation
- => MLflow use cases
- => Apply MLflow to track your ML/DL experiments, Package Project, Serve model, and manage and register model

## Requirements :-

- => Interest to learn
- => ML and DL course
- => Machine Learning and Deep learning concepts
- => Decent internet connection

## Instructors :-

=> Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

=> Section: 1 MLflow Introduction :

- ~ Introduction to MLflow
- ~ Installation and first trial of MLflow

=> Section: 2 MLflow Tracking :

- ~ Simple ML model
- ~ Logging our simple ML model using
- ~ Exploring UI of MLflow
- ~ Packaging a project MLflow way

# Full Stack Data Science Nov'21 Batch

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science-Nov'21-Batch>

## Course Description :-

This is a data science full stack live mentor led certification program along with full time one-year internship provided by iNeuron intelligence private limited, where you will learn all the stack required to work in data science, data analytics and big data industry including ML ops and cloud infrastructure and real time industry project and product development along with iNeuron product development team and you will contribute on various level with iNeuron .

## Course Features :-

- => Full stack Data Science masters certification
- => Job guarantee otherwise refund
- => One year of internship Anytime
- => Online Instructor-led learning: Live teaching by instructors
- => 56 + hands-on industry real-time projects.
- => 500 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Doubt clearing through mail and skype support team
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building Anytime
- => Career guidance Anytime
- => Interview Preparation Anytime
- => Regular assessment
- => Job Fair and Internal Hiring
- => Mock Interview Anytime

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => ML ops
- => Cloud
- => Data structure and algorithm
- => Architecture
- => Domain wise project
- => Databases
- => Negotiations skills
- => Mock interview
- => Interview preparation
- => Resume building after every module

## Requirements :-

=> Dedication

=> Computer with i3 and above configuration

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course Introduction :

- ~ course overview and dashboard description
- ~ Introduction of data science and its application in day to day life
- ~ Programming language overview
- ~ Installation (tools: sublime, vscode, pycharm, anaconda, atom, jupyter notebook, kite)
- ~ Virtual environment
- ~ Why python

=> Python Basic :

- ~ Introduction of python and comparison with other programming language
- ~ Installation of anaconda distribution and other python ide
- ~ Python objects, number & Booleans, strings
- ~ Container objects, mutability of objects
- ~ Operators - arithmetic, bitwise, comparison and assignment operators, operators precedence and associativity
- ~ Conditions (if else, if-elif-else), loops (while, for)
- ~ Break and continue statement and range function

=> String Objects :

- ~ basic data structure in python
- ~ String object basics
- ~ String inbuilt methods
- ~ Splitting and joining strings
- ~ String format functions

=> List Object Basics :

- ~ List methods
- ~ List as stack and queues
- ~ List comprehensions

=> Tuples, Sets, Dictionaries & its Function :

- ~ Dictionary object methods
- ~ Dictionary comprehensions
- ~ Dictionary view objects
- ~ Functions basics, parameter passing, iterators
- ~ Generator functions
- ~ Lambda functions
- ~ Map, reduce, filter functions

=> Memory Management :

- ~ Multithreading
- ~ Multiprocessing

=> OOPs Concepts :

- ~ oops basic concepts.
- ~ Creating classes
- ~ Pillars of oops
- ~ Inheritance
- ~ Polymorphism
- ~ Encapsulation
- ~ Abstraction
- ~ Decorator
- ~ Class methods and static methods
- ~ Special (magic/dunder) methods
- ~ Property decorators - getters, setters, and deletes

=> Files :

- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other file methods
- ~ Logging, debugger
- ~ Modules and import statements

=> Exception Handling and Difference between Exception and Error :

- ~ Exceptions handling with try-except
- ~ Custom exception handling
- ~ List of general use exception
- ~ Best practice exception handling

=> GUI Framework :

- ~ What is desktop and standalone application
- ~ Use of desktop app
- ~ Examples of desktop app
- ~ Tinker
- ~ Kivy

=> Database :

- ~ SQLite
- ~ MySQL
- ~ Mongo dB
- ~ NoSQL - Cassandra

=> Web API :

- ~ What is web API
- ~ Difference b/w API and web API
- ~ Rest and soap architecture
- ~ Restful services

=> Flask :

- ~ Flask introduction
- ~ Flask application
- ~ Open link flask
- ~ App routing flask
- ~ Url building flask
- ~ Http methods flask
- ~ Templates flask
- ~ Flask project: food app
- ~ Postman
- ~ Swagger

=> Django :

- ~ Django introduction
- ~ Django project: weather app
- ~ Django project: memes generator
- ~ Django project: blog app
- ~ Django project in cloud

=> Stream Lit :

- ~ Stream lit introduction
- ~ Stream lit project structure
- ~ Stream lit project in cloud

=> Pandas Basic :

- ~ Python pandas - series
- ~ Python pandas data frame
- ~ Python pandas panel
- ~ Python pandas - basic functionality
- ~ Reading data from different file system

=> Pandas Advance :

- ~ Python pandas re indexing python
- ~ Pandas iteration
- ~ Python pandas sorting.
- ~ Working with text data options & customization
- ~ Indexing & selecting
- ~ Data statistical functions
- ~ Python pandas - window functions
- ~ Python pandas - date functionality
- ~ Python pandas time delta
- ~ Python pandas - categorical data
- ~ Python pandas visualization
- ~ Python pandas - iotools

=> Dask :

- ~ Dask Array
- ~ Dask Bag
- ~ Dask DataFrame
- ~ Dask Delayed
- ~ Dask Futures
- ~ Dask API
- ~ Dask SCHEDULING
- ~ Dask Understanding Performance
- ~ Dask Visualize task graphs
- ~ Dask Diagnostics (local)
- ~ Dask Diagnostics (distributed)
- ~ Dask Debugging
- ~ Dask Ordering

=> Python Numpy :

- ~ Numpy - ND array object.
- ~ Numpy - data types.
- ~ Numpy - array attributes.
- ~ Numpy - array creation routines.
- ~ Numpy - array from existing.
- ~ Data array from numerical ranges.
- ~ Numpy - indexing & slicing.
- ~ Numpy advanced indexing.
- ~ Numpy broadcasting.
- ~ Numpy - iterating over array.
- ~ Numpy - array manipulation.
- ~ Numpy - binary operators.
- ~ Numpy - string functions.
- ~ Numpy - mathematical functions.
- ~ Numpy - arithmetic operations.
- ~ Numpy - statistical functions.
- ~ Sort, search & counting functions.

- ~ *Numpy - byte swapping.*
- ~ *Numpy - copies & views.*
- ~ *Numpy - matrix library.*
- ~ *Numpy - linear algebra*

=> Visualization :

- ~ *Matplotlib*
- ~ *Seaborn*
- ~ *Cufflinks*
- ~ *Plotly*
- ~ *Bokeh*

=> Statistics Basic :

- ~ *Introduction to basic statistics terms*
- ~ *Types of statistics*
- ~ *Types of data*
- ~ *Levels of measurement*
- ~ *Measures of central tendency*
- ~ *Measures of dispersion*
- ~ *Random variables*
- ~ *Set*
- ~ *Skewness*
- ~ *Covariance and correlation*

=> Probability Distribution Function :

- ~ *Probability density/distribution function*
- ~ *Types of the probability distribution*
- ~ *Binomial distribution*
- ~ *Poisson distribution*
- ~ *Normal distribution (Gaussian distribution)*
- ~ *Probability density function and mass function*
- ~ *Cumulative density function*
- ~ *Examples of normal distribution*
- ~ *Bernoulli distribution*
- ~ *Uniform distribution*
- ~ *Z stats*
- ~ *Central limit theorem*
- ~ *Estimation*

=> Statistics Advance :

- ~ *a Hypothesis*
- ~ *Hypothesis testings mechanism*
- ~ *P-value*
- ~ *T-stats*
- ~ *Student t distribution*
- ~ *T-stats vs. Z-stats: overview*
- ~ *When to use a t-tests vs. Z-tests*
- ~ *Type 1 & type 2 error*
- ~ *Bayes statistics (Bayes theorem)*
- ~ *Confidence interval(ci)*
- ~ *Confidence intervals and the margin of error*
- ~ *Interpreting confidence levels and confidence intervals*
- ~ *Chi-square test*
- ~ *Chi-square distribution using python*
- ~ *Chi-square for goodness of fit test*
- ~ *When to use which statistical distribution?*
- ~ *Analysis of variance (anova)*
- ~ *Assumptions to use anova*
- ~ *Anova three type*
- ~ *Partitioning of variance in the anova*
- ~ *Calculating using python*
- ~ *F-distribution*
- ~ *F-test (variance ratio test)*
- ~ *Determining the values of f*
- ~ *F distribution using python*

=> Linear Algebra :

- ~ *linear algebra*
- ~ *Vector*
- ~ *Scaler*
- ~ *Matrix*
- ~ *Matrix operations and manipulations*
- ~ *Dot product of two vectors*
- ~ *Transpose of a matrix*
- ~ *Linear independence of vectors*
- ~ *Rank of a matrix*
- ~ *Identity matrix or operator*
- ~ *Determinant of a matrix*
- ~ *Inverse of a matrix*
- ~ *Norm of a vector*
- ~ *Eigenvalues and eigenvectors*
- ~ *Calculus*

=> Solving Stats Problem with Python

=> Stats Problem Implementation with Spacy

=> Introduction to Machine Learning :

- ~ *AI vs ml vs dl vs ds*
- ~ *Supervised, unsupervised, semi-supervised, reinforcement learning*
- ~ *Train, test, validation split*

- ~ Performance
- ~ Overfitting, under fitting
- ~ Bias vs variance

#### => Feature Engineering :

- ~ Handling missing data
- ~ Handling imbalanced data
- ~ Up-sampling
- ~ Down-sampling
- ~ Smote
- ~ Data interpolation
- ~ Handling outliers
- ~ Filter method
- ~ Wrapper method
- ~ Embedded methods
- ~ Feature scaling
- ~ Standardization
- ~ Mean normalization
- ~ Min-max scaling
- ~ Unit vector
- ~ Feature extraction
- ~ Pca (principle component analysis)
- ~ Data encoding
- ~ Nominal encoding
- ~ One hot encoding
- ~ One hot encoding with multiple categories
- ~ Mean encoding
- ~ Ordinal encoding
- ~ Label encoding
- ~ Target guided ordinal encoding
- ~ Covariance
- ~ Correlation check
- ~ Pearson correlation coefficient
- ~ Spearmans rank correlation
- ~ Vif

#### => Feature Selection :

- ~ Feature selection
- ~ Recursive feature elimination
- ~ Backward elimination
- ~ Forward elimination

#### => Exploratory Data Analysis :

- ~ Feature engineering and selection.
- ~ Analyzing bike sharing trends.
- ~ Analyzing movie reviews sentiment.
- ~ Customer segmentation and effective cross selling.
- ~ Analyzing wine types and quality.
- ~ Analyzing music trends and recommendations.
- ~ Forecasting stock and commodity prices

#### => Regression :

- ~ Linear regression
- ~ Gradient descent
- ~ Multiple linear regression
- ~ Polynomial regression
- ~ R square and adjusted r square
- ~ Rmse , mse, mae comparison
- ~ Regularized linear models
- ~ Ridge regression
- ~ Lasso regression
- ~ Elastic net
- ~ Complete end-to-end project with deployment on cloud and ui

#### => Logistics Regression :

- ~ Logistics regression in-depth intuition
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Hyper parameter tuning
- ~ Grid search cv
- ~ Randomize search cv
- ~ Data leakage
- ~ Confusion matrix
- ~ Precision, recall, f1 score ,roc, auc
- ~ Best metric selection
- ~ Multiclass classification in lr
- ~ Complete end-to-end project with deployment in multi cloud platform

#### => Decision Tree :

- ~ Decision tree classifier
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Confusion matrix
- ~ Precision, recall, f1 score ,roc, auc
- ~ Best metric selection
- ~ Decision tree repressor
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Performance metrics
- ~ Complete end-to-end project with deployment in multi cloud platform

#### => Support Vector Machines :

- ~ Linear svm classification
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Soft margin classification
- ~ Nonlinear svm classification
- ~ Polynomial kernel
- ~ Gaussian, rbf kernel
- ~ Data leakage
- ~ Confusion matrix
- ~ precision, recall, f1 score, roc, auc
- ~ Best metric selection
- ~ Svm regression
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Complete end-to-end project with deployment

#### => Naive Bayes :

- ~ Bayes theorem
- ~ Multinomial naive Bayes
- ~ Gaussian naive Bayes
- ~ Various type of Bayes theorem and its intuition
- ~ Confusion matrix
- ~ precision, recall, f1 score, roc, auc
- ~ Best metric selection
- ~ Complete end-to-end project with deployment

#### => Ensemble Technique and its Types :

- ~ Definition of ensemble techniques
- ~ Bagging technique
- ~ Bootstrap aggregation
- ~ Random forest (bagging technique)
- ~ Random forest regressor
- ~ Random forest classifier
- ~ Complete end-to-end project with deployment

#### => Boosting :

- ~ Boosting technique
- ~ Ada boost
- ~ Gradient boost
- ~ Xgboost
- ~ Complete end-to-end project with deployment

#### => Stacking :

- ~ Stacking technique
- ~ Complete end-to-end project with deployment

#### => KNN :

- ~ Knn classifier
- ~ Knn regressor
- ~ Variants of knn
- ~ Brute force knn
- ~ K-dimension tree
- ~ Ball tree
- ~ Complete end-to-end project with deployment

#### => Dimensionality Reduction :

- ~ The curse of dimensionality
- ~ Dimensionality reduction technique
- ~ Pca (principle component analysis)
- ~ Mathematics behind pca
- ~ Scree plots
- ~ Eigen-decomposition approach

#### => Clustering :

- ~ Clustering and their types
- ~ K-means clustering
- ~ K-means++
- ~ Batch k-means
- ~ Hierarchical clustering
- ~ DbSCAN
- ~ Evaluation of clustering
- ~ Homogeneity, completeness and v-measure
- ~ Silhouette coefficient
- ~ Davies-bouldin index
- ~ Contingency matrix
- ~ Pair confusion matrix
- ~ Extrinsic measure
- ~ Intrinsic measure
- ~ Complete end-to-end project with deployment

#### => Anomaly Detection :

- ~ Anomaly detection types
- ~ Anomaly detection applications
- ~ Isolation forest anomaly detection algorithm
- ~ Isolation forest anomaly detection algorithm
- ~ Support vector machine anomaly detection algorithm
- ~ DbSCAN algorithm for anomaly detection
- ~ Complete end-to-end project with deployment

#### => Time-Series :

- ~ What is a time series?
- ~ Old techniques
- ~ Arima
- ~ Acf and pacf
- ~ Time-dependent seasonal components.
- ~ Autoregressive (ar),
- ~ Moving average (ma) and mixed arma- modeler.
- ~ The random walk model.
- ~ Box-jenkins methodology.
- ~ Forecasts with arima and var models.
- ~ Dynamic models with time-shifted explanatory variables.
- ~ The koyck transformation.
- ~ Partial adjustment and adaptive expectation models.
- ~ Granger's causality tests.
- ~ Stationarity, unit roots and integration
- ~ Time series model performance
- ~ Various approach to solve time series problem
- ~ Complete end-to-end project with deployment
- ~ Prediction of nifty stock price and deployment

#### => NLP Basic :

- ~ Tokenization
- ~ Pos tags and chunking
- ~ Stop words
- ~ Stemming and lemmatization
- ~ Named entity recognition (ner)
- ~ Word vectorization (word embedding)
- ~ Tfidf
- ~ Complete end-to-end project with deployment

#### => Machine Learning Pipeline :

- ~ Aws segmaker
- ~ Aure ml studio
- ~ Ml flow
- ~ Kube flow

#### => Model Retraining Approach

#### => Auto ML :

- ~ H2o
- ~ Pycaret
- ~ Auto sklearn
- ~ Auto time series
- ~ Auto viml
- ~ Auto gluon
- ~ Auto viz
- ~ Tpot
- ~ Auto neuro

#### => Neural Network A Simple perception :

- ~ Detail mathematical explanation
- ~ Neural network overview and its use case.
- ~ Various neural network architect overview.
- ~ Use case of neural network in nlp and computer vision.
- ~ Activation function -all name
- ~ Multilayer network.
- ~ Loss functions. - all 10
- ~ The learning mechanism.
- ~ Optimizers. - all 10
- ~ Forward and backward propagation.
- ~ Weight initialization technique
- ~ Vanishing gradient problem
- ~ Exploding gradient problem
- ~ Visualization of nn

#### => Hardware Setup - GPU :

- ~ Gpu introduction.
- ~ Various type of gpu configuration.
- ~ Gpu provider and its pricing.
- ~ Paper space gpu setup.
- ~ Running model in gpu

#### => Tensor Flow Installation Environment Setup For Deep Learning :

- ~ Colab pro setup
- ~ Tensor flow installation 2.0 .
- ~ Tensor flow installation 1.6 with virtual environment.
- ~ Tensor flow 2.0 function.
- ~ Tensor flow 2.0 neural network creation.
- ~ Tensor flow 1.6 functions.
- ~ Tensor flow 1.6 neural network and its functions.
- ~ Keras introduction.
- ~ Keras in-depth with neural network creation.
- ~ Mini project in tensorflow.
- ~ Tensorspace
- ~ Tensorboard integration
- ~ Tensorflow playground
- ~ Netron

#### => Pytorch :

- ~ pytorch installation.
- ~ Pytorch functional overview.



~ *Pytorch neural network creation.*

=> **Mxnet :**

~ *Mxnet installation*  
~ *Mxnet in depth function overview*  
~ *Mxnet model creation and training*

=> **Keras Tuner :**

~ *Keras tuner installation and overview*  
~ *Finding best parameter from keras tuner*  
~ *Keras tuner application across various neural network*

=> **CNN Overview :**

~ *Cnn definition*  
~ *Various cnn based architecture*  
~ *Explanation end to end cnn network*  
~ *Cnn explainer*  
~ *Training cnn*  
~ *Deployment in azure cloud*  
~ *Performance tuning of cnn network*

=> **Advance Computer Vision - Part 1 :**

~ *Various cnn architecture with research paper and mathematics*  
~ *Lenet-5 variants with research paper and practical*  
~ *Alexnet variants with research paper and practical*  
~ *Googlenet variants with research paper and practical*  
~ *Transfer learning*  
~ *Vggnet variants with research paper and practical*  
~ *Resnet variants with research paper and practical*

=> **Advance Computer Vision - Part 2 :**

~ *Object detection in-depth*  
~ *Transfer learning*  
~ *Ssd with research paper and practical*

=> **Training of Custom Object Detection :**

~ *Tfod introduction*  
~ *Environment setup with tfod*  
~ *Gpu vs tpu vs cpu*  
~ *Various gpu comparison*

=> **Advance Computer Vision - Part 3 :**

~ *Yolo v1 with research paper and practical*  
~ *Yolo v2 with research paper and practical*  
~ *Yolo v3 with research paper and practical*

=> **Object Segmentation :**

~ *Semantic segmentation*  
~ *Masked rcnn*  
~ *Practical with tfod*

=> **Object Tracking :**

~ *Detail of object tracking*  
~ *Deep sort*  
~ *Object tracking live project with live camera testing*

=> **OCR :**

~ *Introduction to ocr*  
~ *Various framework and api for ocr*  
~ *Practical implementation of ocr*

=> **Advance NLP with Deep Learning :**

~ *Overview computational linguistic.*  
~ *History of nlp.*  
~ *Why nlp*  
~ *Use of nlp*

=> **Spacy :**

~ *Spacy overview.*  
~ *Spacy function*  
~ *Nltk*

=> **RNN :**

~ *Recurrent neural networks.*  
~ *Long short term memory (lstm)*  
~ *Bi lstm.*  
~ *Stacked lstm*

=> **Word Embedding :**

~ *Word embedding*  
~ *Word2vec*

=> **Attention Based Model :**

~ *Seq 2 seq.*  
~ *Encoders and decoders.*  
~ *Attention mechanism.*  
~ *Attention neural networks*  
~ *Self-attention*

=> **Transfer Learning in NLP :**

~ *Introduction to transformers.*  
~ *Bert model.*  
~ *Gpt1 model*  
~ *Gpt2 model.*

## => Deployment of Model and Performance Tuning :

- ~ *Deep learning model deployment strategies.*
- ~ *Deep learning project architecture*
- ~ *Deep learning model deployment phase.*

## => Big Data Introduction :

- ~ *What is big data?*
- ~ *Big data application*
- ~ *Big data pipeline*

## => Hadoop :

- ~ *Hadoop introduction*
- ~ *Hadoop setup and installation*

## => Spark :

- ~ *Spark*
- ~ *Spark overview.*
- ~ *Spark installation.*
- ~ *Spark rdd.*
- ~ *Spark data frame.*
- ~ *Spark architecture.*
- ~ *Spark deployment in local server*

## => Kafka :

- ~ *Kafka introduction*
- ~ *Kafka installation*
- ~ *Spark with Kafka*

## => ML Ops :

- ~ *Git*

## => SQL :

- ~ *Introduction*
- ~ *ER Daigram*
- ~ *Schema Design*
- ~ *Normalization*
- ~ *SQL SELECT Statement*
- ~ *SQL SELECT Using common functions*
- ~ *SQL JOIN Overview*
- ~ *INNER JOIN*
- ~ *LEFT JOIN*
- ~ *RIGHT JOIN*
- ~ *FULL JOIN*
- ~ *SQL Best Practice*
- ~ *INNER JOIN - Advanced*
- ~ *INNER JOIN & LEFT JOIN Combo*
- ~ *SELF JOIN*
- ~ *Joins & Aggregation - Subqueries*
- ~ *Sorting*
- ~ *Set Operations*
- ~ *SQL Views*
- ~ *Create a view*
- ~ *Create a view using DDL*
- ~ *SQL Insert - Advanced Technique*
- ~ *INSERT to create a table*
- ~ *INSERT new data to an existing table-1*
- ~ *INSERT new data to an existing table-2*
- ~ *INSERT new data to an existing table-3*
- ~ *INSERT new data to an existing table-4*
- ~ *SQL Update - Advanced Technique and TCL*
- ~ *SQL Aggregations*

## => Advance Excel :

- ~ *Microsoft Excel Fundamentals*
- ~ *Entering and Editing Text and Formulas*
- ~ *Working with Basic Excel Functions*
- ~ *Modifying an Excel Worksheet*
- ~ *Formatting Data in an Excel Worksheet*
- ~ *Inserting Images and Shapes into an Excel Worksheet*
- ~ *Creating Basic Charts in Excel*
- ~ *Printing an Excel Worksheet*
- ~ *Working with Excel Templates*
- ~ *Working with an Excel List*
- ~ *Excel List Functions*
- ~ *Excel Data Validation*
- ~ *Importing and Exporting Data*
- ~ *Excel PivotTables*
- ~ *Working with Excel's PowerPivot Tools*
- ~ *Working with Large Sets of Excel Data*
- ~ *Conditional Functions*
- ~ *Lookup Functions*

## => Tableau :

- ~ *Talking about Business Intelligence*
- ~ *Tools and Methodologies used in BI*
- ~ *Why Visualization is getting more popular*
- ~ *Why Tableau?*
- ~ *Gartner Magic Quadrant of Market Leaders*
- ~ *Future buisness impact of BI*
- ~ *Tableau Products*

- ~ Tableau Architecture
- ~ Tableau Installation in local system
- ~ Introduction to Tableau Prep
- ~ Tableau Prep Builder User Interface
- ~ Data Preparation techniques using Tableau Prep Builder tool
- ~ How to connect Tableau with different data source
- ~ Visual Segments
- ~ Visual Analytics in depth
- ~ Filters, Parameters & Sets
- ~ Tableau Calculations using functions
- ~ Tableau Joins
- ~ Dynamic Dashboards and Stories

=> Power BI :

- ~ Power BI introduction and overview
- ~ Key Benefits of Power BI
- ~ Power BI Architecture
- ~ Power BI Process
- ~ Components of Power BI
- ~ Power BI - Building Blocks
- ~ Power BI vs other BI tools
- ~ Power Installation
- ~ Overview of Power BI Desktop
- ~ Data Sources in Power BI Desktop
- ~ Connecting to a data Sources
- ~ Query Editor in Power BI
- ~ Views in Power BI
- ~ Field Pane
- ~ Visual Pane
- ~ Custom Visual Option
- ~ Filters
- ~ Introduction to using Excel data in Power BI
- ~ Import Power View and Power Pivot to Power BI
- ~ Power BI Publisher for Excel
- ~ Introducing Power BI Mobile
- ~ Power Query Introduction
- ~ Query Editor Interface
- ~ Clean and Transform your data with Query Editor
- ~ Data Type
- ~ Column Transformations vs Adding Columns
- ~ Text Transformations
- ~ Data Modelling
- ~ Calculated Columns

=> GAN

### Project details :-

=> Python Project :

- ~ Weeding script
- ~ Image resizing
- ~ Web scrapping
- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis.
- ~ Integration with web portal.
- ~ Integration with rest api, web portal and mongo db. on azure
- ~ Deployment on web portal on azure.

=> Major Projects :

- ~ Healthcare analytics prediction of medicines based on Fitbit band.
- ~ Revenue forecasting for startups.
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ anomaly detection in inventory packaged material.
- ~ Fault detection in wafers based on sensor data.
- ~ Demand forecasting for fmcg product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with zomato dataset.
- ~ Fault detection in wafers based on sensor data.
- ~ Cement strength reg.
- ~ Credit card fraud.
- ~ Forest cover classification.
- ~ Fraud detection.
- ~ Income prediction.
- ~ Mushroom classifier.
- ~ phishing classifier
- ~ Thyroid detection.
- ~ Visibility climate

=> Computer Vision Project :

- ~ Traffic surveillance system.
- ~ Object identification.
- ~ Object classification.
- ~ Tensorflow object detection.

=> Mini NLP Project :

- ~ Machine translation.
- ~ Abstractive text summarization.
- ~ Keyword spotting.
- ~ Language modelling.

=> NLP Transfer Learning Project :

- ~ *Deployment and integration with UI machine translation.*
- ~ *Question answering (like chat bot)*
- ~ *Sentiment analysis imdb.*
- ~ *Text search (with synonyms).*

=> NLP End to End Project with Architecture and Deployment :

- ~ *Movie review using bert*
- ~ *NER using Bert*
- ~ *Pos bert*
- ~ *Text generation gpt 2*

=> NLP Project End to End with Deployment in Various Cloud and UI Integration :

- ~ *Topic modeling.*
- ~ *Text to speech*

=> Tableau Project :

- ~ *Human Resource - Tableau*

=> Power BI Project :

- ~ *Retail Insights- Power BI*

# Full Stack Data Science Nov'21 Tech Neuron

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science-Nov'21-Tech-Neuron>

## Course Description :-

This is a full stack data science self-paced course with recordings of live mentor-led classes and a full-time one-year internship provided by iNeuron intelligence private limited, where you will learn all the stack required to work in the data science, data analytics, and big data industries, including machine learning operations and cloud infrastructure, as well as real-time industry project and product development with the iNeuron product development team, and you will contribute on various levels.

## Course Features :-

- => Full stack Data Science Recorded Lectures
- => One year of internship Anytime
- => Online Instructor-led learning: Live teaching by instructors
- => 56 + hands-on industry real-time projects.
- => 500 hours live interactive classes.
- => Lifetime Dashboard access
- => Assignment in all the module

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => Architecture
- => Databases

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sunny Savita :

~ I'm an AI enthusiast, graduate in Computer science and engineering. Currently working with iNeuron.ai as a Data Scientist and having 2+ years of experience. I have skills in big data, machine learning, computer vision, Natural language processing. My expertise also includes project design development and implementation with AI/ops tools.

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Python Fundamentals :

- ~ Python Basic
- ~ String, List, Indexing
- ~ Tuple, Set & Dict
- ~ If, Else & For Loop
- ~ For Loops & While loops
- ~ Python Program Discussion in loops

~ *Function Part - 1*

~ *Function Part - 2*

=> **Advanced Python :**

~ *Iterator Generator & File System*

~ *Exception handling Class 1 part 1*

~ *Exception handling Class 1 part 2*

~ *Exception handling Class 2*

~ *Module & Packages*

~ *OOPS Part 1*

~ *OOPS Part 2*

~ *OOPs Concepts - Polymorphism*

=> **Working with Databases & Python :**

~ *SQL Part 1*

~ *SQL Part 2*

~ *OOPS Discussion*

~ *Introduction to MongoDB*

~ *Working with Python & MongoDB Part1*

~ *Working with Python & MongoDB Part2*

~ *SQL lite, map, reduce, filter, zip*

=> **Working with Pandas & Numpy :**

~ *Introduction to Pandas*

~ *Working with Pandas*

~ *Pandas Data Analysis Part 1*

~ *Pandas Data Analysis Part 2*

~ *Pandas and Numpy*

~ *Numpy methods*

=> **GUI Programming :**

~ *GUI Programming with Tkinter*

=> **Working with Graphs & Charts :**

~ *Introduction to Graphs & Charts*

~ *Working with Graphs in Python*

=> **API :**

~ *API Testing*

=> **Python Projects :**

~ *Flask End-to-End Project*

~ *Review Scraper*

~ *Image Scraper and deployment on Heroku, AWS and Azure*

=> **Statistics :**

~ *Introduction to Stats - Day 1*

~ *Stats - Day 2*

~ *Extra doubt session*

~ *Stats - Day 3*

~ *Stats - Day 4*

~ *Stats - Day 5*

=> **EDA & Feature Engineering :**

~ *Introduction to EDA*

~ *Doubt Clearing session*

~ *EDA and Feature Engineering*

=> **Machine Learning :**

~ *Linear Regression*

~ *Ridge Lasso Regression, Elastic & Logistic Regression*

~ *Naive Bayes Algorithm and practical implementation of Ridge Lasso and Logistic Regression*

~ *Logistic Practical, SSVM, SVR*

~ *Decision Tree Classification*

~ *Random Forest & SVM*

~ *Adaboost*

~ *Gradient Boosting*

~ *Clustering*

~ *Introduction to Machine learning*

~ *Linear Regression*

~ *Linear Regression live coding demonstration part-1*

~ *Linear Regression live coding demonstration part-2*

~ *Project Admission Prediction, Lasso, Ridge & Elastic Net*

~ *Project deployment in Heroku, Azure & AWS*

~ *Logistic Regression*

~ *Logistic Regression implementation*

~ *Decision Tree*

~ *Decision Tree Part 2 , Ensemble Tech, Random Forest & Boosting*

~ *KNN and SVM*

~ *Decision Tree Practical Implementation*

~ *Decision Tree Live Coding & Grid Search*

~ *Grid Search, Bagging Classifier & Random Forest*

~ *KNN, SVC, SVR & Stacking*

~ *Clustering*

~ *Clustering and PCA*

~ *PCA practical, DBSCAN and Naive Bayes*

~ *XG Boost, NLTK & TF-IDF*

=> **Machine Learning End to End Project :**

~ *Machine learning project*

~ *Machine learning project*

~ *ML End to End project Pipeline Explanation*

- ~ *ML Project Explanation along with GitHub and Docker*
- ~ *Machine Learning Pipelines Live Coding Part-1*
- ~ *Machine Learning Pipelines Live Coding Part-2*
- ~ *2nd July Live Class*
- ~ *Machine Learning Pipelines Live Coding Part-2*
- ~ *Revision Class*
- ~ *Model training, evaluation, and push*
- ~ *Model training, evaluation, and push*
- ~ *Revision*

=> PCA in ML :

- ~ *PCA*
- ~ *PCA Implementation*

=> NLP for Machine Learning :

- ~ *NLP in ML*
- ~ *Spam Classification*

=> Time Series Analysis :

- ~ *Introduction to Time Series*
- ~ *Time Series Implementation*

=> Stats :

- ~ *Introduction*
- ~ *Different types of Statistics*
- ~ *Population vs Sample*
- ~ *Mean, Median and Mode*
- ~ *Variance, Standard Deviation*
- ~ *Sample Variance why n-1*
- ~ *Standard Deviation*
- ~ *Variables*
- ~ *Random Variables*
- ~ *Percentiles & quartiles*
- ~ *5 number summary*
- ~ *Histograms*
- ~ *Gaussian - Normal distribution*
- ~ *Standard Normal distribution*
- ~ *Application Of Zscore*
- ~ *Basics Of Probability*
- ~ *Addition Rule In Probability*
- ~ *Multiplication rule in probability*
- ~ *Permutation*
- ~ *Combination*
- ~ *Log Normal Distribution*
- ~ *Central Limit theorem*
- ~ *Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median And Mode*
- ~ *Covariance*
- ~ *Pearson And Spearman Rank Correlation*
- ~ *What is P Value*
- ~ *What is Confidence Intervals*
- ~ *How To Perform Hypothesis Testing - Confidence IntervalZ Test Statistics Derive Conclusion*
- ~ *Hypothesis testing part 2*
- ~ *Hypothesis testing part 3*
- ~ *Finalizing statistics*

=> ML Projects :

- ~ *Detailed Project Report explanation*
- ~ *Project :- Wafer Fault Detection Part 1*
- ~ *Project :- Wafer Fault Detection Part 2*
- ~ *Deployment in Heroku using docker and circleci*

=> ML Project 1 :- Fault detection in wafers based on sensor data :

- ~ *Introduction*
- ~ *The problem statement and Data Description*
- ~ *The Application Flow*
- ~ *Ingestion and Validation Part1*
- ~ *Validation Part2*
- ~ *DB Operations*
- ~ *Data Preprocessing*
- ~ *Clustering*
- ~ *Model Selection and Tuning*
- ~ *Prediction*
- ~ *Deployment*

=> ML Project 2 :- Cement Strength Prediction :

- ~ *Introduction*
- ~ *The Problem Statement and Data Description*
- ~ *The Application Flow*
- ~ *Code Intro and Logging*
- ~ *Validation and Transformation*
- ~ *DB Operations*
- ~ *Data Preprocessing*
- ~ *Clustering*
- ~ *Model Selection and Tuning*
- ~ *Prediction*
- ~ *Deployment*

=> ML Project 3 :- Credit Card Defaulters :

- ~ *Introduction*
- ~ *The Problem Statement and Data Description*
- ~ *The Application Flow*

- ~ Code intro and Logging
- ~ Validation and Transformation
- ~ DB Operations
- ~ Data Preprocessing
- ~ Deployment

#### => Time Series :

- ~ Arima, Sarima, Auto Arima
- ~ Time series using RNN LSTM, Prediction of NIFTY stock price
- ~ Time series using RNN LSTM, Prediction of NIFTY stock price

#### => DL ANN - Introduction :

- ~ Introduction to Deep Learning
- ~ Importance of Deep learning
- ~ Why you should study Deep Learning? (Motivation)
- ~ ANN vs BNN
- ~ The first Artificial Neuron

#### => DL ANN - Perceptron :

- ~ Overview of Perceptron
- ~ More about Perceptron
- ~ Perceptron implementation using python - 1
- ~ Perceptron implementation using python - 2
- ~ Perceptron implementation using python - 3
- ~ Perceptron implementation using python - 4
- ~ Perceptron implementation using python - 5
- ~ Perceptron implementation using python - 6
- ~ Perceptron implementation using python - 7
- ~ Python scripting & modular coding for Perceptron
- ~ Python logging basics and docstrings
- ~ Python packaging, Github actions, and PyPI

#### => DL ANN - 1 :

- ~ Multilayer Perceptron
- ~ Forward propagation
- ~ Why we need Activation function?
- ~ ANN implementation using tf.keras - 1
- ~ ANN implementation using tf.keras - 2
- ~ ANN implementation using tf.keras - 3
- ~ ANN implementation using tf.keras - 4
- ~ ANN with Callbacks | Tensorboard | Early Stopping | Model Checkpointing

#### => DL ANN - 2 :

- ~ Vector
- ~ Differentiation
- ~ Partial differentiation
- ~ Maxima and minima concept
- ~ Gradient descent basics
- ~ In-depth understanding of Gradient descent with mathematical proof

#### => DL ANN - 3 :

- ~ Chain rule
- ~ Backpropagation

#### => DL ANN - 4 :

- ~ General problems in training Neural Networks
- ~ Vanishing and Exploding gradients
- ~ Activation Function Basics
- ~ Weight initialization
- ~ Activation Functions - 1
- ~ Activation functions - 2
- ~ Activation functions - 3
- ~ Transfer learning
- ~ Batch normalization -1
- ~ Batch normalization -2
- ~ Batch normalization -3

#### => DL ANN - 5 :

- ~ Introduction to fast optimizers
- ~ Momentum optimization
- ~ NAG
- ~ Elongated bowl problem | AdaGrad
- ~ RMSProp
- ~ Adam
- ~ Loss functions
- ~ Regularization
- ~ Dropout

#### => Computer Vision - Introduction :

- ~ Introduction to Course
- ~ Course Overview
- ~ Installing Anaconda, Pycharm & Postman
- ~ Working with Conda Envs
- ~ Pycharm Introduction
- ~ Pycharm with Conda
- ~ Pycharm with venv
- ~ Pycharm with Pipenv

#### => Computer Vision - CNN Foundations :

- ~ Why CNN? Building an Intuition for CNN
- ~ CNN, Kernels, Channels, Feature Maps, Stride, Padding
- ~ Receptive Fields, Image Output Dimensionality Calculations, MNIST Dataset Explorations with CNN



- ~ *MNIST CNN Intuition, Tensorspace.js, CNN Explained, CIFAR 10 Dataset Explorations with CNN*
- ~ *Dropout & Custom Image Classification Dog Cat Dataset*
- ~ *Deployment in Heroku, AWS, Azure*
- ~ *Deployment in GCP, AWS EBS*

#### => Computer Vision - CNN Architectures :

- ~ *LeNet-5*
- ~ *LeNet-5 Practical*
- ~ *AlexNet*
- ~ *AlexNet Practical*
- ~ *VGGNet*
- ~ *VGG16 Practical*
- ~ *Inception*
- ~ *Inception Practical*
- ~ *ResNet*
- ~ *Resnet Practical*

#### => Computer Vision - Image Classification Hyper Parameter Tuning :

- ~ *Keras Tuner*
- ~ *Building a simple model*
- ~ *Tuning with Keras Tuner*

#### => Computer Vision - Data Augmentation :

- ~ *What is Data Augmentation?*
- ~ *Benefits of Data Augmentation*
- ~ *Exploring Papers like RICAP, Random Erasing, Cutout*
- ~ *Exploring Augmentor*
- ~ *Exploring Roboflow*

#### => Computer Vision - Object Detection Basics :

- ~ *What is Object Detection?*
- ~ *Competitions for Object Detection*
- ~ *Bounding Boxes*
- ~ *Bounding Box Regression*
- ~ *Intersection over Union (IoU)*
- ~ *Precision & Recall*
- ~ *What is Average Precision?*

#### => Computer Vision - Object Detection Architectures :

- ~ *Object Detection Family*
- ~ *RCNN*
- ~ *RCNN Network Architecture*
- ~ *Cons of RCNN*
- ~ *FAST RCNN*
- ~ *FAST RCNN Network Architecture*
- ~ *Cons of FAST RCNN*
- ~ *FASTER RCNN*
- ~ *FASTER RCNN Network Architecture*
- ~ *YOLO*
- ~ *YOLO Architecture*
- ~ *YOLO Limitations*
- ~ *SSD*
- ~ *SSD Network*

#### => Computer Vision - Practicals Object Detection using Tensorflow 1.x :

- ~ *Introduction to TFOD1.x*
- ~ *Using Google Colab with Google Drive*
- ~ *Installation of Libraries in Colab*
- ~ *TFOD1.x Setup in Colab*
- ~ *Visiting the Model Zoo*
- ~ *Inferencing in Colab*
- ~ *Inferencing in Local*
- ~ *Important Configurations Files*
- ~ *Webcam Testing*

#### => Computer Vision - Practicals Training a Custom Cards Detector using Tensorflow1.x :

- ~ *Custom Model Training in TFOD1.x*
- ~ *Our Custom Dataset*
- ~ *Doing Annotations or labeling data*
- ~ *Selection of Pretrained Model from Model Zoo*
- ~ *Files Setup for Training*
- ~ *Let's start Training in Colab*
- ~ *Export Frozen Inference Graph*
- ~ *Inferencing with our trained model in Colab*
- ~ *Training in Local*
- ~ *Inferencing with our trained model in Local*

#### => Computer Vision - Practicals Creating an Cards Detector Web App with TFOD1 :

- ~ *Code Understanding*
- ~ *WebApp Workflow*
- ~ *Code Understanding*
- ~ *Prediction with Postman*
- ~ *Debugging our Application*

#### => Computer Vision - Practicals Object Detection using Tensorflow 2.x :

- ~ *Introduction to TFOD2.x*
- ~ *Using the Default Colab Notebook*
- ~ *Google Colab & Drive Setup*
- ~ *Visiting TFOD2.x Model Garden*
- ~ *Inference using Pretrained Model*
- ~ *Inferencing in Local with a pretrained model*

=> Computer Vision - Practicals Training a Custom Chess Piece Detector using Tensorflow2 :

- ~ Custom Model training in TFOD2.x
- ~ Our Custom Dataset TF2
- ~ File Setup for Training
- ~ Let's start Training
- ~ Let's start Training
- ~ Stop Training or resume Training
- ~ Evaluating the trained model
- ~ Convert CKPT to Saved Model
- ~ Inferencing using the Custom Trained Model in Colab
- ~ Inferencing using the Custom Trained Model in Local PC

=> Computer Vision - Practicals Creating an Chess Piece Detector Web App with TFOD2 :

- ~ Creating a Pycharm project & Environment Setup TF2
- ~ Application Workflow
- ~ Code understanding
- ~ Testing our App with Postman
- ~ Debugging our Application

=> Computer Vision - Practicals Object Detection using Detectron2 :

- ~ Introduction to Detectron2
- ~ Detectron2 Colab Setup
- ~ Visiting Detectron2 Model Zoo
- ~ Detectron2 Pretrained Model Inferencing

=> Computer Vision - Practicals Training a Custom Detector using Detectron2 :

- ~ Detectron2 Custom Training
- ~ Exploring the Dataset
- ~ Registering Dataset for Training
- ~ Let's start Training
- ~ Inferencing using the Custom Trained Model in Colab
- ~ Evaluating the Model

=> Computer Vision - Practicals Creating an Custom Detector Web App with Detectron2 :

- ~ Creating a Pycharm project & Environment Setup Detectron2
- ~ Application Workflow
- ~ Code understanding
- ~ Testing our App with Postman
- ~ Debugging our Application

=> Computer Vision - Practicals Object Detection using YoloV5 :

- ~ Introduction to YoloV5
- ~ YoloV5 Colab Setup
- ~ Inferencing using Pre Trained Model

=> Computer Vision - Practicals Training a Custom Warehouse Apparel Detector using YoloV5 :

- ~ Custom Training with YoloV5
- ~ Exploring the Dataset
- ~ Doing Annotations or labeling data
- ~ Setting up Google Colab & Drive
- ~ Let's start Training
- ~ Inferencing using the Custom Trained Model in Colab

=> Computer Vision - Practicals Creating an Warehouse Apparel Detector Web App with YOLOV5 :

- ~ Creating a Pycharm project & Environment Setup Yolo
- ~ Application Workflow
- ~ Code understanding
- ~ Testing our App with Postman
- ~ Debugging our Application

=> Computer Vision - Image Segmentation :

- ~ Segmentation Introduction
- ~ From Bounding Box to Polygon Masks
- ~ What is Image Segmentation?
- ~ Types of Segmentation
- ~ MASKRCNN
- ~ MASK RCNN Architecture

=> Computer Vision - MASK RCNN Practicals with TFOD :

- ~ Segmentation with TFOD1.x
- ~ Local Setup MASKRCNN
- ~ Exploring the Dataset
- ~ Data Annotation
- ~ Model Selection
- ~ Files Setup for Training
- ~ Model Training
- ~ Export Frozen Inference Graph
- ~ Model Prediction

=> Computer Vision - MASKRCNN practical with Detectron2 :

- ~ Introduction to Detectron2
- ~ Detectron2 Colab Notebook
- ~ Exploring the Model Zoo
- ~ Detectron2 Colab Setup
- ~ Custom Training with Detectron2
- ~ Exploring our Dataset
- ~ Data Annotation
- ~ Data Preparation
- ~ Setup for Training
- ~ Let's start Training
- ~ Inferencing using the Custom Trained Model in Colab

~ *Evaluating the Model*

=> **Computer Vision - Face Recognition Project :**

~ *Introduction to Project*  
~ *Requirement Gathering*  
~ *Techstack Selection*  
~ *Project Installation*  
~ *Project Demo*  
~ *Project Workflow*  
~ *Core Components of the Application*  
~ *Data Collection Module*  
~ *Generate Face Embeddings*  
~ *Training Face Recognition Module*  
~ *Prediction Pipeline*  
~ *Entry point of the Application*  
~ *Application Workflow*  
~ *Debugging our Application*

=> **Computer Vision - Object Tracking Project :**

~ *Object Tracking project*  
~ *Project Installation Tracking*  
~ *Project Demo*  
~ *Code Understanding*

=> **Computer Vision - GANS :**

~ *Introduction to GANS*  
~ *GAN Architecture*  
~ *GAN PRACTICALS Implementation*

=> **Computer Vision Project - Traffic Vehicle Detection :**

~ *Introduction to Vehicle Detection project*  
~ *Requirement Gathering*  
~ *Framework Selection*  
~ *Detailed Project Workflow*  
~ *Data Collection Scrap*  
~ *Data Preparation*  
~ *Data augmentation augmenter*  
~ *Data Annotations*  
~ *Model Training*  
~ *Creating a Pycharm project & Environment Setup TVD*  
~ *WebApp Workflow*  
~ *Code Understanding*  
~ *Prediction with Postman*  
~ *Debugging our Application*

=> **Computer Vision Project - Helmet Detection :**

~ *Introduction to Helmet Detection project*  
~ *Requirement Gathering*  
~ *Techstack Selection*  
~ *Detailed Project Workflow*  
~ *Data Collection*  
~ *Data Preparation*  
~ *Data Augmentation*  
~ *Data Annotations*  
~ *Model Training*  
~ *Creating a Pycharm project & Environment Setup HD*  
~ *WebApp Workflow*  
~ *Code Understanding*  
~ *Prediction with Postman*  
~ *Debugging our Application*

=> **Computer Vision Project - Fashion Apparel Detection :**

~ *Introduction to Fashion Apparel Detection project*  
~ *Requirement Gathering*  
~ *Techstack Selection*  
~ *Detailed Project Workflow*  
~ *Data Collection*  
~ *Data Preparation*  
~ *Data Augmentation*  
~ *Data Annotations*  
~ *Model Training*  
~ *Creating a Pycharm project & Environment Setup FAD*  
~ *Project Demo*  
~ *WebApp Workflow*  
~ *Code Understanding*  
~ *Prediction with Postman*  
~ *Debugging our Application*

=> **Computer Vision Project - Image TO Text OCR :**

~ *Introduction to Project*  
~ *Project Installation OCR*  
~ *Project Demo*  
~ *Application Workflow*  
~ *Code Understanding*  
~ *Debugging our App*  
~ *Different OCR's available*

=> **Computer Vision Project - Shredder System :**

~ *Introduction to Shredder Systems*  
~ *Requirement Gathering*  
~ *Techstack Selection*

- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation
- ~ Model Selection from Zoo
- ~ Model Training
- ~ Creating a Pycharm project & Environment Setup SS
- ~ Application Workflow
- ~ Project Demo
- ~ Code Understanding
- ~ Debugging our Application
- ~ Project Workflow
- ~ Project Workflow

=> Computer Vision Project - Automatic Number plate Recognition with TFOD1.x :

- ~ Introduction to ANPR Project
- ~ Requirement Gathering
- ~ Tech Stack Selection
- ~ Data Collection
- ~ Data Augmentation
- ~ Data Preparation
- ~ Data Annotation
- ~ Model Selection From Zoo
- ~ Model Training
- ~ Creating a Pycharm project & Environment Setup ANPR
- ~ Application Workflow
- ~ Create Google OCR API Key
- ~ Project Demo
- ~ Code Understanding
- ~ Debugging our Application

=> NLP Overview :

- ~ NLP Overview
- ~ NLP very basic

=> NLP Word Embeddings :

- ~ TFIDF
- ~ Word Embeddings Part-1
- ~ Word Embeddings Part-2

=> NLP RNN :

- ~ RNN basic
- ~ RNN Implementation

=> NLP LSTM & GRU :

- ~ LSTM Introduction
- ~ GRU

=> NLP Attention Based Model :

- ~ Encoder Decoder and Attention Mechanism
- ~ Attention All You Need Paper Understanding

=> NLP Transfer Learning in NLP :

- ~ GPT and BERT Model
- ~ SOTA Model with Paper Discussions
- ~ Albert & DistillBert Project Discussion

=> NLP Project :- Megatron :

- ~ Megatron Project

=> NLP Project:- Brand Measures :

- ~ Brand Measures Project

=> NLP Project:- Text to Speech :

- ~ Introduction
- ~ Project Setup Text to Speech
- ~ Project Demo
- ~ Code Explanation
- ~ Project Workflow
- ~ Prediction with Postman
- ~ Debugging Application

=> NLP Project:- Speech To Text :

- ~ Introduction
- ~ Project Setup Speech To Text
- ~ Project Demo
- ~ Code Explanation
- ~ Project Workflow
- ~ Prediction with Postman
- ~ Debugging Application

=> NLP Project:- Spell Corrector :

- ~ Introduction
- ~ Project Setup Spell Corrector
- ~ Project Demo
- ~ Code Explanation
- ~ Project Workflow
- ~ Prediction with Postman
- ~ Debugging Application

=> NLP Project:- Named Entity Recognition :

- ~ NER using BERT

=> NLP Project:- Machine Translation & Keyword Spotting :

- ~ Machine Translation
- ~ Keyword Spotting

=> NLP Project:- Keyword Extractor & Summarization :

- ~ Keyword Extraction
- ~ Extractive Text Summarization

=> NLP project:- Paraphrasing :

- ~ Rephrase Project

=> BigData - Introduction to Big Data and Data Engineering :

- ~ Big Data Engineering

=> BigData - Introduction to Distributed Systems - Hadoop and MapReduce :

- ~ Big Data Engineering Introduction

=> BigData - Map Reduce & YARN :

- ~ Big Data Hadoop Map Reduce YARN
- ~ Hadoop Map Reduce Hands On

=> BigData - Hive :

- ~ Apache hive

=> BigData - Hive Hands On :

- ~ Apache hive Hands On

=> BigData - NoSQL and Hbase :

- ~ Big Data HBase
- ~ Hbase hands On

=> BigData - Sqoop :

- ~ Big Data Sqoop
- ~ Big Data Sqoop Hands On

=> BigData - Spark :

- ~ Spark - Introduction
- ~ Big Data Engineering using PySpark- RDDs
- ~ Spark hands on - RDD
- ~ Big Data Engineering using PySpark- Core, Internals, Architecture
- ~ Apache Spark Actions\_ Transformations
- ~ Apache Spark Caching
- ~ Big Data Engineering using PySpark- Shared Vars , Coalesce Repartition
- ~ Big Data Engineering using PySpark- Dataframes
- ~ Spark hands on - Dataframe
- ~ Spark hands on - Databricks
- ~ Big Data Engineering using PySpark- Catalyst& Tungsten

=> BigData - Spark ML :

- ~ Big Data Engineering using PySpark- MLlib
- ~ Spark hands On - Spark ML Lib

=> BigData - Spark Streaming :

- ~ Big Data Engineering using PySpark- Streaming Part 1
- ~ Big Data Engineering using PySpark- Streaming Part 2
- ~ Spark hands On - Spark Streaming

=> BigData - Kafka :

- ~ Big Data Kafka
- ~ Big Data Kafka Hands on

=> BigData - Apache Airflow - Workflow Management Platform :

- ~ Big Data - Airflow
- ~ Big Data Airflow Hands On

=> Big Data Projects :

- ~ IoT Sensor data pipeline using Kafka-Spark Streaming
- ~ Product Recommendation Engine using Kafka-Spark Streaming
- ~ Short Video App Analytics

=> Basic Charts in Power BI :

- ~ 2.0 Basic Charts in Power BI Desktop
- ~ 2.1 Column Chart in Power BI
- ~ 2.2 Stacked Column Chart in Power BI
- ~ 2.3 Pie Chart in Power BI
- ~ 2.4 Donut Chart in Power BI
- ~ 2.5 Funnel Chart in Power BI
- ~ 2.6 Ribbon Chart
- ~ 2.7 Include and Exclude
- ~ 2.8 Export data from Visual

=> Working with Maps :

- ~ 3.1 Creating a Map in Power BI
- ~ 3.2 Filled Map
- ~ 3.3 Map with Pie Chart
- ~ 3.4 Formatting in Map
- ~ 3.5 Change Background in Map
- ~ 3.6 Map of India in Power BI
- ~ 3.7 Map of Australia in Power BI

=> Tables and Matrix in Power BI :

- ~ 4.0 Table and Matrix in Power BI
- ~ 4.1 Creating a Table in Power BI
- ~ 4.2 Formatting a Table

- ~ 4.3 Conditional Formatting in Table
- ~ 4.4 Aggregation in Table
- ~ 4.5 Matrix in Power BI
- ~ 4.6 Conditional Formatting in Matrix
- ~ 4.7 Hierarchy in Matrix
- ~ 4.8 Sub-Total and Total in Matrix
- ~ 4.9 Number Formatting in Table

#### => Other Charts in Power BI :

- ~ 5.0 Other Charts in Power BI
- ~ 5.1 Line Chart in Power BI
- ~ 5.2 Drill Down in Line Chart
- ~ 5.3 Area Chart in Power BI
- ~ 5.4 Line vs Column Chart in Power BI
- ~ 5.5 Scatter Plot in Power BI
- ~ 5.6 Waterfall Chart in Power BI
- ~ 6.7 TreeMap in Power BI
- ~ 5.8 Gauge Chart in Power BI

#### => Cards and Filters :

- ~ 6.0 Cards and Filters in Power BI
- ~ 6.1 Number Card
- ~ 6.2 Text Card
- ~ 6.2.1 Formatting of Text Card
- ~ 6.3 Date Card
- ~ 6.3.1 Date Card (Relative Filtering)
- ~ 6.4 Multi-Row Card
- ~ 6.5 Filter on Visual
- ~ 6.6 Filter on This Page
- ~ 6.7 Filter on All Pages
- ~ 6.8 Drillthrough in Power BI

#### => Slicers in Power BI :

- ~ 7.0 Slicers in Power BI
- ~ 7.1 Text Slicers in Power BI
- ~ 7.2 Formatting a Text Slicer
- ~ 7.3 Date Slicers in Power BI
- ~ 7.4 Formatting a Date Slicer
- ~ 7.5 Number Slicers in Power BI

#### => Introduction to tableau :

- ~ Tableau Introduction
- ~ Download and Install Tableau
- ~ Tableau Vs Excel

#### => Charts - 1 :

- ~ Column Chart
- ~ Horizontal Bar Chart
- ~ Stacked Column Chart
- ~ Stacked Bar Chart
- ~ Keep Only, Exclude
- ~ Keep Only, Exclude2\_Normal
- ~ Publish to Tableau Public

#### => Charts - 2 :

- ~ Pie Chart
- ~ Multiple Pie Chart
- ~ TreeMap\_Editing
- ~ Packed Bubble Chart
- ~ Word Cloud OR Word Map
- ~ Formatting payal

#### => Charts - 3 :

- ~ Data Types in Tableau
- ~ Filled Map
- ~ Symbol Maps
- ~ India Map
- ~ Histogram

#### => SQL :

- ~ Database Architecture
- ~ Introduction to SQL
- ~ Constraints
- ~ Data Definition Language (DDL)
- ~ Data Query Language (DQL)
- ~ Data Manipulation Language (DML)
- ~ Joins
- ~ Import Export
- ~ Aggregate Functions
- ~ Order by, Having & Limit Clause
- ~ String Functions
- ~ Datetime functions
- ~ Understanding Regular Expressions
- ~ Nested Queries
- ~ Views
- ~ Stored Procedures
- ~ WindowsFn
- ~ Python-SQL Connectivity

#### => Excel :

- ~ Introduction to Excel

- ~ *Pre-defined functions*
- ~ *Datetime Functions*
- ~ *String functions*
- ~ *Mathematical functions*
- ~ *Lookup*
- ~ *Logical & Error Functions*
- ~ *Statistical Functions*
- ~ *Images in Excel*
- ~ *Excel Formatting*
- ~ *Custom Formatting*
- ~ *Conditional Formatting*
- ~ *Charts in Excel*
- ~ *Data Analysis using Excel*
- ~ *Pivot Tables*
- ~ *Dashboarding in Excel*
- ~ *Others*
- ~ *What-If Tools - Scenario Manager, Goal Seek*

# PostgreSQL

---

Topic Name : DATABASE

Sub-topic Name : POSTGRES SQL

Course link : <https://ineuron.ai/course/PostgreSQL>

## Course Description :-

One of the most advanced open-source relational database systems is PostgreSQL. It has a number of features that assist developers in creating applications, administrators in maintaining data integrity, and data analysts and data scientists in managing their data regardless of the dataset size.

## Course Features :-

- => Completion Certificate
- => Quiz in every module
- => Real-time Project
- => Assignment in all modules

## What you will learn :-

- => Exploring Database
- => Understanding how to use relational database.
- => Importance of PostgreSQL.
- => Different operations using PostgreSQL

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Manjunatha A :

*~ Data Scientist with good experience in machine learning, deep learning, and Python programming. I was awarded the Gold medal in my Master's (MCA). I was also privileged to be honoured with the Ace of innovation award. I was also one of the Finalists of SIH -2020 the world's largest hackathon. In my spare time, I enjoy sharing my technological abilities and knowledge through classes. I supervised over 500 students and assisted them in establishing careers in their industries. I also travel frequently.*

## Curriculum details :-

=> Introduction to PostgreSQL :

- ~ Course Introduction Preview
- ~ Who this course for?
- ~ Course Curriculum overview Preview
- ~ Course Outcome
- ~ History of PostgreSQL
- ~ Key Features of PostgreSQL Preview

=> Environment setup and Installation :

- ~ PostgreSQL Download and Installation

=> PostgreSQL server :

- ~ PostgreSQL Server

=> Fundamentals of PostgreSQL :

- ~ Introduction to Database
- ~ Create database
- ~ Connecting Database
- ~ DROP Database
- ~ Introduction to Table
- ~ Create Table
- ~ INSERT data
- ~ SELECT data
- ~ DELETE Record
- ~ Update Table
- ~ Truncate table
- ~ Drop table

=> Schema and Table space :

- ~ Schema
- ~ Table Space

=> Clause :

- ~ WHERE Clause Preview
- ~ GROUP BY clause
- ~ ORDER clause
- ~ HAVING clause



~ *DISTINCT clause*

## => ALTER TABLE :

~ *Introduction to ALTER table*

~ *ADD a column*

~ *DROP a column*

~ *Rename column*

~ *Rename Table*

~ *ADD Constraint*

## => Constraints :

~ *Introduction to Constraint and all the constraint info*

~ *Foreign key Constraint*

## => Operators in postgresQL :

~ *Introduction to Operators*

~ *PostgreSQL Comparison operators*

~ *PostgreSQL BETWEEN Operator*

~ *PostgreSQL NOT BETWEEN Operator*

~ *PostgreSQL NOT Operator*

~ *Postgres LIKE Operator*

~ *PostgreSQL OR Operator*

~ *PostgreSQL AND Operator*

~ *PostgreSQL LIMIT Operator*

~ *PostgreSQL IN Operator*

~ *PostgreSQL UNION Operator*

~ *PostgreSQL INTERSECT Operator*

## => Join in PostgreSQL :

~ *Introduction to Postgres JOINS*

~ *PostgreSQL Inner Join Preview*

~ *PostgreSQL Full Outer Join*

~ *PostgreSQL Left Join*

~ *PostgreSQL Right Join*

~ *PostgreSQL Cross Join*

~ *PostgreSQL Natural Join*

## => View in PostgreSQL :

~ *Views*

## => Database security and roles :

~ *Introduction to database roles management*

~ *Database Role*

~ *Role Attributes*

~ *Default Roles*

## => Database backup and Restore :

~ *Introduction to database backup*

~ *Backing up a database Preview*

~ *Backing up a database using pgadmin*

~ *Restoring a database using cmd*

~ *Restoring a database using pgadmin*

# Arduino Live Class

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/Arduino-Live-Class>

## Course Description :-

This course was created to help students comprehend what they're doing. You can start from the beginning, gain the essential foundation, and learn the entire process of creating Arduino projects through practice and hands-on training.

## Course Features :-

- => Online live classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Arduino UNO
- => Arduino Mega (R3)
- => Arduino Mega
- => Arduino Leonardo
- => Arduino Due
- => Arduino Zero

## Requirements :-

- => A multimeter
- => A breadboard
- => Wires
- => An Arduino UNO board
- => Your dedication

## Instructors :-

=> Kishan Menaria :

~ Having 2+ years of DataScience and AI in Hardware Edge Devices, proficient in data modelling, data preprocessing as well as scripting language Python Programming Language have experience with Robot Operating System (ROS) Have Experience in AI in Edge Devices. Machine Learning and Deep Learning (Computer Vision) are two of my areas of expertise. I love the R&D work and making Businesses models and strategies or Developing technical solutions for business problems.

## Curriculum details :-

=> Introduction to the course :

- ~ Course Overview and Roadmap Ahead
- ~ Hardware parts and tools requirement

=> Overview of Arduino Family :

- ~ Arduino UNO
- ~ Arduino Mega (R3)
- ~ Arduino Mega
- ~ Arduino Leonardo
- ~ Arduino Due
- ~ Arduino Zero
- ~ Arduino 101
- ~ Arduino Pro Mini

=> Installation and Setup IDEs :

- ~ Arduino IDE installation
- ~ Arduino IDE setup
- ~ Understanding the Preferences pane and Menu items
- ~ Connect your Arduino board and Find it on the Arduino IDE
- ~ Using simulator: Tinkercad

=> Detail Explanation of Arduino Boards :

- ~ Atmega328P
- ~ USB
- ~ Shields

- ~ Power
- ~ Clock
- ~ Digital output pins
- ~ Digital input pins
- ~ Analogue output pins
- ~ Analogue input pins

#### => Prototyping in Arduino :

- ~ Understand and configure breadboard Using the breadboard.
- ~ Powering your Arduino with power supplies
- ~ Using the multimeter to measure voltage
- ~ Using the multimeter to measure current
- ~ The multimeter - Resistance and continuity
- ~ Introduction to soldering - the soldering iron
- ~ Soldering - preparation and using holders
- ~ Soldering - using wire cutters and fume extractor
- ~ Soldering - Simple maintenance tips for your solder iron
- ~ A demonstration of soldering a header onto a breakout board
- ~ Introduction to protoboards
- ~ Reading a Schematic
- ~ Applying Ohm's Law

#### => Practice with Arduino pins :

- ~ Digital pins as output pins
- ~ Digital pins as input pins
- ~ Analogue pin
- ~ Serial communication(send data to receive data)
- ~ Time functionality management

#### => Arduino Programming :

- ~ Syntax, Program Flow, and Comments
- ~ Datatypes
- ~ Variables
- ~ Variable scope
- ~ Arithmetic Operators
- ~ Predefine functions
- ~ Custom functions
- ~ Creating custom functions and the return keyword
- ~ Constants
- ~ Introduction to control structures:
- ~ The "if" statement
- ~ The "while" statement
- ~ The "For" statement
- ~ The "Switch" statement
- ~ while loop
- ~ for loop

#### => Arduino advance programming :

- ~ Arrays
- ~ Strings

#### => Connect LCD display with arduino :

- ~ Introduction to LCD
- ~ Add the LCD Screen to Your Circuit
- ~ LCD wiring in 4-bit parallel mode
- ~ LCD demonstration sketch
- ~ Display sensor data in the LCD
- ~ Connect LCD using the I2C adaptor
- ~ Using the RGB LCD and buttons shield

#### => Serial Communication in Arduino :

- ~ Send data
- ~ Receive data

#### => Sensors with Arduino(s) :

- ~ Temperature sensor
- ~ Humidity sensor
- ~ Pressure sensor
- ~ Infrared motion sensor
- ~ Distance sensor(Ultrasonic sensor)
- ~ Sounds sensor
- ~ Detecting acceleration

#### => Wireless connectivity :

- ~ Bluetooth
- ~ Wi-Fi
- ~ Cloud
- ~ Using GPS with Arduino

#### => Robotics with Arduino :

- ~ Introduction: Robots
- ~ Transistors as Switches
- ~ DC Motor
- ~ H-Bridge
- ~ Servo Motor

#### => IOT with Arduino :

- ~ Introduction: The Internet
- ~ CART
- ~ Connecting to WiFi
- ~ GET a Web Page
- ~ JSON Primer

- ~ *Introduction: What is IoT?*
- ~ *Posting to ThingSpeak*
- ~ *Intro to IFTTT*
- ~ *Making Requests to IFTTT*
- ~ *Passing Arguments to IFTTT*
- ~ *IFTTT to ThingSpeak*

=> Major project :

- ~ *Bomb decoder game*
- ~ *Serial LCD screen*
- ~ *Ultrasonic people counter*
- ~ *OLED breath analyser*
- ~ *Ultrasonic soker*
- ~ *Fingerprint scanner*
- ~ *Ultrasonic robot*
- ~ *Internet controlled LED*
- ~ *Voice-controlled LED*
- ~ *GPS speedometer*
- ~ *Obstacle detection*

=> Project categories :

- ~ *Light Project*
- ~ *Sensor project*
- ~ *Sounds project*
- ~ *Power project*
- ~ *Security*
- ~ *Smart machine*
- ~ *USB project with Leonardo*
- ~ *Miscellaneous project*

=> Project Idea :

- ~ *Modem-based security system for restricted area*
- ~ *Campus fire monitoring systems*
- ~ *Light intensity control system*
- ~ *Dc motor control system*
- ~ *Temperature monitoring control system*
- ~ *Home autonomous system*
- ~ *Fingerprint-based autonomous system*
- ~ *Wireless irrigation system for agriculture field*

# Matlab Basic to Advance

---

Topic Name : PROGRAMMING

Sub-topic Name : MATLAB

Course link : <https://ineuron.ai/course/Matlab-Basic-to-Advance>

## Course Description :-

MATLAB(matrix laboratory) is a multi-paradigm numerical computing environment and fourth-generation programming language which is used by engineering and science students. In this course, we will start learning MATLAB from a beginner level, and will gradually move into more technical and advanced topics. This course is designed to be general in scope which means that it will be beneficial to students in any major. Once, passed a certain learning threshold, you will definitely enjoy MATLAB Programming. The key benefit of MATLAB is that it makes the programming available to everyone and is very fast to turn ideas into working products compared to some of the conventional programming languages such as Java, C, C++, visual basic, and others.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => MATLAB Graphical User Interface
- => Operations on matrices
- => Interacting with MATLAB and Graphics
- => Structures and Map Containers

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Course & Instructor Introduction :
  - ~ Introduction to course
  - ~ MATLAB Software Pricing and Online resources
  - ~ MATLAB Graphical User Interface
  - ~ Some Common Operations
- => Handling Variables and Creating Scripts :
  - ~ Let's lay foundations for understanding variables
  - ~ Different types of variables Strings characters and logical
  - ~ Creating scripts and understanding commenting and semicolon effect
  - ~ Data selection with the colon operator
- => Doing Basic Maths in MATLAB :
  - ~ Basic Maths addition multiplication subtraction and powers
  - ~ Understanding operation precedence
  - ~ Computing GCD LCM Permutations and Prime numbers
  - ~ Trigonometric functions
  - ~ Set operations Union intersection complement and others
  - ~ Computing statistics of the matrices
  - ~ Handling random numbers
  - ~ Cross and dot product
  - ~ Basic logical operation And, Or and Not
  - ~ Sign and absolute functions
  - ~ Converting numbers between different bases
  - ~ Discretizing your data
- => Operations on matrices :
  - ~ Determining unique elements
  - ~ Determining membership of elements to a matrix
  - ~ Shifting matrix elements
  - ~ Determinant inverse and diagonal elements
  - ~ Relational operations
  - ~ Commonly used Matrices
  - ~ Sorting matrix values

- ~ Size and length functions
- ~ Concatenating matrices
- ~ Finding non-zero elements
- ~ Frequencies of values within a vector

#### => Advance Math Functions with Symbolic Data Type :

- ~ Symbolic variables
- ~ Differentiation and integration using symbolic variables
- ~ Solving equations
- ~ Symbolic Functions

#### => Interacting with MATLAB and Graphics :

- ~ Input output commands
- ~ More input-output commands
- ~ Plotting Data
- ~ Plotting 3-D data
- ~ More on plotting options
- ~ Combining plots with hold on
- ~ Interacting with the plot using the brush tool
- ~ Creating plots with two y-axes
- ~ Animated line
- ~ Bar graphs
- ~ checking for the existence of files scripts folders functions or class
- ~ Manipulating Directory Part 1
- ~ Manipulating Directory Part 2
- ~ Processing a text file

#### => Importing data into matlab :

- ~ Importing data from excel to matlab
- ~ Importing data in different formats
- ~ Spread Sheet link Introduction and installation
- ~ Passing data between excel and MATLAB
- ~ Calling MATLAB functions from Excel

#### => MATLAB Programming :

- ~ Conditional if Statements Part 1
- ~ Conditional if Statements Part 2
- ~ For loops for iterating through your code
- ~ Nested For loops
- ~ While loops when you don't know the number of iterations
- ~ Breaking out from a loop before final condition
- ~ Continue statement for skipping iterations
- ~ Switching statements for selecting between options

#### => Making your own functions :

- ~ Creating custom build functions
- ~ Functions with inputs
- ~ Functions with multiple inputs and outputs
- ~ Returning from a function

#### => Sharing your MATLAB Results :

- ~ Sharing results with automatically generated reports
- ~ Sharing your results with live script

#### => Cell Data Type :

- ~ Creating and defining Cells
- ~ Accessing Data in a Cells
- ~ Adding and deleting elements from a cell
- ~ Concatenating Cells and Passing Cell Contents to a Function

#### => Tables and Time Tables :

- ~ Creating Tables
- ~ Adding Descriptions Units and Accessing individual columns
- ~ Selecting and reordering rows and columns
- ~ Sorting rows of a table
- ~ Setting Different Properties of the Table
- ~ Reading and Writing Tables into memory
- ~ Storing summary of a table
- ~ Adding and deleting rows from a table
- ~ Adding and deleting columns
- ~ Dealing with Missing Data
- ~ Creating Time tables
- ~ Properties Sorting and data selection in time tables
- ~ Concatenating timetables
- ~ Indexing and retrieving data based on row time

#### => Structures and Map Containers :

- ~ Creating Structures
- ~ Retrieving data from a field of a structure
- ~ Concatenating structures
- ~ Storing data from a structure field in a variable
- ~ More operations on structures
- ~ Creating Map Containers
- ~ Concatenation and more operations on containers Map Containers

#### => Data Type Conversion :

- ~ Conversion from other data types to cell
- ~ Conversion from Cell to other data types
- ~ Conversion from other data types to table
- ~ Conversion from other to table data type

# Mastering DSA with Python

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH PYTHON

Course link : <https://ineuron.ai/course/Mastering-DSA-with-Python>

## Course Description :-

This Python course on Data Structures and Algorithms covers data structures such as linked lists, stacks and queues, binary search trees, heaps, searching, and hashing. This course covers a variety of sorting algorithms, as well as their implementation and analysis. The following topics are covered with Python implementation in this Data Structures in Python course. Analysis of Algorithms, Big O notation, Time Complexity, Singly Linked List, Doubly linked list, Trees, Heaps, Hashing and Sorting algorithms.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Problem Solving
- => Data Structure Introduction
- => Recursion in depth
- => Linked List in depth
- => Circular Linked List in Depth
- => Doubly Linked List in Depth
- => Stack and Queue
- => Binary Search Tree
- => Hashing
- => AVL Tree
- => HEAP
- => Sorting algorithms

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

- => Introduction to DSA :
  - ~ Why we need Data structures and algorithms
  - ~ Time based approach
  - ~ Concept of Big O and graphs
  - ~ Data Structures and Algorithms HB
- => Problem Solving :
  - ~ Start with a challenge - reverse string
  - ~ Reverse a string - solution
  - ~ Interview approach to solve a problem
  - ~ Classic interview steps for DSA problems
- => Data Structure Introduction :
  - ~ Memory process - Stack and Heap
  - ~ Physical and logical data structures
  - ~ Abstract Data Types - ADT
- => Recursion in depth :

- ~ Introduction to recursion
- ~ Tracing the recursion tree
- ~ Trace tree assignment
- ~ Trace tree solution
- ~ Types of Recursion
- ~ Complex recursion tree
- ~ What is Factorial
- ~ Factorial program in Python
- ~ Fibonacci series THEORY
- ~ Fibonacci series and its version Python Code
- ~ What is Power Program
- ~ Power Program Python code
- ~ What is a Combination Program
- ~ Combination Program Python code
- ~ Classic Tower of Hanoi problem
- ~ Classic Tower of Hanoi Python code

#### => Linked List in depth :

- ~ Introduction to Linked List
- ~ Add value in linked list - cases
- ~ Push Append and insert in LinkedList - Python code
- ~ Deletion of linked list THEORY.
- ~ Deletion in linked list Python code
- ~ Delete complete linked list Python code
- ~ Count all nodes in linkedlistPython code
- ~ Reversing a linked list THEORY
- ~ Reversing a linked list Python code

#### => Circular Linked List in Depth :

- ~ Circular linked list THEORY
- ~ Circular Linked List push Python code
- ~ Traverse a circular linked list Python code
- ~ Deletion in circular linked list Python code
- ~ count nodes in circular linked list Python code
- ~ convert linked list to circular linked list Python code

#### => Doubly Linked List in Depth :

- ~ Theory for doubly linked list
- ~ Doubly linked list push Python code
- ~ Insert After in doubly linked list Python code
- ~ add to last in doubly linked list Python code
- ~ Traverse a doubly linked list Python code
- ~ Deleting a node in doubly linked list Python code

#### => Stack and Queue :

- ~ Stack - Push and Pop operation THEORY
- ~ Stack operations with Python code
- ~ Queue concept THEORY
- ~ Queue implementation in Python code
- ~ Circular queue THEORY
- ~ Circular queue Python code

#### => Binary Search Tree :

- ~ What is Binary Search tree and creation THEORY update
- ~ Insertion and Deletion in BST THEORY
- ~ InOrder Traversal of BST THEORY
- ~ Pre Order traversal in BST THEORY
- ~ Post order traversal in BST THEORY
- ~ Creating a Binary Search tree Python code
- ~ Insertion in BST Python code
- ~ deletion of key in BST Python code
- ~ inorder preorder and postorder traversal in BSTPython code

#### => Hashing :

- ~ What is Hashing THEORY
- ~ Hash chaining with linked list
- ~ Linear Hash Shifting
- ~ Square hash shifting

#### => AVL Tree :

- ~ What is AVL tree and height
- ~ Finding balance factor
- ~ Left Left and Right Right Rotation in AVL Tree
- ~ LR and RL rotation with 1 trick
- ~ Creating a AVL tree - Important
- ~ Deletion in AVL Tree.

#### => HEAP :

- ~ Heap - Max and min Heap
- ~ Insertion and deletion in HEAP

#### => Sorting algorithms :

- ~ Categories of sorts
- ~ Selection sort - Theory
- ~ Selection sort - Python Code
- ~ Bubble Sort - Theory
- ~ Bubble Sort - Python Code
- ~ Insertion sort - Theory
- ~ Insertion sort - Python Code
- ~ Quick Sort - Theory
- ~ Quick Sort - Theory part 2



- ~ Quick Sort - Python Code
- ~ Counting Sort - Theory
- ~ Merge Sort Theory
- ~ Merge sort Python code
- ~ Counting Sort - Python Code

# System Design with Design Patterns Tech Neuron

---

Topic Name : SYSTEM DESIGN

Sub-topic Name : SYSTEM DESIGN MASTERS

Course link : <https://ineuron.ai/course/System-Design-with-Design-Patterns-Tech-Neuron>

## Course Description :-

The software engineering interview process includes system design questions as a routine element of the process. The way you perform in these interviews reflects on your ability to work with complicated systems, which is reflected in the position and salary offered by the interviewing organisation. The purpose of this course is to help you master software engineering interviews.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Design principles
- => Introduction and types
- => Creational Patterns
- => Structural Patterns
- => Behavioural Patterns
- => Important System Design Concepts
- => System Design Problems
- => Designing Facebook Messenger
- => Designing Twitter
- => Designing Youtube
- => Designing Netflix

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Design principles :
  - ~ *DRY principles*
  - ~ *KISS principles*
  - ~ *SOLID principles*
  - ~ *CUPID principles*
- => Introduction and types :
  - ~ *OOPS overview*
  - ~ *The Singleton Pattern part 1*
  - ~ *The Singleton Pattern part 2*
  - ~ *The Singleton Pattern part 3*
- => Creational Patterns :
  - ~ *The Factory Pattern*
  - ~ *The Factory Method Pattern*
  - ~ *The Abstract Factory Pattern*
  - ~ *The Singleton Pattern*
  - ~ *The Builder Pattern*
  - ~ *The Prototype Pattern*
  - ~ *Summary of Creational Patterns*
- => Structural Patterns :
  - ~ *The Adapter Pattern*
  - ~ *The Bridge Pattern*
  - ~ *The Composite Pattern*
  - ~ *The Decorator Pattern*
  - ~ *The Faade Pattern*

- ~ *The Flyweight Pattern*
- ~ *The Proxy Pattern*
- ~ *Summary of Structural Patterns*

#### => Behavioural Patterns :

- ~ *Chain of Responsibility Pattern*
- ~ *The Command Pattern*
- ~ *The Interpreter Pattern*
- ~ *The Iterator Pattern*
- ~ *The Mediator Pattern*
- ~ *The Memento Pattern*
- ~ *The Observer Pattern*
- ~ *The State Pattern*
- ~ *The Strategy Pattern*
- ~ *The Template Pattern*
- ~ *The Visitor Pattern*
- ~ *Null Object pattern*

#### => Important System Design Concepts :

- ~ *System Design Basics*
- ~ *Key Characteristics of Distributed Systems*
- ~ *Load Balancing*
- ~ *ClientServer Model*
- ~ *Network Protocols*
- ~ *Storage*
- ~ *Latency And Throughput*
- ~ *Availability*
- ~ *Caching*
- ~ *Data Partitioning*
- ~ *Indexes*
- ~ *Replication*
- ~ *Sharding*
- ~ *Proxies*
- ~ *Redundancy*
- ~ *SQL vs. NoSQL*
- ~ *CAP Theorem and*
- ~ *PACELC Theorem*
- ~ *Consistent Hashing*
- ~ *Long Polling vs WebSockets vs Server Sent Events*
- ~ *Bloom Filters*
- ~ *Quorum, Leader and Follower, Heartbeat, Checksum*
- ~ *Rate Limiting*
- ~ *Logging And Monitoring*
- ~ *Security And HTTPS*
- ~ *API Design*

#### => System Design Problems :

- ~ *System Design Interviews: A step by step guide*
- ~ *Designing a URL Shortening service like TinyURL*
- ~ *Designing Pastebin*
- ~ *Designing Instagram*
- ~ *Designing Dropbox*
- ~ *Designing Facebook Messenger*
- ~ *Designing Twitter*
- ~ *Designing Youtube*
- ~ *Designing Netflix*
- ~ *Designing Typeahead Suggestion*
- ~ *Designing an API Rate Limiter*
- ~ *Designing Twitter Search*
- ~ *Designing a Web Crawler*
- ~ *Designing Facebooks Newsfeed*
- ~ *Designing Yelp or Nearby Friends*
- ~ *Designing Uber backend*
- ~ *Designing Ticketmaster*

# Power Automate

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Power-Automate>

## Course Description :-

This program is specialized in providing theoretical as well as practical understanding of the Microsoft Power Automate platform which allows you to learn how to automate certain process using Microsoft power automate. Course curriculum includes power automate features, dashboard, cloud functions and much more!

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Micosoft Power automate in details
- => Account Creation
- => Dashboard Overview
- => Cloud flow

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

- => Course Introduction :
  - ~ Overview Preview
  - ~ Power Automate detail Preview
- => Account Creation :
  - ~ Create account
- => Dashboard :
  - ~ Dashboard overview
- => Cloud Flow :
  - ~ Cloud flow
  - ~ Create Cloud flow

# Artificial Neural Network

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/Artificial-Neural-Network>

## Course Description :-

Artificial neural networks (ANNs) are computer systems that are modelled after the biological neural networks that make up animal brains. It processes data and creates patterns for use in decision-making in the same way that the human brain does. Using the most up-to-date frameworks, you'll learn Artificial neural networks, Transfer Learning, and more.

## Course Features :-

- => Source code
- => Roadmap
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Neural Network
- => Perceptron
- => Evaluation of Neural Network
- => Maths behind concepts of Neural Networks
- => Back Propagation
- => Problems faced while training Neural Network and its solution
- => Building solutions

## Requirements :-

- => Basic Programming Knowledge
- => A System with a decent internet connection
- => Your dedication

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

- => Introduction
- => AI | Deep Learning | Evolution of ANNs :
  - ~ Introduction Preview
  - ~ Introduction
- => Perceptron
- => Perceptron Implementation
- => Perceptron Implementation | Python scripting and packaging | Modular coding
- => Python logging basics in previous codes, docstrings
- => Python packaging | Github Actions | PyPI
- => Neural Network
- => ANN Derivation
- => ANN implementation using tf.keras
- => ANN implementation using python scripting
- => ANN implementation using python scripting continued
- => Callbacks in Tensorflow
- => ANN with Callbacks | Tensorboard | Early Stopping | Model Checkpointing
- => Mathematics in DL
- => THEORY: Vectors

- => THEORY Differentiation | Partial Diff | Gradients | Ascent and Descent
- => THEORY Problems in training NN | Vanishing and Exploding gradients
- => Tensorflow Framework
- => TF 2.x low-level API
- => TF 2.x low-level API PART 2
- => Activation Function
- => Activation Function - Started
- => Activation Function -continued
- => Activation function final
- => Weight initialization, Transfer learning, Batch Normalization
- => Weight initialization and Transfer learning
- => Batch Normalization: Theory and Practical
- => MLFlow
- => Optimizers, Regularization and Loss function
- => Fast Optimizers | Momentum Optimization
- => NAG
- => AdaGrad
- => RMS Prop | Adam
- => Regularization | Dropout | Loss function

# Complete Excel Course

---

Topic Name : DATA ANALYTICS

Sub-topic Name : EXCEL

Course link : <https://ineuron.ai/course/Complete-Excel-Course>

## Course Description :-

Excel is perhaps the most widely used spreadsheet on personal computers. It's simple to use for a variety of computations and comes with a Data Analysis Tool Pack and a collection of statistical tools. So, if you suddenly find yourself looking to undertake some statistical analysis, you may choose to start using Excel. So, in this course, you will study all there is to know about Excel, from the fundamentals to advanced topics, and you will have a thorough knowledge of its power and how to use it for data analysis and other applications.

## Course Features :-

- => Course Materials
- => Self Paced Learning
- => Lifetime Dashboard Access
- => Completion Certificate

## What you will learn :-

- => String functions on Excel
- => Mathematical functions on Excel
- => Logical & error functions on Excel
- => Excel formatting
- => Custom formatting
- => Conditional formatting
- => Charts in Excel
- => Data analysis using Excel
- => Pivot tables
- => Dashboarding in Excel

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Dr Nishtha Jain :

*~ I am a doctor by profession but a teacher by passion. I have been into the teaching profession for the last 3 years. I have been and am still a mentor for various courses which include technical as well as non-technical ones. These include MS-Excel, Tableau, Computer basics, Biology, English, etc. I love to learn, explore and share my knowledge to whatever extent possible. Being an ardent educator, I have always helped all my students and will continue to do the same.*

## Curriculum details :-

=> Microsoft Office in Brief :

- ~ Microsoft account creation
- ~ MS Office installation
- ~ Office Web version and Microsoft 365 free trial

=> What, Why and How of Excel :

- ~ MS Excel and why to use it
- ~ Use cases of excel in companies
- ~ Basic tasks in Excel

=> MS Excel Introduction and UI :

- ~ MS Excel Introduction
- ~ User interface or Parts of Excel, Part 1
- ~ User interface or Parts of Excel, Part 2

=> Working on Rows and Columns :

- ~ Inserting and Deleting Rows, Columns, and Cells
- ~ Hide Unhide rows and columns, Modify column and row size, Freeze and Unfreeze

=> Editing and customizing data :

- ~ Editing data and customizing it
- ~ Cut, Copy, Paste, Redo, Undo

- ~ *Clipboard, Split panes*
- => Paste special :
  - ~ *Use Paste Special, Part 1*
  - ~ *Use Paste Special, Part 2*
- => Working on worksheets :
  - ~ *Working on worksheets*
- => Basic formulas and functions :
  - ~ *Basic Formulas in Excel*
  - ~ *Basic functions in Excel*
- => Find, Select, Replace, Go to :
  - ~ *Find and Select*
  - ~ *Go to, Go to Special*
  - ~ *Find and Replace*
- => Filling, Copying :
  - ~ *FILL types in Excel*
  - ~ *Copying formulas without changing the cell references*
- => Cell referencing :
  - ~ *Absolute, mixed and relative cell referencing*
- => Subtotal :
  - ~ *Subtotal*
- => Text functions :
  - ~ *Text Functions, Part 1*
  - ~ *ROUND functions*
  - ~ *Text functions, Part 2*
  - ~ *Text functions, Part 3*
- => Logical or Comparative Operators :
  - ~ *Logical or Comparative Operators*
- => Text operations :
  - ~ *Text to columns, Part 1*
  - ~ *Text to columns, Part 2*
  - ~ *Textsplit function*
  - ~ *Textbefore function*
  - ~ *Textafter functions*
- => IF and IFs functions :
  - ~ *How to use IF function*
  - ~ *Some IF functions*
  - ~ *IFs functions*
- => Database Functions :
  - ~ *Database Functions*
- => Logical functions :
  - ~ *Logical functions along with IF function*
- => Speak cells, Show formulas :
  - ~ *Show and Hide formulas, Speak cells*
- => Basic unconditional formatting in Excel :
  - ~ *Basic unconditional formatting in Excel*
- => Protection in Excel :
  - ~ *Protecting Cells and Worksheets*
  - ~ *Protecting files and Workbooks*
- => Formula auditing :
  - ~ *Formula Auditing, Part 1*
  - ~ *Formula Auditing, Part 2*
- => Naming, Sorting and Filtering :
  - ~ *Naming Ranges*
  - ~ *Basic Sorting and Filtering*
  - ~ *Advanced Sorting and Filtering*
- => Tables in Excel :
  - ~ *Tables in Excel, Part 1*
  - ~ *Tables in Excel, Part 2*
- => Printing :
  - ~ *Printing, Part 1*
  - ~ *Printing, Part 2*
- => Data Validation and Dropdown :
  - ~ *Data Validation, Part 1*
  - ~ *Data Validation, Part 2*
- => Index, Match :
  - ~ *Index and Match functions*
- => LOOKUP in Excel :
  - ~ *HLOOKUP*
  - ~ *VLOOKUP*
  - ~ *Lookup*
  - ~ *XLOOKUP*
  - ~ *Lookup*



=> Conditional Formatting :

- ~ *Conditional formatting, Part 1*
- ~ *Conditional Formatting*

=> Date and Time Functions :

- ~ *Date and Time Functions*

=> Pivot Tables in Excel :

- ~ *Pivot Table, Part 1*
- ~ *Pivot table, Part 2*

=> Charts in Excel :

- ~ *Charts in Excel, Part 1*
- ~ *Charts in Excel, Part 2*
- ~ *Charts in Excel, Part 3*
- ~ *Charts in Excel, Part 4*

=> Excel Power Query :

- ~ *Power Query in Excel Part 1*
- ~ *Power query in Excel, Part 2*
- ~ *Power Query in Excel, Part 3*

=> Excel Dashboarding :

- ~ *Dashboard in Excel*

# Latex

---

Topic Name : PROGRAMMING

Sub-topic Name : LATEX

Course link : <https://ineuron.ai/course/Latex>

## Course Description :-

LaTeX is a document preparation system that is widely used in many scientific domains, including mathematics, statistics, computer science, engineering, chemistry, physics, economics, and linguistics. This is the place to start if you've never used LaTeX before or if it's been a while and you need a refresher. This course will get you started writing LaTeX right away with interactive exercises that you may complete online instead of downloading and installing LaTeX on your own computer.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Latex Online Environment
- => Mathematical equation and algorithms
- => Figures and Tables
- => Beamer

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Manjunatha A :

*~ Data Scientist with good experience in machine learning, deep learning, and Python programming. I was awarded the Gold medal in my Master's (MCA). I was also privileged to be honour with the Ace of innovation award. I was also one of the Finalists of SIH -2020 the world's largest hackathon. In my spare time, I enjoy sharing my technological abilities and knowledge through classes. I supervised over 500 students and assisted them in establishing careers in their industries. I also travel frequently.*

## Curriculum details :-

=> Introduction to Latex :

- ~ Course Introduction
- ~ Course Curriculum Overview
- ~ Course Outcome
- ~ Key Features of Latex

=> Setup and Installation :

- ~ Latex Environment setup
- ~ MikTex Installation
- ~ TexStudio Installation

=> Latex Online Environment :

- ~ Introduction to Latex online editor
- ~ Exploring Overleaf dashboard
- ~ First project creation in Overleaf

=> Latex Basics :

- ~ Introduction to Latex
- ~ Exploring Latex Dashboard
- ~ Latex commands and file structure
- ~ First project creation in Latex
- ~ Text Formatting in Latex
- ~ Lists in Latex
- ~ Installing missing packages

=> Mathematical equation and algorithms :

- ~ Latex Mathematical notations
- ~ Mathematical symbols in Latex
- ~ Mathematical equations
- ~ Arithmetic, subscript and accent

- ~ *Binomial, Integration and delimiter*
- ~ *Simple, annotate and case equations*
- ~ *Summation, product and matrices*
- ~ *Algorithm and pseudocode*
- ~ *Algorithm and pseudocode practical demonstration*
- ~ *Conditional statement*
- ~ *Loops(For,While)*

=> **Figures and Tables :**

- ~ *Representing image*
- ~ *Accessing image with different sources*
- ~ *Introduction to Table and Table creation*
- ~ *Table alignment and centering*
- ~ *Complex table*
- ~ *Table creation using TexStudio*

=> **Bibliography :**

- ~ *Introduction to Bibliography*
- ~ *Bibliography styles*

=> **Beamer :**

- ~ *Introduction to Beamer*
- ~ *Beamer Title creation*
- ~ *Create and organize frames*
- ~ *Beamer Table of contents*
- ~ *Formatting Text in beamer*
- ~ *Effects in presentation*
- ~ *Themes in Beamer*

=> **Scientific Report Writing using Templates :**

- ~ *Research paper templates*
- ~ *Splitting document into multiple files*

=> **CV and Poster creation :**

- ~ *Overleaf CV and poster creation*

# Data Structure and Algorithm Projects

---

Topic Name : DATA STRUCTURE

Sub-topic Name : ADVANCED DSA

Course link : <https://ineuron.ai/course/Data-Structure-and-Algorithm-Projects>

## Course Description :-

Algorithms Programming is a must-have skill in the field of software development. Learners have to analyse and construct algorithms for finding, sorting, and indexing data, as well as create trees and graphs and deal with intractability. Industry-level projects that will help you upscale your skills for handling and managing real-world projects.

## Course Features :-

- => Project source code
- => Covering multiple domains
- => Interview questions
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Create efficient algorithms
- => Assess the complexity of the time and memory
- => Architecture design
- => Solution design
- => Tech stack integration
- => Software development

## Requirements :-

- => Detailed knowledge of Data Structure and Algorithms
- => Knowledge of Python programming language.
- => A system with internet connection.
- => Your dedication

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

=> URL Shortener :

- ~ Introduction & understanding of the Algorithm behind the project Preview
- ~ Implementation

=> Data Structure Implementation: BFS and DFS :

- ~ Introduction & understanding of the Algorithm behind the project
- ~ Implementation

=> Topological Sorting of Graph :

- ~ Introduction & understanding of the Algorithm behind the project
- ~ Implementation

=> Phone Directory Application :

- ~ Introduction & understanding of the Algorithm behind the project
- ~ Implementation

# Full Stack Data Science Feb'21 Batch

---

Topic Name : DATA SCIENCE

Sub-topic Name : FULL STACK DATA SCIENCE

Course link : <https://ineuron.ai/course/Full-Stack-Data-Science-Feb'21-Batch>

## Course Description :-

This is a data science full stack live mentor led certification program along with full time one-year internship provided by iNeuron intelligence private limited, where you will learn all the stack required to work in data science, data analytics and big data industry including ML ops and cloud infrastructure and real time industry project and product development along with iNeuron product development team and you will contribute on various level with iNeuron .

## Course Features :-

- => Full stack Data Science masters certification
- => Job guarantee otherwise refund
- => One year of internship
- => Online Instructor-led learning: Live teaching by instructors
- => 56 + hands-on industry real-time projects.
- => 400 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access.
- => Doubt clearing one to one
- => Doubt clearing through mail and skype support team
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building
- => Career guidance
- => Interview Preparation
- => Regular assessment

## What you will learn :-

- => Python
- => Stats
- => Machine learning
- => Deep learning
- => Computer vision
- => Natural language processing
- => Data analytics
- => Big data
- => ML ops
- => Cloud
- => Data structure and algorithm
- => Architecture
- => Domain wise project
- => Databases
- => Negotiations skills
- => Mock interview
- => Interview preparation
- => Resume building after every module

## Requirements :-

- => Dedication
- => Computer with i3 and above configuration

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Course introduction :

- ~ a. course overview and dashboard description
- ~ b. Introduction of data science and its application in day to day life
- ~ c. Programming language overview
- ~ d. Installation (tools: sublime, vscode, pycharm, anaconda, atom, jupyter notebook, kite)
- ~ e. Virtual environment
- ~ f. Why python

=> Python basic :

- ~ a. Introduction of python and comparison with other programming language
- ~ b. Installation of anaconda distribution and other python ide
- ~ c. Python objects, number & Booleans, strings.
- ~ d. Container objects, mutability of objects
- ~ e. Operators - arithmetic, bitwise, comparison and assignment operators, operators precedence and associativity
- ~ f. Conditions (if else, if-elif-else), loops (while, for)
- ~ g. Break and continue statement and range function

=> String objects :

- ~ a. basic data structure in python
- ~ b. String object basics
- ~ c. String inbuilt methods
- ~ d. Splitting and joining strings
- ~ e. String format functions

=> List object basics :

- ~ a. List methods
- ~ b. List as stack and queues
- ~ c. List comprehensions

=> Tuples, set, dictionaries & its function :

- ~ Dictionary object methods
- ~ Dictionary comprehensions
- ~ Dictionary view objects
- ~ Functions basics, parameter passing, iterators.
- ~ Generator functions
- ~ Lambda functions
- ~ Map, reduce, filter functions.

=> Memory management :

- ~ Multithreading
- ~ Multiprocessing

=> OOps concepts :

- ~ oops basic concepts.
- ~ Creating classes
- ~ Pillars of oops
- ~ Inheritance
- ~ Polymorphism
- ~ Encapsulation
- ~ Abstraction
- ~ Decorator
- ~ Class methods and static methods
- ~ Special (magic/dunder) methods
- ~ Property decorators - getters, setters, and deletes

=> Files :

- ~ Working with files
- ~ Reading and writing files
- ~ Buffered read and write
- ~ Other file methods.
- ~ Logging, debugger
- ~ Modules and import statements

=> Exception handling difference between exceptions and error :

- ~ Exceptions handling with try-except
- ~ Custom exception handling
- ~ List of general use exception
- ~ Best practice exception handling

=> Gui framework :

- ~ What is desktop and standalone application
- ~ Use of desktop app
- ~ Examples of desktop app
- ~ Tinker
- ~ Kivy

=> Database :

- ~ SQLite
- ~ MySQL
- ~ Mongo dB
- ~ NoSQL - Cassandra

=> Web API :

- ~ What is web API
- ~ Difference b/w API and web API
- ~ Rest and soap architecture
- ~ Restful services

=> Flask :

- ~ Flask introduction
- ~ Flask application
- ~ Open link flask
- ~ App routing flask
- ~ Url building flask
- ~ Http methods flask
- ~ Templates flask
- ~ Flask project: food app
- ~ Postman
- ~ Swagger

=> Django :

- ~ Django introduction
- ~ Django project: weather app
- ~ Django project: memes generator
- ~ Django project: blog app
- ~ Django project in cloud

=> Stream lit :

- ~ Stream lit introduction
- ~ Stream lit project structure
- ~ Stream lit project in cloud

=> Pandas basic :

- ~ Python pandas - series
- ~ Python pandas data frame
- ~ Python pandas panel
- ~ Python pandas - basic functionality
- ~ Reading data from different file system

=> Pandas advance :

- ~ Python pandas re indexing python
- ~ Pandas iteration
- ~ Python pandas sorting.
- ~ Working with text data options & customization
- ~ Indexing & selecting
- ~ Data statistical functions
- ~ Python pandas - window functions
- ~ Python pandas - date functionality
- ~ Python pandas time delta
- ~ Python pandas - categorical data
- ~ Python pandas visualization
- ~ Python pandas - iotools

=> Dask :

- ~ Dask Array
- ~ Dask Bag
- ~ Dask DataFrame
- ~ Dask Delayed
- ~ Dask Futures
- ~ Dask API
- ~ Dask SCHEDULING
- ~ Dask Understanding Performance
- ~ Dask Visualize task graphs
- ~ Dask Diagnostics (local)
- ~ Dask Diagnostics (distributed)
- ~ Dask Debugging
- ~ Dask Ordering

=> Python numpy :

- ~ Numpy - ND array object.
- ~ Numpy - data types.
- ~ Numpy - array attributes.
- ~ Numpy - array creation routines.
- ~ Numpy - array from existing.
- ~ Data array from numerical ranges.
- ~ Numpy - indexing & slicing.
- ~ Numpy advanced indexing.
- ~ Numpy broadcasting.
- ~ Numpy - iterating over array.
- ~ Numpy - array manipulation.
- ~ Numpy - binary operators.
- ~ Numpy - string functions.
- ~ Numpy - mathematical functions.
- ~ Numpy - arithmetic operations.
- ~ Numpy - statistical functions.
- ~ Sort, search & counting functions.

- ~ *Numpy - byte swapping.*
- ~ *Numpy - copies & views.*
- ~ *Numpy - matrix library.*
- ~ *Numpy - linear algebra*

=> Visualization :

- ~ *Matplotlib*
- ~ *Seaborn*
- ~ *Cufflinks*
- ~ *Plotly*
- ~ *Bokeh*

=> Statistics basic :

- ~ *Introduction to basic statistics terms*
- ~ *Types of statistics*
- ~ *Types of data*
- ~ *Levels of measurement*
- ~ *Measures of central tendency*
- ~ *Measures of dispersion*
- ~ *Random variables*
- ~ *Set*
- ~ *Skewness*
- ~ *Covariance and correlation*

=> Probability distribution function :

- ~ *Probability density/distribution function*
- ~ *Types of the probability distribution*
- ~ *Binomial distribution*
- ~ *Poisson distribution*
- ~ *Normal distribution (Gaussian distribution)*
- ~ *Probability density function and mass function*
- ~ *Cumulative density function*
- ~ *Examples of normal distribution*
- ~ *Bernoulli distribution*
- ~ *Uniform distribution*
- ~ *Z stats*
- ~ *Central limit theorem*
- ~ *Estimation*

=> Statistics advance :

- ~ *a Hypothesis*
- ~ *Hypothesis testings mechanism*
- ~ *P-value*
- ~ *T-stats*
- ~ *Student t distribution*
- ~ *T-stats vs. Z-stats: overview*
- ~ *When to use a t-tests vs. Z-tests*
- ~ *Type 1 & type 2 error*
- ~ *Bayes statistics (Bayes theorem)*
- ~ *Confidence interval(ci)*
- ~ *Confidence intervals and the margin of error*
- ~ *Interpreting confidence levels and confidence intervals*
- ~ *Chi-square test*
- ~ *Chi-square distribution using python*
- ~ *Chi-square for goodness of fit test*
- ~ *When to use which statistical distribution?*
- ~ *Analysis of variance (anova)*
- ~ *Assumptions to use anova*
- ~ *Anova three type*
- ~ *Partitioning of variance in the anova*
- ~ *Calculating using python*
- ~ *F-distribution*
- ~ *F-test (variance ratio test)*
- ~ *Determining the values of f*
- ~ *F distribution using python*

=> Linear algebra :

- ~ *linear algebra*
- ~ *Vector*
- ~ *Scaler*
- ~ *Matrix*
- ~ *Matrix operations and manipulations*
- ~ *Dot product of two vectors*
- ~ *Transpose of a matrix*
- ~ *Linear independence of vectors*
- ~ *Rank of a matrix*
- ~ *Identity matrix or operator*
- ~ *Determinant of a matrix*
- ~ *Inverse of a matrix*
- ~ *Norm of a vector*
- ~ *Eigenvalues and eigenvectors*
- ~ *Calculus*

=> Solving stats problem with python

=> Stats problem implementation with spicy

=> Introduction to machine learning :

- ~ *AI vs ml vs dl vs ds*
- ~ *Supervised, unsupervised, semi-supervised, reinforcement learning*
- ~ *Train, test, validation split*



- ~ Performance
- ~ Overfitting, under fitting
- ~ Bias vs variance

#### => Feature engineering :

- ~ Handling missing data
- ~ Handling imbalanced data
- ~ Up-sampling
- ~ Down-sampling
- ~ Smote
- ~ Data interpolation
- ~ Handling outliers
- ~ Filter method
- ~ Wrapper method
- ~ Embedded methods
- ~ Feature scaling
- ~ Standardization
- ~ Mean normalization
- ~ Min-max scaling
- ~ Unit vector
- ~ Feature extraction
- ~ Pca (principle component analysis)
- ~ Data encoding
- ~ Nominal encoding
- ~ One hot encoding
- ~ One hot encoding with multiple categories
- ~ Mean encoding
- ~ Ordinal encoding
- ~ Label encoding
- ~ Target guided ordinal encoding
- ~ Covariance
- ~ Correlation check
- ~ Pearson correlation coefficient
- ~ Spearmans rank correlation
- ~ Vif

#### => Feature selection :

- ~ Feature selection
- ~ Recursive feature elimination
- ~ Backward elimination
- ~ Forward elimination

#### => Exploratory data analysis :

- ~ Feature engineering and selection.
- ~ Analyzing bike sharing trends.
- ~ Analyzing movie reviews sentiment.
- ~ Customer segmentation and effective cross selling.
- ~ Analyzing wine types and quality.
- ~ Analyzing music trends and recommendations.
- ~ Forecasting stock and commodity prices

#### => Regression :

- ~ Linear regression
- ~ Gradient descent
- ~ Multiple linear regression
- ~ Polynomial regression
- ~ R square and adjusted r square
- ~ Rmse , mse, mae comparison
- ~ Regularized linear models
- ~ Ridge regression
- ~ Lasso regression
- ~ Elastic net
- ~ Complete end-to-end project with deployment on cloud and ui

#### => Logistics regression :

- ~ Logistics regression in-depth intuition
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Hyper parameter tuning
- ~ Grid search cv
- ~ Randomize search cv
- ~ Data leakage
- ~ Confusion matrix
- ~ Precision, recall, f1 score ,roc, auc
- ~ Best metric selection
- ~ Multiclass classification in lr
- ~ Complete end-to-end project with deployment in multi cloud platform

#### => Decision tree :

- ~ Decision tree classifier
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Confusion matrix
- ~ Precision, recall, f1 score ,roc, auc
- ~ Best metric selection
- ~ Decision tree repressor
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Performance metrics
- ~ Complete end-to-end project with deployment in multi cloud platform

=> Support vector machines :

- ~ Linear svm classification
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Soft margin classification
- ~ Nonlinear svm classification
- ~ Polynomial kernel
- ~ Gaussian, rbf kernel
- ~ Data leakage
- ~ Confusion matrix
- ~ precision, recall, f1 score, roc, auc
- ~ Best metric selection
- ~ Svm regression
- ~ In-depth mathematical intuition
- ~ In-depth geometrical intuition
- ~ Complete end-to-end project with deployment

=> Nave Bayes :

- ~ Bayes theorem
- ~ Multinomial nave Bayes
- ~ Gaussian nave Bayes
- ~ Various type of Bayes theorem and its intuition
- ~ Confusion matrix
- ~ precision, recall, f1 score, roc, auc
- ~ Best metric selection
- ~ Complete end-to-end project with deployment

=> Ensemble techniques and its types :

- ~ Definition of ensemble techniques
- ~ Bagging technique
- ~ Bootstrap aggregation
- ~ Random forest (bagging technique)
- ~ Random forest repressor
- ~ Random forest classifier
- ~ Complete end-to-end project with deployment

=> Boosting :

- ~ Boosting technique
- ~ Ada boost
- ~ Gradient boost
- ~ Xgboost
- ~ Complete end-to-end project with deployment

=> Stacking :

- ~ Stacking technique
- ~ Complete end-to-end project with deployment

=> Knn :

- ~ Knn classifier
- ~ Knn repressor
- ~ Variants of knn
- ~ Brute force knn
- ~ K-dimension tree
- ~ Ball tree
- ~ Complete end-to-end project with deployment

=> Dimensionality reduction :

- ~ The curse of dimensionality
- ~ Dimensionality reduction technique
- ~ Pca (principle component analysis)
- ~ Mathematics behind pca
- ~ Scree plots
- ~ Eigen-decomposition approach

=> Clustering :

- ~ Clustering and their types
- ~ K-means clustering
- ~ K-means++
- ~ Batch k-means
- ~ Hierarchical clustering
- ~ Dbscan
- ~ Evaluation of clustering
- ~ Homogeneity, completeness and v-measure
- ~ Silhouette coefficient
- ~ Davies-bouldin index
- ~ Contingency matrix
- ~ Pair confusion matrix
- ~ Extrinsic measure
- ~ Intrinsic measure
- ~ Complete end-to-end project with deployment

=> Anomaly detection :

- ~ Anomaly detection types
- ~ Anomaly detection applications
- ~ Isolation forest anomaly detection algorithm
- ~ Density-based anomaly detection (local outlier factor) algorithm
- ~ Support vector machine anomaly detection algorithm
- ~ Dbscan algorithm for anomaly detection
- ~ Complete end-to-end project with deployment

=> Time-series :

- ~ What is a time series?
- ~ Old techniques
- ~ Arima
- ~ Acf and pacf
- ~ Time-dependent seasonal components.
- ~ Autoregressive (ar),
- ~ Moving average (ma) and mixed arma- modeler.
- ~ The random walk model.
- ~ Box-jenkins methodology.
- ~ Forecasts with arima and var models.
- ~ Dynamic models with time-shifted explanatory variables.
- ~ The koyck transformation.
- ~ Partial adjustment and adaptive expectation models.
- ~ Granger's causality tests.
- ~ Stationarity, unit roots and integration
- ~ Time series model performance
- ~ Various approach to solve time series problem
- ~ Complete end-to-end project with deployment
- ~ Prediction of nifty stock price and deployment

=> NLP basic :

- ~ Tokenization
- ~ Pos tags and chunking
- ~ Stop words
- ~ Stemming and lemmatization
- ~ Named entity recognition (ner)
- ~ Word vectorization (word embedding)
- ~ Tfidf
- ~ Complete end-to-end project with deployment

=> Machine learning pipeline :

- ~ Aws segmaker
- ~ Aure ml studio
- ~ Ml flow
- ~ Kube flow

=> Model retraining approach

=> Auto ML :

- ~ H2o
- ~ Pycaret
- ~ Auto sklearn
- ~ Auto time series
- ~ Auto viml
- ~ Auto gluon
- ~ Auto viz
- ~ Tpot
- ~ Auto neuro

=> Neural network a simple perception. :

- ~ Detail mathematical explanation
- ~ Neural network overview and its use case.
- ~ Various neural network architect overview.
- ~ Use case of neural network in nlp and computer vision.
- ~ Activation function -all name
- ~ Multilayer network.
- ~ Loss functions. - all 10
- ~ The learning mechanism.
- ~ Optimizers. - all 10
- ~ Forward and backward propagation.
- ~ Weight initialization technique
- ~ Vanishing gradient problem
- ~ Exploding gradient problem
- ~ Visualization of nn

=> Hardware setup GPU :

- ~ Gpu introduction.
- ~ Various type of gpu configuration.
- ~ Gpu provider and its pricing.
- ~ Paper space gpu setup.
- ~ Running model in gpu

=> Tensor flow installation environment setup for deep learning :

- ~ Colab pro setup
- ~ Tensor flow installation 2.0 .
- ~ Tensor flow installation 1.6 with virtual environment.
- ~ Tensor flow 2.0 function.
- ~ Tensor flow 2.0 neural network creation.
- ~ Tensor flow 1.6 functions.
- ~ Tensor flow 1.6 neural network and its functions.
- ~ Keras introduction.
- ~ Keras in-depth with neural network creation.
- ~ Mini project in tensorflow.
- ~ Tensorspace
- ~ Tensorboard integration
- ~ Tensorflow playground
- ~ Netron

=> Pytorch :

- ~ pytorch installation.
- ~ Pytorch functional overview.

~ Pytorch neural network creation.

=> Mxnet :

~ Mxnet installation  
~ Mxnet in depth function overview  
~ Mxnet model creation and training

=> Kears tuner :

~ Keras tuner installation and overview  
~ Finding best parameter from keras tuner  
~ Keras tuner application across various neural network

=> Cnn overview :

~ Cnn definition  
~ Various cnn based architecture  
~ Explanation end to end cnn network  
~ Cnn explainer  
~ Training cnn  
~ Deployment in azure cloud  
~ Performance tuning of cnn network

=> Advance computer vision part 1 :

~ Various cnn architecture with research paper and mathematics  
~ Lenet-5 variants with research paper and practical  
~ Alexnet variants with research paper and practical  
~ Googlenet variants with research paper and practical  
~ Transfer learning  
~ Vggnet variants with research paper and practical  
~ Resnet variants with research paper and practical  
~ Inception net variants with research paper and practical  
~ Darknet variants with research paper and practical

=> Advance computer vision part 2 :

~ Object detection in-depth  
~ Transfer learning  
~ Rcnn with research paper and practical  
~ Fast rcnn with research paper and practical  
~ Faster r cnn with research paper and practical  
~ Ssd with research paper and practical  
~ Ssd lite with research paper and practical

=> Training of custom object detection :

~ Tfod introduction  
~ Environment setup with tfod  
~ Gpu vs tpu vs cpu  
~ Various gpu comparison

=> Advance computer vision part 3 :

~ Yolo v1 with research paper and practical  
~ Yolo v2 with research paper and practical  
~ Yolo v3 with research paper and practical  
~ Yolo v4 with research paper and practical  
~ Yolo v5 with research paper and practical  
~ Retina net  
~ Face net  
~ Detectron2 with practical and live testing

=> Object segmentation :

~ Semantic segmentation  
~ Panoptic segmentation  
~ Masked rcnn  
~ Practical with detectron  
~ Practical with tfod

=> Object tracking :

~ Detail of object tracking  
~ Kalman filtering  
~ Sort  
~ Deep sort  
~ Object tracking live project with live camera testing

=> OCR :

~ Introduction to ocr  
~ Various framework and api for ocr  
~ Practical implementation of ocr

=> Advance NLP with deep-learning :

~ Overview computational linguistic.  
~ History of nlp.  
~ Why nlp  
~ Use of nlp

=> Text processing importing text. :

~ Web scrapping.  
~ Text processing  
~ Understanding regex.  
~ Text normalization  
~ Word count.  
~ Frequency distribution.  
~ Text annotation.  
~ Use of annotator.  
~ String tokenization

- ~ *Annotator creation.*
- ~ *Sentence processing.*
- ~ *Lemmatization in text processing*
- ~ *Pos.*
- ~ *Named entity recognition*
- ~ *Dependency parsing in text.*
- ~ *Sentimental analysis*

=> **Spacy :**

- ~ *Spacy overview.*
- ~ *Spacy function*
- ~ *Spacy function implementation in text processing.*
- ~ *Pos tagging, challenges and accuracy.*
- ~ *Entities and named entry recognition*
- ~ *Interpolation, language models*
- ~ *Nltk*
- ~ *Text blob*
- ~ *Stanford nlp*

=> **RNN :**

- ~ *Recurrent neural networks.*
- ~ *Long short term memory (lstm)*
- ~ *Bi lstm.*
- ~ *Stacked lstm*
- ~ *Gru implementation.*
- ~ *Building a story writer using character level rnn.*

=> **Word embedding :**

- ~ *Word embedding*
- ~ *Co-occurrence vectors*
- ~ *Word2vec*
- ~ *Doc2vec*

=> **Attention based model :**

- ~ *Seq 2 seq.*
- ~ *Encoders and decoders.*
- ~ *Attention mechanism.*
- ~ *Attention neural networks*
- ~ *Self-attention*

=> **Transfer learning in nlp :**

- ~ *Introduction to transformers.*
- ~ *Bert model.*
- ~ *Elmo model.*
- ~ *Gpt1 model*
- ~ *Gpt2 model.*
- ~ *Albert model.*
- ~ *Distilbert model*

=> **Deployment of model and performance tuning :**

- ~ *Deep learning model deployment strategies.*
- ~ *Deep learning project architecture*
- ~ *Deep learning model deployment phase.*
- ~ *Deep learning model retraining phase.*
- ~ *Deep learning model deployment in aws.*
- ~ *Deep learning model deployment in azure.*
- ~ *Deep learning model deployment in gcloud.*

=> **Big data introduction :**

- ~ *What is big data?*
- ~ *Big data application*
- ~ *Big data pipeline*

=> **Hadoop :**

- ~ *Hadoop introduction*
- ~ *Hadoop setup and installation*

=> **Spark :**

- ~ *Spark*
- ~ *Spark overview.*
- ~ *Spark installation.*
- ~ *Spark rdd.*
- ~ *Spark data frame.*
- ~ *Spark architecture.*
- ~ *Spark ml lib*
- ~ *Spark NLP*
- ~ *Spark linear regression*
- ~ *Spark logistic regression*
- ~ *Spark decision tree*
- ~ *Spark naive bayes*
- ~ *Spark xg boost.*
- ~ *Spark time series*
- ~ *Spark deployment in local server*
- ~ *Spark job automation with*
- ~ *Scheduler*

=> **Kafka :**

- ~ *Kafka introduction*
- ~ *Kafka installation*
- ~ *Spark streaming*
- ~ *Spark with Kafka*

=> Tableau :

- ~ Talking about Business Intelligence
- ~ Tools and Methodologies used in BI
- ~ Why Visualization is getting more popular
- ~ Why Tableau?
- ~ Gartner Magic Quadrant of Market Leaders
- ~ Future business impact of BI
- ~ Tableau Products
- ~ Tableau Architecture
- ~ BI Project Execution
- ~ Tableau Installation in local system
- ~ Introduction to Tableau Prep
- ~ Tableau Prep Builder User Interface
- ~ Data Preparation techniques using Tableau Prep Builder tool
- ~ How to connect Tableau with different data source
- ~ Visual Segments
- ~ Visual Analytics in depth
- ~ Filters, Parameters & Sets
- ~ Tableau Calculations using functions
- ~ Tableau Joins
- ~ Working with multiple data source (Data Blending)
- ~ Building Predictive Models
- ~ Dynamic Dashboards and Stories
- ~ Sharing your Reports
- ~ Tableau Server
- ~ User Security
- ~ Scheduling

=> Power BI :

- ~ Power BI introduction and overview
- ~ Key Benefits of Power BI
- ~ Power BI Architecture
- ~ Power BI Process
- ~ Components of Power BI
- ~ Power BI - Building Blocks
- ~ Power BI vs other BI tools
- ~ Power Installation
- ~ Overview of Power BI Desktop
- ~ Data Sources in Power BI Desktop
- ~ Connecting to a data Sources
- ~ Query Editor in Power BI
- ~ Views in Power BI
- ~ Field Pane
- ~ Visual Pane
- ~ Custom Visual Option
- ~ Filters
- ~ Introduction to using Excel data in Power BI
- ~ Exploring live connections to data with Power BI
- ~ Connecting directly to SQL Azure, HD Spark, SQL Server Analysis Services/ My SQL
- ~ Import Power View and Power Pivot to Power BI
- ~ Power BI Publisher for Excel
- ~ Content packs
- ~ Introducing Power BI Mobile
- ~ Power Query Introduction
- ~ Query Editor Interface
- ~ Clean and Transform your data with Query Editor
- ~ Data Type
- ~ Column Transformations vs Adding Columns
- ~ Text Transformations
- ~ Cleaning irregularly formatted data -Transpose
- ~ Date and Time Calculations
- ~ Advance editor: Use Case
- ~ Query Level Parameters
- ~ Combining Data Merging and Appending
- ~ Data Modelling
- ~ Calculated Columns
- ~ Measures/New Quick Measures
- ~ Calculated Tables
- ~ Optimizing Data Models
- ~ Row Context vs Set Context
- ~ Cross Filter Direction
- ~ Manage Data Relationship
- ~ Why is DAX important?
- ~ Advanced calculations using Calculate functions
- ~ DAX queries

=> Reinforcement Learning

Project details :-

=> Python Project :

- ~ Weeding script
- ~ Image resizing
- ~ Jupyter notebook merging, reading etc.
- ~ Sending emails
- ~ Weather app
- ~ Memes generator
- ~ Food log app
- ~ Web scrapping

- ~ Web crawlers for image data sentiment analysis and product review sentiment analysis.
- ~ Integration with web portal.
- ~ Integration with rest api, web portal and mongo db. on azure
- ~ Deployment on web portal on azure.
- ~ Text mining
- ~ Social media data churn
- ~ Mass copy, paste

=> Chatbot projects :

- ~ Chatbot using Microsoft Luis
- ~ Chatbot using google dialog flow
- ~ Chatbot using amazon lex
- ~ Chatbot using rasa nlu
- ~ Deployment of Chabot with web , telegram , WhatsApp, skype

=> Major projects :

- ~ Healthcare analytics prediction of medicines based on Fitbit band.
- ~ Revenue forecasting for startups.
- ~ Prediction of order cancellation at the time of ordering inventories.
- ~ anomaly detection in inventory packaged material.
- ~ Fault detection in wafers based on sensor data.
- ~ Demand forecasting for fmcg product.
- ~ Threat identification in security system.
- ~ Defect detection in vehicle engine.
- ~ Food price forecasting with zomato dataset.
- ~ Fault detection in wafers based on sensor data.
- ~ Cement strength reg.
- ~ Credit card fraud.
- ~ Forest cover classification.
- ~ Fraud detection.
- ~ Income prediction.
- ~ Mushroom classifier.
- ~ phishing classifier
- ~ Thyroid detection.
- ~ Visibility climate

=> Computer vision project :

- ~ Traffic surveillance system.
- ~ Object identification.
- ~ Object tracking.
- ~ Object classification.
- ~ Tensorflow object detection.
- ~ Image to text processing.
- ~ Speech to speech analysis.
- ~ Vision based attendance system

=> Mini NLP project :

- ~ Machine translation.
- ~ Abstractive text summarization.
- ~ Keyword spotting.
- ~ Language modelling.
- ~ Document summarization

=> Nlp transfer learning project :

- ~ Deployment and integration with UI machine translation.
- ~ Question answering (like chat bot)
- ~ Sentiment analysis imdb.
- ~ Text search (with synonyms).
- ~ Text classifications.
- ~ Spelling corrector.
- ~ Entity (person, place or brand) recognition.
- ~ Text summarization.
- ~ Text similarity (paraphrase).
- ~ Topic detection.
- ~ Language identification.
- ~ Document ranking.
- ~ Fake news detection
- ~ Plagiarism checker
- ~ Text summarization extractive
- ~ Text summarization abstractive.

=> NLP end to end project with architecture and deployment :

- ~ Movie review using bert
- ~ Ner using bert
- ~ Pos bert
- ~ Text generation gpt 2
- ~ Text summarization xlnet
- ~ Abstract bert
- ~ Machine translation
- ~ Nlp text summarization custom
- ~ Keras/tensorflow
- ~ Language identification
- ~ Text classification using fast bert
- ~ Neuralcore
- ~ Detecting fake text using gltr with bert and gpt2
- ~ Fake news detector using gpt2
- ~ Python plagiarism checker type a message
- ~ Question answering

# DSA with Java

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH JAVA

Course link : <https://ineuron.ai/course/DSA-with-Java>

## Course Description :-

This course has been designed to help you become a complete and professional Java developer at the conclusion of the course, rather than only teaching essential Java skills. After completing this course, you will have a thorough understanding of various Data Structures and Algorithms in Java which will further enhance your career as a java developer.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Problem Solving
- => Time-based DSA
- => Big O notation
- => Time and space complexity
- => Recursion
- => Power program theory
- => Combination theory
- => Stacks, queues, linked lists, trees
- => Searching, sorting, hashing

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to DSA :
  - ~ Why we need Data structures and algorithms Preview
  - ~ Time based approach Preview
  - ~ Concept of Big O and graphs Preview
  - ~ Data Structures and Algorithms HB
- => Problem Solving :
  - ~ Start with a challenge - reverse string
  - ~ Reverse a string - solution
  - ~ Interview approach to solve a problem
  - ~ Classic interview steps for DSA problems
- => Data Structure Introduction :
  - ~ Memory process - Stack and Heap
  - ~ Physical and logical data structures
  - ~ Abstract Data Types - ADT Preview
- => Recursion in depth :
  - ~ Introduction to recursion
  - ~ Tracing the recursion tree
  - ~ Trace tree assignment
  - ~ Trace tree solution
  - ~ Types of Recursion



- ~ Complex recursion tree
- ~ What is Factorial
- ~ DSA08 Factorial program in JAVA
- ~ Fibonacci series THEORY
- ~ Fibonacci series and its version JAVA Code
- ~ What is Power Program
- ~ Power Program JAVA code
- ~ What is a Combination Program
- ~ Combination Program JAVA code
- ~ Classic Tower of Hanoi problem
- ~ Classic Tower of Hanoi JAVA code

=> Linked List in depth :

- ~ Introduction to Linked List Preview
- ~ Add value in linked list - cases
- ~ Push Append and insert in LinkedList - JAVA code
- ~ Deletion of linked list THEORY.
- ~ Deletion in linked list JAVA code
- ~ Delete complete linked list JAVA code
- ~ Count all nodes in linkedlist JAVA code
- ~ Reversing a linked list THEORY
- ~ Reversing a linked list JAVA code

=> Circular Linked List in Depth :

- ~ Circular linked list THEORY Preview
- ~ Circular Linked List push JAVA code
- ~ Traverse a circular linked list JAVA code
- ~ Deletion in circular linked list JAVA code
- ~ count nodes in circular linked list JAVA code
- ~ convert linked list to circular linked list JAVA code

=> Doubly Linked List in Depth :

- ~ Theory for doubly linked list Preview
- ~ Doubly linked list push JAVA code
- ~ Insert After in doubly linked list JAVA code
- ~ add to last in doubly linked list JAVA code
- ~ Traverse a doubly linked list JAVA code
- ~ Deleting a node in doubly linked list JAVA code

=> Stack and Queue :

- ~ Stack - Push and Pop operation THEORY
- ~ Stack operations with JAVA code
- ~ Queue concept THEORY Preview
- ~ Queue implementation in JAVA code
- ~ Circular queue THEORY
- ~ Circular queue JAVA code

=> Binary Search Tree :

- ~ What is Binary Search tree and creation THEORY update
- ~ Insertion and Deletion in BST THEORY
- ~ InOrder Traversal of BST THEORY
- ~ Pre Order traversal in BST THEORY
- ~ Post order traversal in BST THEORY
- ~ Creating a Binary Search tree JAVA code
- ~ search a key in BST JAVA code
- ~ Insertion in BST JAVA code
- ~ deletion of key in BST JAVA code
- ~ inorder preorder and postorder traversal in BST JAVA code

=> Hashing :

- ~ What is Hashing THEORY
- ~ Hash chaining with linked list
- ~ Linear Hash Shifting
- ~ Square hash shifting

=> AVL Tree :

- ~ What is AVL tree and height
- ~ Finding balance factor
- ~ Left Left and Right Right Rotation in AVL Tree
- ~ LR and RL rotation with 1 trick
- ~ Creating a AVL tree - Important
- ~ Deletion in AVL Tree.

=> HEAP :

- ~ Heap - Max and min Heap
- ~ Insertion and deletion in HEAP

=> Sorting algorithms :

- ~ Categories of sorts
- ~ Selection sort - Theory
- ~ Selection sort - Java Code
- ~ Bubble Sort - Theory
- ~ Bubble Sort - Java Code
- ~ Insertion sort - Theory
- ~ Insertion sort - Java Code
- ~ Quick Sort - Theory
- ~ Quick Sort - Theory part 2
- ~ Counting Sort - Theory
- ~ Merge Sort Theory
- ~ Merge sort JAVA code
- ~ Counting Sort - Java Code

# Pro Live Classes

---

Sub-topic Name : Null

Course link : <https://ineuron.ai/course/Pro-Live-Classes>

## Course Description :-

Pro Live Classes are designed and taught by industry expert specializing in various domains and sharing their experience to our students.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises

## What you will learn :-

- => Developing and Open Sourcing an ML/DL package

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Day1 :
  - ~ *Developing and Open Sourcing an ML/DL package*
- => Day 2 :
  - ~ *Dynamic Programming*
- => Day 3 :
  - ~ *Path To Be A Data Engineer*

# Google Cloud Platform

---

Topic Name : CLOUD

Sub-topic Name : GCP

Course link : <https://ineuron.ai/course/Google-Cloud-Platform>

## Course Description :-

Google Cloud Platform is one of the most used cloud platforms across the globe that work on the same infrastructure that Google uses internally for its end-user products such as - Google Search, Gmail, Google Drive, and YouTube. In addition, we can use these services in different stacks like Machine Learning, Deep Learning, Big Data, etc. This course gives you a detailed understanding of GCP from essential to advanced, which helps you get various opportunities.

## Course Features :-

- => Challenges
- => Downloadable Resources
- => Quizzes
- => Assignments in each module
- => Completion Certificate

## What you will learn :-

- => Cloud Computing
- => Compute Service
- => App Engine
- => Cloud Storage
- => Google Cloud Billing
- => Cloud Pricing
- => IAM
- => GCP cloud service with respect to ML, DL, NLP with project implementation
- => Structured certification based sample questions

## Requirements :-

- => No prior knowledge of any language
- => A system with an internet connection
- => Dedication

## Instructors :-

=> Khushali Shah :

*~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.*

## Curriculum details :-

- => Introduction :
  - ~ Overview Preview
  - ~ Introduction to cloud computing
  - ~ Introduction to google cloud platform and GCP products
  - ~ Why GCP? creating GCP account Preview
  - ~ Google cloud overview
- => GCP Regions :
  - ~ GCP regions Preview
- => Platform Comparision :
  - ~ GCP with AWS
  - ~ GCP with Azure
  - ~ Compare AWS and Azure services to Google cloud
- => Google Pricing :
  - ~ Pricing overview
  - ~ Price list
  - ~ Pricing calculator
  - ~ Google cloud free program
- => IAM Role and Access Management :
  - ~ Overview
  - ~ Understanding roles
  - ~ Understanding IAM custom roles

- ~ Service accounts
- ~ Workload identity federation
- ~ Understanding policies
- ~ Overview of IAM conditions
- ~ Tags and access control
- ~ Hands-On IAM access
- ~ Hands-On IAM troubleshoot access
- ~ Hands-On IAM policy analyser
- ~ Hands-On IAM roles
- ~ IAM recommendation
- ~ Hands-On IAM condition

#### => Cloud Billing :

- ~ Cloud billing access control
- ~ A custom role for billing
- ~ Manage billing account
- ~ View Billing reports
- ~ Cost table
- ~ Price list
- ~ Cost breakdown
- ~ Resource-based CUD
- ~ Spend-based CUD
- ~ Export to BigQuery
- ~ Setting up billing data to BigQuery
- ~ BigQuery example overview
- ~ GCP data studio
- ~ Set budget
- ~ Automated cost example

#### => Google App Engine :

- ~ Overview
- ~ Choosing the right app engine environment
- ~ App engine standard environment
- ~ App engine price
- ~ Hello world application using app engine
- ~ Hands-On implementation part-1 Preview
- ~ Hands-On implementation part-2
- ~ Hands-On implementation part-3
- ~ Hands-On implementation part-4
- ~ Hands-On implementation part-5
- ~ Hands-On implementation part-6
- ~ Hands-On implementation part-7
- ~ Hands-On implementation part-8
- ~ Hands-On implementation part-9
- ~ App engine locations

#### => Compute Engine :

- ~ Overview
- ~ About virtual machine instances
- ~ VM instances lifecycle
- ~ VM instances pricing
- ~ Creating VM instance Preview
- ~ Creating VM instance-2
- ~ Connect to VM instance
- ~ Different machine families
- ~ Transfer files using VM/bucket
- ~ Transfer files using GCloud
- ~ Transfer files using WinSCP
- ~ Manage access in VM instances
- ~ Confidential VM's in compute engine
- ~ Using the compute engine API through client libraries
- ~ VM migration feature compute engine

#### => BookShelf App :

- ~ Overview
- ~ Create Firestore DB and enable API
- ~ Running application
- ~ Deploying app
- ~ Configuring bucket and giving public access-1
- ~ Configuring bucket and giving public access-2
- ~ Logging and error reporter
- ~ Code walkthrough-1
- ~ Code walkthrough-2
- ~ Clean up/ delete used resources

#### => Storage :

- ~ Overview
- ~ Cloud storage - Object storage/ Archival storage
- ~ Block storage - Persistent disks & local SSD
- ~ File storage
- ~ Storage transfer service
- ~ Firebase
- ~ Google workspace

#### => Certifications :

- ~ Overview of available certification
- ~ Cloud leader certification sample questions

# AWS Architect for the Real World

---

Topic Name : CLOUD

Sub-topic Name : AWS

Course link : <https://ineuron.ai/course/AWS-Architect-for-the-Real-World>

## Course Description :-

This Amazon AWS course will take you from AWS fundamentals to being a professional AWS cloud practitioner. From foundations to advanced topics, you will master general cloud computing principles and AWS. There are many hands-on activities that you may do with an Amazon Web Services (AWS) free tier account to gain expertise. This is the greatest approach to get started if you want to get into a high-paying profession working with cloud computing services. You'll go from beginning to advanced ideas, with lots of opportunity to put what you've learned into practise.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practicals in AWS
- => Course completion certificate

## What you will learn :-

- => AWS fundamentals
- => S3 buckets
- => Lambda functions
- => SES
- => API gateways
- => Load balancing
- => Migration
- => Container services
- => AWS serverless

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

=> Course Intro :

~ AWS Architect for real world

=> Getting started with AWS and IAM :

- ~ FAQ for aws architect course
- ~ Getting started with AWS and expectation
- ~ Tour of AWS console with ROOT user
- ~ AWS Infra - Region and AZ
- ~ Securing root account and MFA
- ~ Custom signin link for IAM
- ~ Why groups are created
- ~ Creating groups and users
- ~ What are roles in IAM
- ~ Temporary security credentials in IAM
- ~ Billing alarms in Cloudwatch
- ~ Password compliance
- ~ buying domain on Route 53

=> Amazon Elastic Compute Cloud -EC2 :

- ~ What is Elastic Compute
- ~ Instance types and limits
- ~ your first EC2 instance
- ~ In depth guide for EC2 options

- ~ Connecting to cloud instance
- ~ Configure an AWS web server
- ~ Stress testing, Cloud watch alarms and clean up
- ~ What are user data scripts
- ~ What is instance meta-data
- ~ Docs and hands on with Elastic IP
- ~ Custom network interface cards in AWS
- ~ creating custom AMI
- ~ Launch with custom image and clean up
- ~ Placement groups - Cluster, partition and Spread
- ~ EC2 pricing - OnDemand, spot and reserved
- ~ Just for Exam

#### => Virtual Private Cloud - VPC :

- ~ Why you should focus more on VPC
- ~ Understand the default VPC
- ~ Create diagram of default VPC
- ~ CIDR deep dive
- ~ Your custom VPC
- ~ Creating subnet in custom VPC
- ~ Internet Gateway and route table
- ~ Lets complete the diagram
- ~ VPC DNS hostname and resolution
- ~ updates from corporate in VPC structure
- ~ Clean up the resources
- ~ Security groups VS NACL
- ~ Understand the next diagram for VPC
- ~ Diagram 2 - VPC and subnets
- ~ Diagram 2 - Route tables and IGW
- ~ Configure a NAT gateway
- ~ SSH agent forwarding
- ~ Bastion host and testing of diagram
- ~ Bastion host and testing of diagram part 2
- ~ NAT instance and configurations
- ~ VPC peering connection
- ~ What are transit gateways
- ~ A use case of Endpoints in VPC
- ~ preparing logs for audit - flowLogs
- ~ Resources for hybrid cloud - VPN and more
- ~ Lets audit the logs with Athena and Glue
- ~ Egress gateway cloudhub and clean up

#### => Load Balancing and scalability :

- ~ What are load balancers
- ~ Type of Load Balancer
- ~ Prep work for load balancers
- ~ Configure target groups
- ~ Creating an Application load balancer
- ~ Path and HOST based routing on domain
- ~ Cross Zone load balancer
- ~ Case of Sticky session
- ~ Clean up for ALB
- ~ Network Load Balancer
- ~ Scaling - Horizontal and Vertical
- ~ Auto Scaling Group configuration
- ~ Clean up for ASG resources

#### => Route 53 in Depth :

- ~ Welcome to Route 53
- ~ What are hosted zone - Public and Private
- ~ AWS DNS records - A and Alias
- ~ Creating instance in multiple region
- ~ Route 53 Health Checks
- ~ Simple and weighted route policy
- ~ FailOver and latency based policies
- ~ Multi value and restricting content on geo location
- ~ Clean up for Route 53

#### => Storage in AWS - S3 :

- ~ lets start with AWS storage
- ~ Introduction to S3 buckets
- ~ Permissions in S3 buckets
- ~ Static website hosting in S3 buckets
- ~ S3 bucket - Versioning and encryption
- ~ S3 event notifications
- ~ Access log BILLS and requester pays
- ~ S3 storage class
- ~ Data replication - CRR and SRR
- ~ S3 Select, Athena and Redshift - Query
- ~ Data life cycle policy
- ~ Getting started with cloudfront and OAI
- ~ Setup a cloudfront and OAI for a website

#### => Storage - Block and Object :

- ~ Instance Store - ephemeral
- ~ Types of EBS volume and IOPS
- ~ Creating and mounting EBS volume
- ~ Getting a snapshot of EBS
- ~ Re attach EBS volume
- ~ Data migration between AZ and Region

- ~ RAID 0 and 1 config
- ~ Creating and mounting Elastic File Storage
- ~ FSx for Windows and Lustre
- ~ Storage Gateway - Hybrid cloud
- ~ Storage Gateway NOT by LCO

#### => Databases in AWS :

- ~ Introduction to Databases in AWS
- ~ OLTP vs OLAP
- ~ Production level RDS walkthrough
- ~ Create a mysql db in AWS
- ~ Multi AZ replica RDS
- ~ Creating read replicas
- ~ Read Replica VS Multi AZ deployment
- ~ AWS aurora Docs walkthrough
- ~ Getting started with DynamoDB
- ~ Creating a table in DynamoDB
- ~ Reading the DAX Docs
- ~ ElasticCache memcached
- ~ ElasticCache Redis and Redis cluster
- ~ Redshift Overview

#### => Application integration in AWS :

- ~ Application integration services by AWS
- ~ Simple queue service
- ~ Creating our first queue service
- ~ FIFO vs standard queue
- ~ Delay, visibility and retention time
- ~ Dead letter queue
- ~ Long polling and short polling
- ~ Attaching lambda to SQS
- ~ Clean up all the sqs resources
- ~ Step function and simple workflow service
- ~ Amazon MQ, Rabbit MQ and other services

#### => PAAS and IAAS in AWS :

- ~ Getting started with PAAS and IAAS
- ~ Cloudformation inDepth guide
- ~ Beanstack application deployment

#### => Process and Migrate the Data :

- ~ Kinesis and shards
- ~ Kinesis analytics and firehose
- ~ What is Elastic MapReduce
- ~ What is Athena, Glue and Glue Studio
- ~ Import from other Virtualization Services
- ~ Database Migration service and Schema Conversion Tool

#### => Security Compliance :

- ~ Security and Compliance - SOX, PCI and more
- ~ Key Management Service
- ~ Hardware Security Module in Cloud
- ~ AWS WAF and shield service
- ~ Active Directory in AWS
- ~ What is AWS Cognito
- ~ AWS single sign on
- ~ AWS Directory service

#### => Container Service :

- ~ What are container service in AWS
- ~ What is Docker
- ~ What is Elastic Container Registry
- ~ What are microservices
- ~ What is Elastic Container service
- ~ What is Fargate
- ~ What is Elastic Kubernetes Service
- ~ AWS walkthrough for ECS and EKS

#### => AWS Serverless :

- ~ Getting started with AWS serverless
- ~ A common warning for AWS
- ~ Route 53
- ~ Get Started with S3 bucket
- ~ Struggle of web page hosting
- ~ Hosting with policies
- ~ GET vs POST and handling response
- ~ Your first lambda in AWS
- ~ Lambda permission and cloud watch
- ~ Introducing API gateway
- ~ Lambda for POST information
- ~ Post Data and CORS error
- ~ First look at SES
- ~ New user for SES and lambda
- ~ Sending email from SES and lambda

# Vedic Math

---

Topic Name : K12

Sub-topic Name : CLASS8

Course link : <https://ineuron.ai/course/Vedic-Math>

## Course Description :-

This course will help you solve complex mathematical problems using Vedic Mathematics. This course is curated on a set of concepts that will help you improve your calculations to an extent where before you pick up a pen, you would find the answers by simplifying calculations into simple steps. If you despise numbers, This course will help you interactively appreciate the beauty of mathematics.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction about Vedic math's
- => Benefits of Vedic math's
- => Addition of numbers
- => Subtraction of numbers
- => Multiplication of numbers
- => Division of numbers
- => Square of a number
- => Cube of a number
- => Square root of a number
- => Cube root of a number

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to VEDIC MATHS :
  - ~ What is VEDIC MATHS?
  - ~ Benefits of Vedic Maths
  - ~ Why we learn VEDIC MATHS ?
- => Basics VEDIC MATHS :
  - ~ Find the complement of 1 digit number/ 2 digit number/ 3 digit number/ any digit number?
  - ~ Tables for 9/19/29/39/129/149/.
  - ~ All from 9 and the Last From 10
  - ~ Multiplication of 2 numbers where number of digits are same in both number and sum of unit digit number is 10
  - ~ Multiplication of 2 numbers where digits are same in both numbers except unit digit number
  - ~ Multiplication with 11
  - ~ Multiplication with 12
  - ~ Universal multiplication like multiplication for following exs:- 1)  $2^2 \times 2$  2)  $3^3 \times 3$  3)  $4^4 \times 4$  4)  $2^3 \times 5$  5)  $3^4 \times 6$  6)  $2^4 \times 7$  7)  $3^2 \times 2$  etc (called as URDHVA TIRYA GAMYAM)
  - ~ Square of different types of number ( For example, whose unit digit /the last digit is 1/5/6/4
  - ~ Cubes of 2 digit number
- => Intermediate :
  - ~ Division of any number by 5, 8 & 98
  - ~ Division of any number by 11
  - ~ Division of any number by 12-19
  - ~ Division of any number by 25, 50 and 100



- ~ Division by factors
- ~ Percentages
- ~ Addition of odd, even series of numbers
- ~ Multiplication of numbers ending with 5
- ~ Multiplication of numbers with 15
- ~ Multiplication of whole number with mixed fractions

=> Advanced VEDIC MATHS :

- ~ Division of whole number with mixed fractions
- ~ Addition of special fractions
- ~ Square of any number nearer to base
- ~ Multiplications with 9/99/999 when
- 1) Multipliers are same digit
- 2) Less digits than multiplier
- 3) More digits than multiplier
- ~ Different types of base multiplication for ex.  $97 * 94$  (less than base 100),  $14 * 12$
- ~ Multiplication of numbers like  $(42 * 46)$  considering primary & secondary base
- ~ different (in this ex, 10 & 50 are two different base)
- ~ Cubes of numbers closer to bases
- ~ Division (Nikhilam method where divider is less than base number or nearer to base number)
- ~ Quickest division by 9,99,999,9999
- ~ Repeated digit base number squares
- ~ Vinculum of number at Unit and tens places
- ~ Squares by duplex method
- ~ Square root
- ~ Cube root

=> Application: :

~ We will make a UI where we will provide a set of questions where kids can answer those questions & check if it is correct or not/ Similarly, In the backend, they can do logic according to Vedic formula.  
Plus, we will provide a hint that explains the Vedic formula/procedure to find the solution in the front end.

# Data Warehouse

---

Topic Name : BIG DATA

Sub-topic Name : TECH STACK

Course link : <https://ineuron.ai/course/Data-Warehouse>

## Course Description :-

A data warehouse is a type of data management system that is designed to enable and support business intelligence (BI) activities, especially analytics. Data warehouses are solely intended to perform queries and analysis and often contain large amounts of historical data.

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Master the techniques needed to build a data warehouse for your organization
- => Determine your options for the architecture of your data warehousing environment.
- => Apply the key design principles of dimensional data modelling
- => Combine various models and approaches to unify and load data within your data warehouse.

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => What is data warehouse :
  - ~ Course overview Preview
  - ~ What is data warehousing Preview
  - ~ History of datawarehouse
  - ~ How datawarehouse works
  - ~ Types of data warehouse
  - ~ General stages of data warehouse
  - ~ Components of data warehouse
  - ~ What is data warehouse used for
  - ~ data warehouse implementation
  - ~ data warehouse tools
- => database vs data warehouse :
  - ~ Key differences
  - ~ characteristics
  - ~ Difference between database and warehouse
  - ~ Application of database
  - ~ Application of data warehousing
- => Data Warehouse Architecture :
  - ~ Characteristics of Data Warehouse
  - ~ Data warehouse architecture
  - ~ Data Warehouse Components
  - ~ Main types of datawarehouse
- => What is ETL :
  - ~ what is ETL Preview
  - ~ ETL Process in Data warehouses
- => ETL vs ELT :
  - ~ Difference between ETL and ELT

# Job Guaranteed BigData Bootcamp 2.0

---

Topic Name : BIG DATA

Sub-topic Name : BIG DATA MASTERS

Course link : <https://ineuron.ai/course/Job-Guaranteed-BigData-Bootcamp-2.0>

## Course Description :-

By using the tools and processes that big data scientists and engineers use on a daily basis, you will be intelligent enough to understand the insights that big data may offer. With a general understanding of how large data is arranged, examined, and evaluated, you can make better business judgments. This unique industry program will help to learn the entire stack of Big Data and be ready to crack jobs in leading organizations.

## Course Features :-

- => High Quality Premium Big Data Labs Included
- => Full stack Big Data certification
- => Job guarantee otherwise refund
- => One year of internship Anytime
- => 1:1 Personalized Mentorship
- => Revision Classes
- => Online Instructor-led learning: Live teaching by instructors
- => 20 + hands-on industry real-time projects.
- => 200 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Doubt clearing through mail and chat support team
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building Anytime
- => Career guidance Anytime
- => Interview Preparation Anytime
- => Regular assessment
- => Job Fair and Internal Hiring
- => Mock Interview Anytime

## What you will learn :-

- => No Local installation - Premium Big Data Labs Included
- => Big Data
- => Hadoop
- => HDFS
- => YARN
- => Map-Reduce
- => Linux Commands
- => DBMS
- => MySQL
- => Complete SQL
- => Apache Hive
- => Apache Sqoop
- => Data Warehousing
- => NoSQL Databases
- => Apache Hbase
- => Cassandra

=> Neo4J  
=> MongoDB  
=> Confluent Kafka  
=> Python Programming  
=> Apache Spark  
=> Airflow  
=> BigData On Cloud  
=> AWS  
=> GCP  
=> Azure  
=> Tableau  
=> PowerBI  
=> Batch Data Pipelines  
=> Real Time Data Pipelines

### Requirements :-

=> Premium Big Data Labs Included  
=> System with Internet Connection  
=> Interest to learn  
=> Dedication

### Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sunny Savita :

~ I'm an AI enthusiast, graduate in Computer science and engineering. Currently working with iNeuron.ai as a Data Scientist and having 2+ years of experience. I have skills in big data, machine learning, computer vision, Natural language processing. My expertise also includes project design development and implementation with AI/ops tools.

=> Shashank Mishra :

~ Experienced Data Engineer with a demonstrated history of working in service and product companies. Solved data mysteries for different domains like Aviation, Pharmaceutical, FinTech, Telecom and Employee Services. Have designed scalable & optimized data pipelines to handle PetaBytes of data, with Batch & Real Time frequency. Got good exposure on different BigData frameworks (Hadoop, Spark, Hive, Sqoop, Flume, Flink, Kafka, Docker), Databases (MySQL, HBase, Cassandra, Redshift, Elastic Search), AWS Services (S3, Lambda, EMR, Glue, Cloudwatch, Redshift, SNS, SQS, Athena, Appflow), Dashboarding Tools (Grafana, Kibana, QuickSight, DataDog, Data Studio), Monitoring Tools (Airflow, Azkaban), Web Development (HTML, CSS, Scala Play, Django, Rest API, JavaScript, Ajax, JQuery), Good command over programming languages (Python, Java, Scala, Shell Scripting) and strong Data Structures & Algorithm fundamentals.

### Curriculum details :-

=> Introduction to the Course :

~ Welcome to the course  
~ Course Introduction  
~ Introduction to data engineer profile  
~ Introduction to data engineer Technologies  
~ Data engineer role and responsibility  
~ Introduction to data engineer technologies

=> BigData Introduction :

~ What is Big Data?  
~ Evolution of Big Data  
~ Why learn Big Data technologies?  
~ Examples of Big Data  
~ Who is using Big Data?  
~ Why is Data so important?  
~ Characteristics of Big Data  
~ Challenges of Big Data  
~ Data scale  
~ Manage, store and process Big Data  
~ 5 Vs of Big Data  
~ Sources of Data flood  
~ Exploding data problem  
~ OLTP and OLAP  
~ Operational vs Analytical Big Data  
~ Possible solutions: scaling up vs. scaling out  
~ Challenges of scaling up and scaling out

=> Hadoop Fundamentals :

~ What is Hadoop?  
~ Hadoop in layman's term  
~ History and timeline of Hadoop  
~ Evolutionary features of Hadoop  
~ Why is Hadoop in demand?  
~ Components of Hadoop ecosystem  
~ Hadoop architecture

- ~ How does Hadoop solve the data explosion problem?
- ~ Differences between Hadoop 1.X and Hadoop 2.X and Hadoop 3.X
- ~ Hadoop 1.x 2.x 3.x architecture, components and working of those Components

=> HDFS :

- ~ Design of HDFS
- ~ HDFS architecture
- ~ HDFS features
- ~ Difference between HDFS NTFS FAT EXT file system
- ~ Name node and data node
- ~ Secondary name node
- ~ Job tracker
- ~ Task tracker
- ~ Client nodes
- ~ Explain master-slaves
- ~ Pseudo-distributed
- ~ Fully-distributed
- ~ Data replication
- ~ How does a file read and write work?
- ~ Local file system and HDFS
- ~ Rack awareness
- ~ Arrangement of racks
- ~ Arrangement of machines and racks
- ~ Checkpointing in Hadoop
- ~ Benefits of replica placement and rack awareness
- ~ URL And URN
- ~ HDFS commands
- ~ HDFS web interface
- ~ Fault tolerance
- ~ Name node failure management
- ~ Anatomy of file read and write from HDFS
- ~ Important java classes to write data to HDFS
- ~ Inputsplit and data blocks difference
- ~ Why Is the block size 128 MB?
- ~ Recordreader
- ~ Inputformat
- ~ Default Inputformat: TextInputformat
- ~ Outputformat
- ~ What is a partitioner?
- ~ Using partitioner
- ~ Map only job
- ~ Flow of operations in MapReduce
- ~ Serialisation in MapReduce

=> Linux Fundamentals & Commands :

- ~ What is linux?
- ~ Linux file system
- ~ How to use linux commands from terminal?
- ~ Remote server access, SSH Login
- ~ Working with remote directories
- ~ Working with files
- ~ ls - The most frequently used command in Linux to list directories
- ~ pwd - Print working directory command in Linux
- ~ cd - Linux command to navigate through directories
- ~ mkdir - Command used to create directories in Linux
- ~ mv - Move or rename files in Linux
- ~ cp - Similar usage as mv but for copying files in Linux
- ~ rm - Delete files or directories
- ~ touch - Create blank/empty files
- ~ cat - Display file contents on the terminal
- ~ clear - Clear the terminal display
- ~ echo - Print any text that follows the command
- ~ man - Access manual pages for all Linux commands
- ~ uname - Linux command to get basic information about the OS
- ~ whoami - Get the active username
- ~ tar - Command to extract and compress files in Linux
- ~ grep - Search for a string within an output
- ~ head - Return the specified number of lines from the top
- ~ tail - Return the specified number of lines from the bottom
- ~ diff - Find the difference between two files
- ~ cmp - Allows you to check if two files are identical
- ~ sort - Linux command to sort the content of a file while outputting
- ~ export - Export environment variables in Linux
- ~ zip - Zip files in Linux
- ~ unzip - Unzip files in Linux
- ~ ssh - Secure Shell command in Linux
- ~ service - Linux command to start and stop services
- ~ ps - Display active processes
- ~ kill and killall - Kill active processes by process ID or name
- ~ df - Display disk filesystem information
- ~ chmod - Command to change file permissions
- ~ chown - Command for granting ownership of files or folders
- ~ wget - Direct download files from the internet
- ~ apt, pacman, yum, rpm - Package managers depending on the distro
- ~ sudo - Command to escalate privileges in Linux
- ~ whereis - Locate the binary, source, and manual pages for a command
- ~ whatis - Find what a command is used for

=> HDFS Operations :

- ~ Start HDFS
- ~ Listing files in HDFS
- ~ Writing a file into HDFS
- ~ Reading data from HDFS
- ~ Shutting down HDFS
- ~ Listing contents of directory
- ~ Displaying and printing disk usage
- ~ Moving files & directories
- ~ Copying files and directories
- ~ Displaying file contents

=> Map-Reduce :

- ~ Distributed Computation
- ~ Map-Reduce Architecture
- ~ Working of Map
- ~ Working Of Reduce
- ~ Shuffling, Sorting, Combining and Partitioning in Map-Reduce Execution
- ~ Word Count problem using Map-Reduce in Python

=> Yarn :

- ~ What is Yarn?
- ~ Why Yarn?
- ~ Classic MapReduce v/s Yarn
- ~ Yarn architecture
- ~ Resource Manager
- ~ Node manager
- ~ Application master
- ~ Node manager containers
- ~ Resource manager components
- ~ Advantages & disadvantages of Yarn
- ~ Yarn applications
- ~ Scheduling in Yarn
- ~ Fair Scheduler
- ~ Fault Tolerance
- ~ Schedulers in Yarn
- ~ FIFO scheduler
- ~ Capacity scheduler
- ~ Fair scheduler

=> Introduction to transactional Databases or RDBMS :

- ~ What is Database?
- ~ Why to use Databases?
- ~ What is RDBMS?
- ~ Operations in Databases
- ~ ER Diagrams
- ~ Concepts of Keys - Primary Key, Foreign Keys, Composite Keys, Candidate Keys
- ~ Joining Datasets in Databases - Inner, Left, Right, full outer
- ~ ACID Properties
- ~ Transactions and Transaction control
- ~ Indexing

=> Introduction to SQL :

- ~ Why SQL?
- ~ Application of SQL
- ~ Characteristics of SQL
- ~ MySQL Installation guide
- ~ Connection & set up
- ~ DDL, DML, DCL in SQL
- ~ Data type of SQL
- ~ Binary data types
- ~ Approximate numeric data type
- ~ Exact numeric data type
- ~ Character string data type
- ~ Date and time datatype

=> SQL Operations :

- ~ CREATE DATABASE
- ~ DROP DATABASE
- ~ CREATE table
- ~ CREATE table with PRIMARY KEY
- ~ CREATE table with FOREIGN KEY
- ~ DELETE table
- ~ TRUNCATE table
- ~ TEMP table
- ~ RENAME table
- ~ DROP table
- ~ COPY table
- ~ ALTER table
- ~ INSERT query
- ~ UPDATE query
- ~ DELETE query
- ~ SELECT statement
- ~ SELECT DISTINCT
- ~ SELECT COUNT
- ~ SELECT TOP
- ~ SELECT LAST
- ~ SQL DESC statement
- ~ SQL USE statement
- ~ SQL COMMIT statement
- ~ SQL ROLLBACK statement

- ~ *CREATE View*
- ~ *SELECT View*
- ~ *DROP View*

#### => SQL Operators :

- ~ *Addition*
- ~ *- (Subtraction)*
- ~ *\* (Multiplication)*
- ~ */ (Division)*
- ~ *% (Modulus)*
- ~ *EXISTS*
- ~ *IN, NOT IN*
- ~ *ANY, ALL*
- ~ *NULL, NOT NULL*
- ~ *LIKE*
- ~ *BETWEEN*
- ~ *Equal to (=)*
- ~ *Greater than(>)*
- ~ *Less than(<)*
- ~ *Not equal to(!=)*

#### => SQL Clauses :

- ~ *SQL WHERE clause*
- ~ *SQL ORDER BY clause*
- ~ *SQL GROUP BY clause*
- ~ *SQL HAVING clause*
- ~ *SQL CASE-WHEN clause*

#### => Joins in SQL :

- ~ *INNER JOIN*
- ~ *LEFT JOIN*
- ~ *RIGHT JOIN*
- ~ *FULL JOIN*
- ~ *SELF JOIN*
- ~ *CARTESIAN JOIN*

#### => Nested Subqueries in SQL :

- ~ *Subquery in FROM clause*
- ~ *Subquery in SELECT clause*
- ~ *Subquery in WHERE clause*
- ~ *Correlated subqueries*
- ~ *Filter query results using query on different table*

#### => Aggregation & Window Functions in SQL :

- ~ *Conditional aggregation*
- ~ *SUM(), MIN(), MAX(),AVG(),COUNT()*
- ~ *Window Function Syntax*
- ~ *OVER Clause*
- ~ *Partition By*
- ~ *Order By*
- ~ *Row\_Number()*
- ~ *Rank()*
- ~ *Dense\_Rank()*
- ~ *Lead()*
- ~ *Lag()*
- ~ *First\_Value()*
- ~ *Last\_Value()*
- ~ *NTILE()*
- ~ *RANGE BETWEEN*
- ~ *ROWS BETWEEN*

#### => CTE (Common Table Expression) in SQL :

- ~ *NON Recursive CTE*
- ~ *Recursive CTE*

#### => Hive Introduction :

- ~ *What is Hive?*
- ~ *Hive Vs Map Reduce*
- ~ *Hive Vs Relational databases*

#### => Hive Architecture :

- ~ *Hive architecture*
- ~ *Different modes of Hive*
- ~ *Data Types in Hive*
- ~ *Operators in Hive*

#### => HQL (Hive Query Language) :

- ~ *Hive DDL*
- ~ *How to create and drop databases?*
- ~ *Hive create table: internal table, external table , alter, drop*
- ~ *Create*
- ~ *Show*
- ~ *Describe*
- ~ *Use*
- ~ *Drop*
- ~ *Alter*
- ~ *Truncate*
- ~ *Hive DML*
- ~ *Load*
- ~ *Select*
- ~ *Insert*

- ~ Delete
- ~ Update
- ~ Export
- ~ Import
- ~ HiveQL- Where
- ~ HiveQL- Order By
- ~ HiveQL- Group By
- ~ HiveQL- Joins and types
- ~ HiveQL- SubQuery
- ~ Analytic and windowing in Hive
- ~ Hive view and index
- ~ What is Hive metastore?
- ~ How to install and configure Hive metastore?
- ~ What is Hive data modelling?
- ~ Hive ETL: loading JSON, XML, text data
- ~ Working with arrays, maps and struct data types
- ~ Sort by and order by

=> Working with different file formats in Hive :

- ~ File formats in Hive
- ~ Text files
- ~ Input formats in Hive
- ~ Sequence files in Hive
- ~ RC file in Hive
- ~ ORC files in Hive
- ~ Avro files
- ~ Parquet file
- ~ SerDe in Hive
- ~ Accessing Hive with JDBC
- ~ Working with UDF
- ~ Optimizations in Hive

=> Hive Partitioning and Bucketing :

- ~ Partitioning in Hive with Queries
- ~ Static and dynamic partitioning
- ~ Bucketing in Hive with Queries
- ~ Bucketing vs Partitioning
- ~ Distribute by and cluster by

=> Joins in Hive and Optimizations :

- ~ Map-Side Join
- ~ Bucket-map join
- ~ Sort-Merge-Bucket-Map join
- ~ Left semi join

=> Introduction to Sqoop :

- ~ Sqoop introduction
- ~ How does Sqoop work?
- ~ Why do we use Sqoop?
- ~ Features of Sqoop

=> Sqoop Tools :

- ~ Sqoop architecture and working
- ~ Using command aliases
- ~ Controlling the Hadoop installation
- ~ Using generic and specific arguments
- ~ Using options files to pass arguments

=> Sqoop import :

- ~ Purpose of Sqoop import
- ~ Connecting to a database server
- ~ Selecting the data to import
- ~ Free-form query imports
- ~ Controlling the import process
- ~ Controlling transaction isolation
- ~ Controlling type mapping
- ~ Incremental imports
- ~ File formats
- ~ Large objects
- ~ Importing data into Hive
- ~ Importing data into HBase
- ~ Importing data into Accumulo
- ~ Connecting to a Mainframe

=> Sqoop export :

- ~ Purpose of Sqoop export
- ~ Inserts vs Updates
- ~ Exports and Transactions

=> Sqoop - Job :

- ~ Create Job
- ~ Verify Job
- ~ Inspect Job
- ~ Execute Job

=> Validation in Sqoop :

- ~ Introduction to the validation
- ~ Purpose of Validation
- ~ Limitations of Validations

=> Data Warehousing :

- ~ OLAP vs OLTP



- ~ What is a Data Warehouse?
- ~ Difference between Data Warehouse, Data Lake and Data Mart
- ~ Fact Tables
- ~ Dimension Tables
- ~ Slowly changing Dimensions
- ~ Types of SCDs
- ~ Star Schema Design
- ~ Snowflake Schema Design
- ~ Data Warehousing Case Studies

=> Introduction To NoSQL databases :

- ~ Challenges with traditional RDBMS
- ~ What is a Nosql database?
- ~ History behind the creation of Nosql databases
- ~ Features of Nosql database
- ~ Different types of Nosql databases
- ~ When should Nosql be used?
- ~ Advantages of Nosql
- ~ Disadvantages of Nosql
- ~ CAP theorem

=> Introduction of HBase :

- ~ What is HBase?
- ~ HDFS and HBase
- ~ HBase vs RDBMS
- ~ HBase vs HIVE
- ~ HBase storage mechanism
- ~ Feature of HBase
- ~ Applications of HBase

=> HBase architecture :

- ~ Architecture of HBase
- ~ Components of HBase architecture
- ~ Client library
- ~ Zookeeper
- ~ HMaster server
- ~ HBase regions servers

=> CRUD operations in HBase :

- ~ What is HBase shell?
- ~ HBase shell usage
- ~ Starting HBase shell
- ~ Creating table
- ~ Inserting a row
- ~ Updating a row
- ~ Retrieving a row
- ~ Retrieving a range of rows
- ~ Deleting a row
- ~ Deleting a table
- ~ Retrieve rows within a time range
- ~ Filter by column value - SingleColumnValueFilter
- ~ Filter by Row id - RowFilter
- ~ Apply multiple conditions - Filterlis

=> HBase commands :

- ~ General commands
- ~ status
- ~ table\_help
- ~ version
- ~ whoami
- ~ Data definition commands
- ~ alter
- ~ alter\_async
- ~ alter\_status
- ~ create
- ~ drop
- ~ drop\_all
- ~ enable
- ~ enable\_all
- ~ exists
- ~ get\_table
- ~ is\_disabled
- ~ is\_enabled
- ~ show\_filters
- ~ Data manipulation commands
- ~ append
- ~ count
- ~ delete
- ~ deleteall
- ~ get\_table
- ~ get\_counter
- ~ put
- ~ truncate
- ~ truncate\_preserve
- ~ Other HBase shell commands
- ~ Admin commands
- ~ Replication commands
- ~ Snapshot commands
- ~ Visibility labels commands
- ~ Security commands

=> Introduction and overview of cassandra :

- ~ *What is Apache Cassandra?*
- ~ *History of Cassandra*
- ~ *Cassandra Database vs Relational Database*
- ~ *Apache Cassandra features*
- ~ *Cassandra use cases and applications*
- ~ *Advantages of Cassandra*
- ~ *Disadvantages of Cassandra*

=> Cassandra Architecture :

- ~ *Cassandra architecture*
- ~ *Cassandra data model*
- ~ *Cassandra as a distributed database*
- ~ *Node*
- ~ *Data centre*
- ~ *Cluster*
- ~ *Commit log*
- ~ *Mem-table*
- ~ *SSTable*
- ~ *Data replication*
- ~ *Write operation*
- ~ *Read operation*
- ~ *Data compaction*

=> Cassandra Data Modelling :

- ~ *Data modelling basics*
- ~ *Cassandra data modelling*
- ~ *Cassandra column types*
- ~ *Cassandra keyspace*

=> Setup, installation and configuration :

- ~ *Cassandra configuration with datastax*
- ~ *Understanding different ways to communicate with cassandra*
- ~ *Using cqlsh*

=> Cassandra cluster and node :

- ~ *Configure and managing a cluster*
- ~ *Cluster and nodes*
- ~ *Adding nodes to cluster*
- ~ *Monitoring a cluster*
- ~ *Repairing a nodes*
- ~ *Removing a node*

=> Cassandra - Shell Commands :

- ~ *Help*
- ~ *Capture*
- ~ *Consistency*
- ~ *Copy*
- ~ *Describe table*
- ~ *Describe keyspaces*
- ~ *Expand*
- ~ *Exit*
- ~ *Show*
- ~ *Source*

=> Cassandra Query Language(CQL) :

- ~ *CQL Data Definition Commands*
- ~ *Cassandra CQL Data Types*
- ~ *Creating Database*
- ~ *Creating Keyspace*
- ~ *Use Keyspace*
- ~ *Alter Keyspace*
- ~ *Drop Keyspace*
- ~ *Create Table*
- ~ *Alter table*
- ~ *Drop table*
- ~ *Truncate*
- ~ *Create Index*
- ~ *Drop Index*
- ~ *CQL Data Manipulation Commands*
- ~ *Insert*
- ~ *Update*
- ~ *Delete*
- ~ *Batch*
- ~ *CQL Clauses*
- ~ *Select*
- ~ *Cassandra Where Clause*
- ~ *Cassandra Order by Clause*

=> Cassandra CRUD Operation :

- ~ *Create data*
- ~ *Update data*
- ~ *Read data*
- ~ *Delete data*
- ~ *Maps*
- ~ *Sets*
- ~ *Lists*
- ~ *Key and indexing*

=> Advanced CQL :

- ~ CQL Collections
- ~ CQL User-Defined Types
- ~ Defining a Primary key
- ~ Defining a Partition key
- ~ Introduction to User-defined types(UDT)
- ~ How to Create a UDT?
- ~ UDT literals
- ~ How to alter a UDT?
- ~ How to drop a UDT?

#### => Introduction to MongoDB :

- ~ Introduction
- ~ key characteristic of MongoDB
- ~ Understanding MongoDB ecosystem
- ~ Advantages & disadvantages of using MongoDB

#### => MongoDB Atlas setup :

- ~ What is Apache Atlas
- ~ Features of Apache Atlas
- ~ MongoDB atlas setup
- ~ Basic search in Atlas UI
- ~ Advanced search in Atlas UI

#### => Architecture of MongoDB :

- ~ Architecture of MongoDB
- ~ Understanding databases, collections & documents
- ~ Creating databases & collections
- ~ Understanding JSON Data
- ~ Comparing JSON & BSON
- ~ Storage engines
- ~ Read path
- ~ Write path
- ~ Working set
- ~ Capped collection
- ~ Oplog collection
- ~ TTL index
- ~ Gridfs

#### => CRUD operations MongoDB :

- ~ MongoDB data types
- ~ Finding, Inserting, Deleting & Updating elements
- ~ Querying the documents
- ~ Bulk insert operations
- ~ Updating multiple document
- ~ Limiting documents
- ~ Understanding insertOne vs insertMany()
- ~ Updateone() vs updateMany()
- ~ Understanding find() & fetchall()
- ~ Understanding "deleteOne()" & "deleteMany()"
- ~ Filtering documents

#### => Schema design and data modelling :

- ~ Why do we use Schemas?
- ~ What is data modelling?
- ~ RDBMS and MongoDB data modelling difference
- ~ Embedding document
- ~ Reference document
- ~ Structuring documents
- ~ Understanding relations
- ~ One To One
- ~ One To Many
- ~ Many To Many

#### => Database administration in MongoDB :

- ~ Database status
- ~ Troubleshooting issues
- ~ Current operations
- ~ Rotating log files
- ~ Users and roles
- ~ Copy and clone database
- ~ DB and collection stats
- ~ Explain plan
- ~ Profiling
- ~ Changing configuration files
- ~ Upgrading the database

#### => MongoDB: backup and security :

- ~ Concept of backups
- ~ Mongoexport/mongoimport
- ~ Mongodump/mongorestore
- ~ Oplog backups
- ~ LVM backups
- ~ Backups using MMS/Ops manager
- ~ Purpose of security
- ~ Authentication and authorization
- ~ Role based access control

#### => Working with python driver :

- ~ Splitting work between the Driver & the Shell
- ~ Preparing our project
- ~ Installing Visual Studio Code or Pycharm

- ~ Installing the Python
- ~ Connecting Python & the MongoDB cluster
- ~ Storing products in the database
- ~ Fetching data from the database
- ~ Getting a single product
- ~ Editing & deleting products
- ~ Implementing pagination
- ~ Adding an index
- ~ Adding an index to make the Email unique
- ~ Adding user sign-in

#### => Replication in MongoDB :

- ~ Concept of replication
- ~ Replica Set member roles
- ~ Voting and electing primary
- ~ Role of oplog in replication
- ~ Read and write concern
- ~ Arbiter, Hidden and Delayed replica node
- ~ Priority settings
- ~ Replica Set nodes health check
- ~ Concept of resyncing the nodes
- ~ Rollbacks during failover
- ~ Keyfile authentication

#### => MongoDB scalability :

- ~ Concept of scalability
- ~ Sharding concept
- ~ Sharding and chunks
- ~ Choosing shard key
- ~ Sharding components
- ~ Types of sharding
- ~ Balanced data distribution
- ~ Sharded and non-sharded collection
- ~ Sharded replica set
- ~ Tag aware sharding

#### => MongoDB Monitoring :

- ~ MMS manager
- ~ Ops manager
- ~ MongoDB utility commands
- ~ MongoDB developer tools
- ~ MongoDB client drivers

#### => Introduction to Graph Database :

- ~ What is a Graph Database?
- ~ Why Graph Database?
- ~ Graph DB Data Model
- ~ Graph DB vs RDBMS
- ~ The Property Graph Model
- ~ What is Neo4j
- ~ Advantages of Neo4j
- ~ Features of Neo4j
- ~ Neo4j Property Graph Data Model

#### => Setup to Neo4j :

- ~ Environment setup for Neo4j
- ~ Installation of Neo4j on Windows
- ~ Installation of Neo4j on Linux
- ~ Installation of Neo4j on Mac
- ~ Exploring Neo4j Bloom

#### => Neo4j CQL :

- ~ Introduction to Neo4j CQL
- ~ Neo4j CQL clauses
- ~ Neo4j CQL Functions
- ~ Neo4j CQL Data Types
- ~ Neo4j CQL operators
- ~ Neo4j CQL Boolean operators
- ~ Neo4j CQL Comparison operators
- ~ Node Creation in Neo4j CQL
- ~ Relationship creation in Neo4j CQL

#### => Neo4j CQL Operators :

- ~ Neo4j CQL Operators
- ~ Comparison Operators
- ~ Boolean Operators
- ~ String Operators
- ~ List Operators
- ~ Regular Expression
- ~ String matching

#### => Neo4j clauses :

- ~ Match Clause
- ~ Optional Match Clause
- ~ Where Clause
- ~ Count Function
- ~ Return Clause
- ~ Order by Clause
- ~ Limit Clause
- ~ Skip Clause

## => Introduction To Kafka :

- ~ *What is Kafka?*
- ~ *What is a messaging queue?*
- ~ *What is the Pub-Sub Model?*
- ~ *How Do Real Time Data Streaming Services work?*
- ~ *Advantages of Kafka*
- ~ *Use Cases Of Kafka*

## => Kafka Architecture :

- ~ *Kafka Cluster*
- ~ *Brokers*
- ~ *Topics*
- ~ *Partitions*
- ~ *Producer-Consumer*
- ~ *Offset Management*
- ~ *Replicas*
- ~ *Commits*
- ~ *Sync & Async Commits*

## => Kafka Producer-Consumer on Confluent Cloud :

- ~ *Confluent Kafka Setup*
- ~ *Topic Creation*
- ~ *Schema Registry*
- ~ *Key, Value Message*
- ~ *Message in Kafka Topics based on Random and Constant Keys*
- ~ *Kafka Producer Code with Serialisation*
- ~ *Kafka Consumer Code with De-Serialization*
- ~ *Consumer Groups*
- ~ *Working with JSON, CSV Data*

## => Python Programming :

- ~ *Python Installation*
- ~ *Python Basics*
- ~ *Input/Output*
- ~ *Data Types*
- ~ *Variables*
- ~ *Operators*
- ~ *List, Tuple, Dictionary, Set*
- ~ *List Comprehension*
- ~ *Dictionary Comprehension*
- ~ *Control Flow*
- ~ *Functions*
- ~ *Lambda Functions*
- ~ *Map(), Filter(), Reduce()*
- ~ *Regex*
- ~ *Object Oriented Concepts*
- ~ *Exception Handling*

## => Getting started with Spark :

- ~ *What is Spark and what is the purpose?*
- ~ *Why is Spark faster than Hadoop?*
- ~ *What is in-memory computation?*
- ~ *Features of Spark*
- ~ *Explain unified architecture of Spark*
- ~ *Components of the Spark unified architecture - Cluster, Master Node, Worker Node, Driver, Executor, Partition, Task*
- ~ *Spark Context, Spark Session, Driver Program*
- ~ *Deployment Modes in Spark - Client mode, Cluster Mode*
- ~ *Application Master in Spark Cluster*
- ~ *Spark Submit Utility with Parameters*

## => Spark on Azure databricks :

- ~ *What is databricks?*
- ~ *Why databricks in the cloud?*
- ~ *How to create an Azure databricks cluster?*
- ~ *Demo provision databricks, clusters and workbook*
- ~ *Pyspark in databricks notebook*
- ~ *Demo mount data lake to databricks DBFS*
- ~ *Demo Explore, Analyse, Clean, Transform and load data in databricks*
- ~ *Databricks monitoring*

## => Spark Terminologies and RDD :

- ~ *Overview of RDD's*
- ~ *Features of RDD*
- ~ *Actions & Transformations*
- ~ *Job, Stage and Task creation in Spark Application*
- ~ *Lineage Graph in Spark*
- ~ *Lazy Evaluation in Spark*
- ~ *RDD and pair RDDs and RDD performance*
- ~ *Limitations of Spark RDD*
- ~ *RDD persistence - Persist & Cache*
- ~ *Data loading in RDD*
- ~ *Partitions*
- ~ *Map, Flat Maps and Filters*
- ~ *Group by*
- ~ *Group by key*
- ~ *Reduce by key*
- ~ *Map partitions*
- ~ *Union*
- ~ *Join*
- ~ *Distinct*

- ~ Coalesce
- ~ Key by
- ~ Partition by
- ~ Zip
- ~ Collect
- ~ Reduce by key
- ~ Aggregate
- ~ Shared variables and broadcast variables
- ~ Accumulators

=> Spark SQL, DataFrames and Datasets :

- ~ Introducing Spark SQL
- ~ Introducing datasets and DataFrame
- ~ Data sources
- ~ Distributed SQL engine
- ~ Creating DataFrame
- ~ DataFrame operations
- ~ DataFrame from csv
- ~ DataFrame from db tables
- ~ DataFrame from hive NoSQL table
- ~ DataFrame from json
- ~ DataFrame from RDD
- ~ Different operations on DataFrame
- ~ Filter
- ~ Join
- ~ Group
- ~ Aggregation
- ~ Having
- ~ Where
- ~ User defined function(UDF)
- ~ Grouping aggregation
- ~ Multiple grouping
- ~ More aggregation
- ~ Hash aggregation
- ~ Spark SQL vs RDD
- ~ Executing SQL commands and SQL-style functions on a DataFrame
- ~ Using DataFrames instead of RDD's
- ~ Different operations with dataframes with DataFrames
- ~ Word Count with DataFrames
- ~ DataFrames vs RDDs
- ~ Operations on DFs
- ~ Parquet files with Spark Sql Read, Write, Partitioning, Merging schema
- ~ ORC files
- ~ JSON files

=> Spark streaming :

- ~ Basic concepts of Spark Streaming
- ~ Linking
- ~ Initialising Streaming Context
- ~ Discretized Streams (DStreams)
- ~ Input DStreams and Receivers
- ~ Transformations on DStreams
- ~ Output operations on DStreams
- ~ DataFrame and SQL operations
- ~ MLlib operations
- ~ Caching / Persistence
- ~ Checkpointing
- ~ Accumulators, Broadcast Variables, and Checkpoints
- ~ Deploying applications
- ~ Performance tuning
- ~ Reducing the batch processing times
- ~ Setting the right batch interval
- ~ Memory tuning
- ~ Sliding window operations
- ~ Overview Spark Streaming and Structure Streaming and kafka streaming with kafka
- ~ Developing Spark Streaming applications Integration with Hbase
- ~ Kafka Twitter data setup
- ~ Writing Producer in Python
- ~ Writing Consumer in Python
- ~ Kafka Integration with Spark Streaming
- ~ Fault-tolerance semantics
- ~ Spark Cassandra

=> Spark Structure streaming :

- ~ Handling Event-time and Late Data
- ~ API using Datasets and DataFrames
- ~ Creating streaming DataFrames and streaming Datasets
- ~ Input Sources
- ~ Schema inference and partition of streaming DataFrames/Datasets
- ~ Operations on streaming DataFrames/Datasets
- ~ Basic Operations - Selection, Projection, Aggregation
- ~ Window Operations on Event Time
- ~ Handling Late Data and Watermarking
- ~ Types of time windows
- ~ Join Operations
- ~ Stream-static Joins
- ~ Stream-stream Joins
- ~ Inner Joins with optional Watermarking
- ~ Outer Joins with Watermarking

- ~ *Semi Joins with Watermarking*
- ~ *Support matrix for joins in streaming queries*
- ~ *Streaming Deduplication*
- ~ *Policy for handling multiple watermarks*
- ~ *Arbitrary Stateful Operations*
- ~ *Unsupported Operations*
- ~ *Limitation of global watermark*
- ~ *State Store*
- ~ *HDFS state store provider*
- ~ *RocksDB state store implementation*
- ~ *State Store and task locality*
- ~ *Starting Streaming Queries*

=> Spark configuration, monitoring and tuning :

- ~ *Understand components of spark cluster*
- ~ *configure spark to modify the spark properties, environmental variables, or logging properties*
- ~ *Monitor Spark using the web UIs, metrics, and external instrumentation*

=> Apache Airflow - Workflow Management Platform :

- ~ *Introduction and Setup of Airflow*
- ~ *Components of Airflow*
- ~ *Installing Airflow on mac*
- ~ *Installing Airflow on linux*
- ~ *Installing Airflow on windows*
- ~ *Run Airflow locally*
- ~ *Introduction to the Airflow UI*
- ~ *What you need to know about the UI*
- ~ *Introduction to the Airflow CLI*

=> Core concepts of Airflow :

- ~ *What is DAG?*
- ~ *DAG skeleton*
- ~ *Default arguments*
- ~ *Instantiate a DAG*
- ~ *Jinja templating with Airflow*
- ~ *What are tasks?*
- ~ *What are operators?*
- ~ *How to set up dependencies?*
- ~ *What are hooks*
- ~ *What are executors*

=> Advanced concepts in Airflow :

- ~ *Adios repetitive patterns*
- ~ *Minimising DAG's with SubDAG*
- ~ *Adios SubDAG, welcome task groups!*
- ~ *Sharing data between tasks with xcoms*
- ~ *Choosing a specific path in your DAG*
- ~ *How Tasks get triggered?*

=> Testing Airflow DAGS :

- ~ *Load test DAG's*
- ~ *Unit test DAG's and operators*
- ~ *Unit test custom operators*

=> Docker Image for Apache Airflow :

- ~ *Introduction to Docker*
- ~ *Why custom image?*
- ~ *How to build your own image?*
- ~ *Extending vs. customising the image*
- ~ *Executors*
- ~ *Configure celery executors*
- ~ *Running Airflow on docker with celery executor*
- ~ *Configure local executors*
- ~ *Running Airflow on docker with local executor*
- ~ *Service level agreement*
- ~ *Security: Authentication, Roles, Encryption*

=> Monitoring Airflow :

- ~ *Airflow monitoring with StatsD*
- ~ *Airflow monitoring with Prometheus*
- ~ *Airflow monitoring with Grafana*
- ~ *Error tracking with Sentry*

=> Big data on cloud :

- ~ *Introduction to cloud*
- ~ *Introduction to Cloud Computing*
- ~ *Cloud models*
- ~ *Different cloud providers*
- ~ *Regions and Availability Zones*
- ~ *Understanding regions and availability zones in Azure*
- ~ *Creating microsoft Azure account*
- ~ *Resource Hierarchy*
- ~ *Understanding resource hierarchy*
- ~ *Demo on resource hierarchy*
- ~ *Resource groups, subscription and management groups*
- ~ *Active directory*

=> AWS EC2 :

- ~ *Launch a Basic EC2 Instance*
- ~ *Different Types of instances - Reserved, On-Demand, Spot, Dedicated*
- ~ *Different configurations of EC2 machines*

- ~ Attaching detaching of EBS Volume in EC2
- ~ Practising few commands on EC2

#### => AWS IAM :

- ~ The Mechanics behind IAM
- ~ Managing IAM Users
- ~ IAM Administration (Guide) (Listing, Deleting Users & Accounts)
- ~ Managing Permissions for IAM Users
- ~ Changing IAM User Permissions
- ~ Creating and Administering IAM Groups
- ~ Creating and Administering IAM Group Policies
- ~ Assigning Preset and Custom Group Policies

#### => AWS Secret Manager :

- ~ Create and Maintain secrets
- ~ Accessing credentials from Secret Manager using Boto3

#### => AWS S3 :

- ~ Buckets
- ~ Objects
- ~ Upload, Delete Files
- ~ Data Encryptions
- ~ Pricing & Data Limitation on S3
- ~ S3 Versioning
- ~ Version ID
- ~ Bucket policy
- ~ Notifications from S3
- ~ Work with S3 using AWS CLI

#### => AWS Lambda :

- ~ What is AWS Lambda and Why is it needed?
- ~ Features & Limitations of Lambda
- ~ Hello world program using Lambda
- ~ Auto trigger Lambda Function based on S3 file upload notification
- ~ Access other services from Lambda

#### => AWS EMR :

- ~ Setting up EMR Cluster
- ~ Install Spark, Hive, Hadoop
- ~ Resource types in EMR cluster
- ~ Data Processing on EMR Cluster
- ~ AWS Glue
- ~ Setting up cluster in Glue
- ~ Properties of Glue
- ~ Creating Catalogues in Glue
- ~ Read partitioned Data
- ~ Bulk and Incremental data processing from S3 in Glue
- ~ Data Processing in Glue
- ~ Glue jobs and Triggers

#### => AWS SNS :

- ~ What is SNS?
- ~ How does SNS work?
- ~ Creating SNS Topics and subscribing
- ~ Different types of subscribers
- ~ Sending notifications via SNS

#### => AWS SQS :

- ~ What is SQS?
- ~ Different types of SQS?
- ~ At-Least once and Exactly once delivery via SQS
- ~ Ingesting data to SQS
- ~ Inflight messages
- ~ Consume data from SQS
- ~ Dead Letter Queue

#### => AWS DMS :

- ~ What is DMS?
- ~ Capturing CDC event in DMS where Database as a source
- ~ Capture CDC events and sending it to downstream systems

#### => AWS Kinesis :

- ~ Creating Kinesis Streams
- ~ Ingesting real time data in Kafka Streams
- ~ Consume real time data from Kafka Streams

#### => AWS RDS :

- ~ MySQL Database using AWS RDS
- ~ Scalability & Limitations of AWS RDS
- ~ Creating tables and loading data in AWS RDS
- ~ Querying data from RDS

#### => AWS Athena :

- ~ What is serverless database services
- ~ Athena vs RDS
- ~ Table metadata in Athena for the data residing in S3
- ~ Creating table for S3 data
- ~ Querying S3 data using Athena

#### => AWS Redshift :

- ~ What are Data warehousing services?
- ~ Architecture of Redshift



- ~ Resources types in Redshift Cluster
- ~ Creating tables in Redshift
- ~ Internal & External tables
- ~ Partitioning, Sort Keys, Column compression
- ~ Querying data in Redshift
- ~ Views & Materialised views in Redshift

=> AWS Dynamo :

- ~ Architecture of DynamoDB
- ~ Creating tables and Ingesting data into DynamoDB table
- ~ Querying data from DynamoDB

=> AWS Cloudwatch :

- ~ Cron based triggers
- ~ Event pattern based triggers
- ~ Monitoring & Alerting using Cloudwatch

=> AWS QuickSight :

- ~ Creating business dashboards using Quick sight

=> Introduction to azure cloud computing :

- ~ Azure services overview
- ~ Managed and unmanaged service
- ~ Demo create Azure SQL Database service

=> Azure sql database :

- ~ Module Introduction
- ~ Introduction
- ~ Why choosing SQL Server in Azure
- ~ Azure IaaS vs PaaS database offerings
- ~ SQL server PaaS deployment options
- ~ Introduction to Azure sql server in virtual machine
- ~ SQL Server in Azure virtual machine
- ~ SQL server in azure virtual machine
- ~ Introduction Azure single database
- ~ Demo Azure single database
- ~ Purchasing Models and service tier
- ~ Azure database vs azure data warehouse
- ~ Introduction elastic data pool
- ~ Azure Elastic Database
- ~ Azure Elastic Database
- ~ Introduction managed instance Database
- ~ Azure managed instance Database
- ~ Difference between on premises and managed instance
- ~ Service tiers for managed instance
- ~ Management operations
- ~ Demo managed instance

=> Azure synapse :

- ~ Module introduction
- ~ Why warehouse in the cloud?
- ~ Traditional vs modern warehouse architecture
- ~ What is synapse analytics service?
- ~ Demo create dedicated sql pool
- ~ Demo connect sql pool with SSMS
- ~ Demo create Azure synapse analytics workspace
- ~ Demo explore synapse studio v2
- ~ Demo create dedicated sql pool and spark pool from inside synapse studio
- ~ Demo analyse data using dedicated sql pool
- ~ Analyse data using apache spark notebook
- ~ Demo analyse data using serverless sql
- ~ Demo data factory copy tool from synapse integrate tab
- ~ Demo monitor synapse analytics studio
- ~ Azure synapse a game changer
- ~ Azure synapse benefits

=> Azure databricks :

- ~ Spark Basics
- ~ Why is spark difficult?
- ~ Why databricks in the cloud?
- ~ How to save databricks demo cost
- ~ Demo provision databricks, clusters and workbook
- ~ Demo mount data lake to databricks DBFS
- ~ Demo Explore, Analyse, Clean, Transform and load data in databricks
- ~ Azure databricks cluster
- ~ Azure databricks other important components
- ~ Databricks monitoring

=> Azure data factory :

- ~ What is a Data Factory?
- ~ Data factory in azure ecosystem
- ~ Provision Azure data factory instance
- ~ Data factory components
- ~ Data factory pipeline and activities
- ~ Data factory linked service and datasets
- ~ Data factory integration runtime
- ~ Data factory triggers
- ~ Data factory copy data activity demo
- ~ Copy data activity using author demo
- ~ Secure input and output property
- ~ User properties

- ~ Data factory parameters
- ~ Data flow concept
- ~ Mapping data flow
- ~ Wrangling data flow
- ~ Monitoring
- ~ Metrics and diagnostic settings

#### => Introducing Google Cloud Platform :

- ~ Google platform fundamentals overview.
- ~ Google cloud platform Big Data products.

#### => Compute and Storage Fundamentals :

- ~ CPUs on demand (compute engine).
- ~ A global file system (cloud storage).
- ~ CloudShell.
- ~ Set up an Ingest-Transform-Publish data processing pipeline.

#### => Data Analytics on the Cloud :

- ~ Stepping-stones to the cloud.
- ~ Cloud SQL: your SQL database on the cloud.
- ~ Importing data into CloudSQL and running queries.
- ~ Spark on Dataproc.
- ~ Machine Learning recommendations with Spark on Dataproc.

#### => Scaling Data Analysis :

- ~ Fast random access.
- ~ Datalab
- ~ BigQuery.

#### => Data Processing Architectures :

- ~ Message-oriented architectures with Pub/Sub.
- ~ Real time streaming using Pub/Sub
- ~ Creating pipelines with Dataflow.
- ~ Reference architecture for real-time and batch data processing.
- ~ Google Online Transfer
- ~ Cloud Storage Transfer
- ~ Google Cloud BigTable
- ~ Google Cloud Dataflow
- ~ Google Cloud Dataproc
- ~ Google Cloud Pub/Sub
- ~ Google Cloud Composer
- ~ Google Cloud Data Fusion
- ~ Automating ETL jobs with composer and fusion
- ~ Google Cloud data catalogue
- ~ Google Data studio
- ~ Architecture: Optimising large-scale Ingestion
- ~ GCP Big Data Outro

#### => Dashboarding Tools :

- ~ PowerBI
- ~ Tableau

#### => Enterprise Big Data ETL Tools :

- ~ Introduction to ETL from Talend Studio- Integration with HDFS, Hive, Sqoop, Spark etc
- ~ Introduction to ETL from Informatica BDM- Integration with HDFS, Hive, Sqoop, Spark etc

#### => PROJECT AND INTERVIEW PREPARATION :

- ~ End-to-end Big Data batch and streaming(Real-Time) Pipeline
- ~ 15+ industry ready projects.
- ~ Involving all Major components like Sqoop, Hdfs, Hive, Hbase, Spark... etc.
- ~ Interview Preparation Tips
- ~ Sample Resume
- ~ 300+ Mock Interview Recordings
- ~ Mock Interview QA
- ~ Interview Questions
- ~ How to Handle Various Interview Round Qs
- ~ Career Guidance
- ~ One to One Resume Discussion
- ~ Certification

# Industry Ready Data Science Projects Tech Neuron

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING PROJECT

Course link : <https://ineuron.ai/course/Industry-Ready-Data-Science-Projects-Tech-Neuron>

## Course Description :-

Ready to use end-to-end data science projects for real-world business use cases. We will be discussing projects from very scratch such as understanding problem statements, capturing requirements, and various aspects of project design using different documentation such as High-Level Design, Low-Level Design, and Architecture Design. Practical use of MLOPS practices using tools such as MLFLOW, Wandb. Pipeline implementation for training, retraining, and inferencing. Designing dashboard to present important KPIs to monitor system and model performance and generate alert to notify the appropriate parties to address serious problems if it is about to occur.

## Course Features :-

- => Online Instructor-led learning
- => Doubt Clearing
- => Proper Roadmap for building AI projects
- => Lifetime Dashboard access
- => Recording of Live Class
- => Material
- => Interview Questions
- => Resume Building
- => Career Guidance
- => Quiz in every module - Based on Real Time Questions
- => Certificate
- => Industry Level Projects and Case studies
- => Capstone Projects

## What you will learn :-

- => System Architecture
- => High Level Design
- => Component Selection
- => Low Level Design
- => Core utility design
- => Deployment Architecture
- => Multistage pipeline for CI/CD
- => ML Pipeline Understanding
- => Training Pipeline Implementation
- => Inference Pipeline Implementation
- => Retraining Pipeline Implementation
- => Deployment of ML Pipeline on Cloud
- => Monitoring of System and Model Performance

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Knowledge of Python
- => Knowledge of Machine Learning

## Instructors :-

=> Avnish Yadav :

~ 3+ years of experience in various domains such as data scientist, data analyst, database developer, and .net developer. Implemented various sophisticated business requirements, performed an analysis of various data to capture insights and hidden patterns. Fine and tuned various regression and classification-based algorithms for prediction. Implemented various ETL pipelines to fulfil the business requirement. Automated various machine learning pipelines such as data loading, data cleaning, data validation, model selection, model tuning, and model monitoring pipeline. Implemented

machine learning pipeline in azure machine learning studio. I have a keen interest to solve complicated machine learning problems to fulfil business requirements.

=> Ketan Gangal :

~ I have worked in data science for more than two years, and I have a track record of successfully implementing data science pipelines in production with practical expertise using ML-Ops, deep learning & machine learning. I also love sequence processing because it is deeply inspired by humans as our feeling, thoughts, emotions, sensations, language are sequential in nature if we can enable machine to understand sequence of information and act accordingly we can make significant progress towards true artificial intelligence.

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Project - Sensor Fault Detection :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pandas, Sklearn, Mlflow, Cloud, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Customize Model Training
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation
- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud
- ~ Schedule and Orchestrate Training Pipeline
- ~ Deployment of Inference Pipeline as an API
- ~ Deployment of Retraining Pipeline
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

=> Project - Financial Product Complaint :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pytorch, Cloud, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation

- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud
- ~ Schedule and Orchestrate Training Pipeline
- ~ Deployment of Inference Pipeline as an API on Elastic Container Serving
- ~ Deployment of Retraining Pipeline
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

#### => Project - Face Authenticator :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, MongoDB, Deepface, Flask, Docker, EC2 Instance, Git, Github, SQL
- ~ Face Authenticator Pipeline
- ~ Understanding Face Authenticator mechanism
- ~ Face Registration Pipeline
- ~ Face Identification Pipeline
- ~ Face Registration Pipeline
- ~ Capturing Images of a Person
- ~ Generating Embedding of Facial Image
- ~ Save Embedding in Database
- ~ Face Identification Pipeline
- ~ Detecting face of a Person at login portal
- ~ Generate embedding of captured face
- ~ Search Generated Embedding in DB using similarity metrics Triplet Loss
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

#### => Project - Embedding based search engine :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pytorch, Hugging Face, Transformer, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation
- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud

- ~ *Schedule and Orchestrate Training Pipeline*
- ~ *Deployment of Inference Pipeline as an API*
- ~ *Deployment of Retraining Pipeline*
- ~ *Monitoring of System and Model Performance*
- ~ *Importance of Monitoring*
- ~ *Visualization of KPI and Other Indicator*
- ~ *System and Model Performance Visualization*
- ~ *Implementation of Alert and Notification to prevent Failure*
- ~ *Project Conclusion*

# Introduction to AIOps

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : <https://ineuron.ai/course/Introduction-to-AIOps>

## Course Description :-

Artificial Intelligence Operations is the most in-demand technological skill these days (AIOps). It facilitates the use of DevOps techniques in the creation of AI products. This course will provide a thorough theoretical knowledge of the AIOps concept.

## Course Features :-

- => Quizzes
- => Completion Certificate
- => Assignment

## What you will learn :-

- => Introduction of AIOps
- => AI project generic steps
- => Project workflow
- => Project deployment

## Requirements :-

- => Basic understanding of AI/ML/DL
- => A System with a decent internet connection
- => Your dedication

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

=> AIOps Introduction Theory :

- ~ Introduction 1 Preview
- ~ Introduction 2 Preview
- ~ Introduction 3
- ~ Challenges
- ~ AIML Generic Steps
- ~ Level 0 workflow
- ~ Level 0 characteristics observation
- ~ Level 1 workflow
- ~ Level 1 aim
- ~ Level 1 characteristics
- ~ Frequently used terms
- ~ Data validation
- ~ Model validation Offline
- ~ Model validation Online
- ~ Feature Store
- ~ Metadata Storage
- ~ Pipeline Trigger
- ~ Final summary
- ~ Level 2 aim
- ~ Level 2 CI CD workflow detail discussion part 1
- ~ Level 2 CI CD workflow detail discussion part 2
- ~ Level 2 More on CI
- ~ Level 2 More on CD
- ~ Level 2 Deployment types
- ~ Level 2 Summary final

# C Sharp Programming

---

Topic Name : PROGRAMMING

Sub-topic Name : C Sharp

Course link : <https://ineuron.ai/course/C-Sharp-Programming>

## Course Description :-

Learn the fundamentals of C# programming.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Understanding .NET SDK CLI ,creatingbuilding & running first project.
- => Framework vs Language
- => C# is case sensitive and understanding compile errors in CLI.
- => Creating projects using visual studio
- => Project and Solution in visual studio
- => Building and compile using visual studio
- => Difference between C# and .NET
- => CLR
- => Architecture of .NET Framework
- => C# Operators
- => C# Operator Precedence & Associativity
- => C# Bitwise Operators
- => C# Arrays
- => C# Multidimensional Arrays
- => C# Inheritance
- => C# using
- => C# Type Conversion & Casting
- => C# Preprocessor Directives

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction :
  - ~ DownloadInstallation of .NET &.NET core vs .NET Framework
  - ~ Understanding .NET SDK CLI ,creatingbuilding & running first project.
  - ~ Framework vs Language
  - ~ C# is case sensitive and understanding compile errors in CLI.
  - ~ Introduction, Download and Installation of VS Code editor.
  - ~ Opening folder in VS code , intellisense and terminal.
  - ~ NET Compilation process & Intermediate language code
  - ~ Visual studio vs VS Code vs Visual studio for Mac
  - ~ Installing and configuring visual studio work loads part 1
  - ~ Installing and configuring visual studio work loads part 2
  - ~ Creating projects using visual studio
  - ~ Project and Solution in visual studio
  - ~ Building and compile using visual studio
  - ~ Running projects as startup
  - ~ Understanding basic code class, namespace and scopes
  - ~ Assembly , Exe and DLL Part 1
  - ~ Assembly , Exe and DLL Part 2



## => C# Introduction :

- ~ *Introduction*
- ~ *Difference between C# and .NET*
- ~ *CLR*
- ~ *Architecture of .NET Framework*
- ~ *C# Hello World*
- ~ *C# Keywords & Identifiers*
- ~ *C# Variables and Data types*
- ~ *Demo C# Variables*
- ~ *C# Operators*
- ~ *C# Operator Precedence & Associativity*
- ~ *C# Bitwise Operators*
- ~ *C# Basic IO*
- ~ *C# Expressions & Statements*
- ~ *C# Comments*

## => C# Flow Control :

- ~ *C# if...else*
- ~ *C# switch Statement*
- ~ *C# Ternary Operator*
- ~ *C# while Loop*
- ~ *C# for Loop*
- ~ *C# Nested Loops*
- ~ *C# break Statement*
- ~ *C# continue Statement*

## => C# Arrays :

- ~ *C# Arrays*
- ~ *C# Multidimensional Arrays*
- ~ *C# Jagged Array*
- ~ *C# foreach Loop*

## => C# OOP part 1 :

- ~ *C# Class and Objects*
- ~ *C# Methods*
- ~ *C# Access Modifiers*
- ~ *C# Variable Scope*
- ~ *C# Constructors*
- ~ *C# this Keyword*
- ~ *C# static Keyword*
- ~ *C# Strings*

## => C# OOP (II) :

- ~ *C# Inheritance*
- ~ *C# Abstract Class & Methods*
- ~ *C# Nested Class*
- ~ *C# Partial Class & Method*
- ~ *C# Sealed Class & Method*
- ~ *C# Interface*
- ~ *C# Method Overloading*
- ~ *C# Constructor Overloading*

## => Additional Topics :

- ~ *C# using*
- ~ *C# Type Conversion & Casting*
- ~ *C# Preprocessor Directives*
- ~ *C# Namespaces*
- ~ *C# struct*

# Marketing Mix Modelling

---

Topic Name : DATA SCIENCE

Sub-topic Name : STATS

Course link : <https://ineuron.ai/course/Marketing-Mix-Modelling>

## Course Description :-

This course is for marketing and Data Analysis students who are required to integrate both statistics and marketing. Detailed Explanation of Advanced regression technique and mathematics behind each metrics is given in this course.

## Course Features :-

- => Assignment
- => Quiz
- => Downloadable Resources
- => Completion Certificate

## What you will learn :-

- => Marketing strategy
- => Workflow of models
- => Data types and sources
- => Correlation
- => Regression Techniques
- => Statistics behind regression
- => Optimization

## Requirements :-

- => Prior knowledge in Marketing
- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Basics of Python

## Instructors :-

=> Bharath J P V :

~ Enthusiast Data Scientist with a strong background in Mathematics and Statistics. Completed My Master in Statistics. Have experience teaching Mathematics and Statistics for more than a year. I thought for more than 1000 students and helped them make their careers in their respective fields. I believe in "we rise by lifting others". Following this principle, I hope to make your life easier.

## Curriculum details :-

=> Introduction :

- ~ Course introduction Preview
- ~ Who is this course for?
- ~ Course overview
- ~ Course outcome

=> Introduction to Marketing Mix models :

- ~ Introduction to Marketing Mix Models Preview
- ~ Understanding Marketing Mix Models

=> Data for Marketing Mix Models :

- ~ Workflow of marketing mix models
- ~ Data request and Data Sources
- ~ Data Categories

=> Data PreProcessing :

- ~ Data Exploration
- ~ Data Processing
- ~ AdStock
- ~ Decay Function and Diminishing Returns

=> Regression analysis :

- ~ Introduction to correlation and regression
- ~ Variance and Coefficient of determination
- ~ Standard Error and Residual Analysis
- ~ VIF, Durbin-Watson and Jarque-Bera Test
- ~ F-Statistic and t-Statistic Preview
- ~ Model Fitting and evaluation

=> Advance regression technique :

- ~ Feature Selection

- ~ *Contribution, ROI and Diminishing Returns*
- ~ *Log-Linear Model*
- ~ *Nested Model*

=> Optimization :

- ~ *Predictions and Optimization*

# UI Designing using Figma

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/UI-Designing-using-Figma>

## Course Description :-

Students will learn how to design user interfaces with Figma in this course. They will be introduced to the Figma interface in the first session, which includes how to use features to make the designs responsive. Students will also be taught how to make components to establish a library of design elements that can be reused across projects. They will learn real-time team communication and how to work on a project with other designers. Also, the research and prototyping process is explained in the later part of the course. Through a practical hands-on approach, students who complete this course will be able to apply their knowledge to solve real-world user interface problems.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Live Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to Figma
- => Uses Figma
- => Advantages of Figma
- => Overview of Figma Interface
- => Pen tools
- => Masking in Figma
- => Plugs in Figma
- => Figma components
- => Animations using Figma

## Requirements :-

- => Interest to learn
- => Dedication
- => System with good internet connection

## Curriculum details :-

- => Introduction :
  - ~ Course Introduction
  - ~ For whom is this course designed?
  - ~ Course Overview
  - ~ Course Outcome
  - ~ What is UI design?
  - ~ Where can it be used?
  - ~ Why should you learn it?
  - ~ What can you do with Figma?
  - ~ Who can use Figma?
  - ~ Why use Figma?
  - ~ What are some other similar design tools?
- => Assignment 1 :
  - ~ List out features of Figma and how they are different from its competitor designing tools.
- => Getting started with Figma :
  - ~ Which plan is best for you?
  - ~ Running Figma in browser and desktop app
  - ~ Figma setup

- ~ Sign up for Figma
- ~ Install fonts
- ~ How to navigate? (sample file)
- ~ Why should you start with a sample file?

=> Assignment 2 :

- ~ State the difference between Fonts, Typefaces and Typography
- ~ Install any of your favorite fonts and explain the entire process.

=> Interface Overview :

- ~ What is layer and layer panel?
- ~ What is a properties panel?

=> Assignment 3 :

- ~ Go through the sample file. Make one more page and name it 'Shapes'.

=> Setup the first file :

- ~ What are frames?
- ~ Shapes & Corner radius adjustment
- ~ Colors
- ~ Pen tool
- ~ Masking
- ~ Importing Icons and other Graphics
- ~ Text and Fonts
- ~ Components
- ~ Constraints
- ~ Layers
- ~ Assets
- ~ Export

=> Assignment 4 :

- ~ On the page 'Shapes', make a frame of size 1200\*900.
- ~ Make a rectangle, give it some color and stroke of your choice.
- ~ Make a button of size 100\*60 with text.

=> Requirements for designing an app and brief about the research :

- ~ What are research methods?
- ~ Why is research necessary?
- ~ Gather info and do competitive research
- ~ Set a user flow
- ~ How to create wireframes?
- ~ How to decide on a theme? What does it include?

=> Assignment 5 :

- ~ What kind of information is acquired in primary and secondary research?

=> Factors to consider while designing an app :

- ~ Navigation
- ~ Usability
- ~ Picking colors and fonts
- ~ Visual Hierarchy
- ~ Taking feedbacks

=> Assignment 6 :

- ~ Explain various vital factors that are considered while designing.

=> How to make use of already created components :

- ~ Community
- ~ Plugins, Icons, Files

=> Assignment 7 :

- ~ What are plugins? How can they be accessed?

=> Project:- Start designing/home page demo :

- ~ Buttons
- ~ Scaling
- ~ Navbar
- ~ Hero banner
- ~ Different sections like features and testimonials
- ~ Footer
- ~ Prototyping
- ~ Basics of Animation in Figma
- ~ Hover Interaction
- ~ How to add designers or clients?
- ~ How to export a file?

=> Assignment 8 :

- ~ Design every possible state of a button.
- ~ Try to find out the logo design process of an application you use the most.
- ~ Visit a website of your choice to study how their product resonates with the website design.

=> Conclusion :

- ~ What makes Figma different from other tools?
- ~ Future journey as a UI/UX Designer

# Pro Python Programming Language

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Python-Programming-Language>

## Course Description :-

This course is designed mostly for Python Coding test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Python Theoretical Test

=> Python Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Python Coding Test :

- ~ Python Test 1
- ~ Python Test 2
- ~ Python Test 3
- ~ Python Test 4
- ~ Python Test 5
- ~ Python Test 6
- ~ Python Test 7
- ~ Python Test 8
- ~ Python Test 9
- ~ Python Test 10

# Full Stack Blockchain Development

---

Topic Name : BLOCKCHAIN

Sub-topic Name : BLOCKCHAIN MASTERS

Course link : <https://ineuron.ai/course/Full-Stack-Blockchain-Development>

## Course Description :-

Full Stack Blockchain Development course is a live mentor-led certification program with by iNeuron. In this course you will learn the entire stack required to work in Permissionless Blockchain development. This course focuses on latest Blockchain industry standards like Ethereum Blockchain, Solidity, Decentralized Autonomous Organisations, Decentralized Finance, Non Fungible Tokens, Polygon Network, Polkadot Blockchain, Oracles along with complete development stack in Javascript and many more Blockchain concepts.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Web 1.0 vs Web 2.0 vs Web 3.0
- => What is Blockchain technology?
- => Bitcoin Blockchain
- => Ethereum Blockchain
- => Solidity
- => Oracles
- => DAO
- => DeFi
- => NFT
- => Layer 2 Blockchain
- => Truffle Suite
- => Hardhat
- => Polkadot

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sanjeevan Thorat :

~ Data Scientist and Blockchain developer, with experience in developing and managing end to end solutions. I have hands-on experience in Python Programming Language, Machine Learning Deep Learning and Natural language processing. Blockchain development experience in smart contracts, Decentralised Finance applications, DAOs, NFTs and Oracles running on Ethereum and Polygon blockchains. I have worked with various clients for different industry projects in the blockchain space. I specialize in building smart contracts on the Ethereum blockchain along with JavaScript integration for enhancing user experience to generate maximum returns on investment.

=> Navin Reddy :

~ I am Corporate Java trainer. Since past few years successfully trained many professionals at JP Morgan, Accenture, Polaris and L&T infotech. My youtube channel "Telusko" presently has 1.7 million subscribers. Passionate about Java Technology for over a decade and moved on as a corporate trainer. I am certified blockchain developer and Currently, building Applications running on Blockchain (dapps).

## Curriculum details :-

- => Course Introduction :
  - ~ Course overview
  - ~ A brief history of internet technologies
  - ~ Web 1.0 vs Web 2.0 vs Web 3.0

- ~ *What is Blockchain technology?*
- ~ *Why do we need Blockchain technology?*
- ~ *The connected world and the Blockchain: A disruptive computing paradigm*
- ~ *Centralized vs Decentralized networks*
- ~ *Distributed Systems overview*

=> Web Development :

- ~ *What is Web Development?*
- ~ *Client-Server Architecture*
- ~ *What are APIs?*
- ~ *What is Front-end web development?*
- ~ *What is Back-end web development?*
- ~ *Components of Full-Stack Web Development Applications*

=> HTML :

- ~ *How do websites work?*
- ~ *Preview*
- ~ *HTML vs CSS vs Javascript*
- ~ *HTML files*
- ~ *Doctype & HTML Boilerplate*
- ~ *Spaces & Line Breaks*
- ~ *Heading Tag*
- ~ *Paragraph & Pre Tag*
- ~ *Difference between Elements, Attributes & Tags*
- ~ *Comments*
- ~ *Useful Tags*
- ~ *Nesting of Tags*
- ~ *Extensions in HTML*
- ~ *Live Server in VSCode*
- ~ *Formatting Tags*
- ~ *Article in HTML*
- ~ *Time & Address Tag*
- ~ *Quote & Cite*
- ~ *Strike*
- ~ *Progress Bar*
- ~ *Anchor Tag Styling*
- ~ *Image Tag*
- ~ *HTML Table*
- ~ *List*
- ~ *Input Tags,iframe*
- ~ *Forms*
- ~ *Video & Audio*
- ~ *iframe*
- ~ *Embed pdf*
- ~ *Maps*
- ~ *Symbols*
- ~ *Meta Tags*
- ~ *SVG*
- ~ *Emoji*

=> CSS :

- ~ *CSS Introduction*
- ~ *Inline vs Internal vs External*
- ~ *Priority between Inline, Internal & External*
- ~ *Multiple Properties in Single Element*
- ~ *Types of Selectors*
- ~ *Priority between Id, Class & Element*
- ~ *Comments*
- ~ *Colors*
- ~ *Backgroud*
- ~ *Border*
- ~ *Height & Width*
- ~ *Padding*
- ~ *Margin*
- ~ *Box Model*
- ~ *Text Properties*
- ~ *Anchor Tag Styling*
- ~ *Fonts*
- ~ *Cursor*
- ~ *!Important in CSS*
- ~ *Box Shadow*
- ~ *Opacity*
- ~ *Filter*
- ~ *Gradient*
- ~ *Overflow*
- ~ *List*
- ~ *Tables*
- ~ *Box Sizing*
- ~ *Inherit & Initial*
- ~ *Object Fit*
- ~ *Pseudo Classes*
- ~ *Pseudo Elements*
- ~ *Display*
- ~ *Position*
- ~ *Z-Index*
- ~ *Floats*
- ~ *2D Transform*
- ~ *Transitions*
- ~ *Flex*



- ~ Flex Direction & Wrap
- ~ Justify & Align in Flex
- ~ Order in Flex
- ~ Grow & Basis in Flex
- ~ Aling Items in Flex
- ~ Grids
- ~ Rows, Columns & Gap in Grids
- ~ Justify & Align in Grids
- ~ CSS Validator (Final Video)

=> Javascript :

- ~ Introduction
- ~ Running Javascript in Browser
- ~ Console
- ~ Strings & Numbers
- ~ var, let & const
- ~ Data Types
- ~ Type Conversions
- ~ Arithmetic Operators
- ~ Assignment Operator
- ~ Comparision Operator
- ~ Logir Not, Or and And
- ~ Swap Numbers
- ~ String Handling
- ~ String Searching
- ~ Arrays
- ~ Objects
- ~ Dates
- ~ Maths
- ~ If & Else
- ~ Challenge - If & Else
- ~ Switch Case
- ~ Challenge - Switch Case
- ~ JS Loops
- ~ For Loops
- ~ Nested Loops
- ~ Break & Continue
- ~ Arrays, Strings & Objects
- ~ For-in
- ~ For-of
- ~ While Loops
- ~ Do while Loops
- ~ Loops Exercies
- ~ Functions
- ~ Variable Scopes in Functions
- ~ Nested Functions
- ~ Parameters & Arguments
- ~ How function is useful?
- ~ Return in Function
- ~ Anonymous Functions
- ~ Calculator Exercise
- ~ Arrow Functions
- ~ forEach
- ~ maps
- ~ String Literals
- ~ Filter, Reduce & Every
- ~ Spread Operator
- ~ Challengege
- ~ Window & Document
- ~ Document Access
- ~ innerText & innerHTML
- ~ HTML Calculator
- ~ Query Selector
- ~ Styling in JS
- ~ Advance DOM Manipulation
- ~ Events
- ~ Basic Events
- ~ Time Events
- ~ Pop-up Boxes
- ~ Error Handling
- ~ Form Validation
- ~ Asynchronous JS
- ~ this keyword
- ~ useStrict
- ~ Hoisting
- ~ Local Storage
- ~ Session Storage
- ~ Cookies
- ~ Cookies vs Local Storage vs Session Storage
- ~ JSON vs Object literals
- ~ API
- ~ Fetching
- ~ Methods & Status Codes
- ~ Post Method
- ~ Put Method
- ~ Guess the Number
- ~ Generators
- ~ Regex

## => The JavaScript Standard Library :

- ~ *The JavaScript Standard*
- ~ *Sets and Maps*
- ~ *Typed Arrays and Binary Data*
- ~ *Pattern Matching with Regular Expressions*
- ~ *Dates and Times*
- ~ *Error Classes*
- ~ *JSON Serialization and Parsing*
- ~ *The Internationalization API*
- ~ *The Console API*
- ~ *URL APIs*
- ~ *Timers*

## => Iterators and Generators :

- ~ *What are Iterators and Generators?*
- ~ *How Iterators Work?*
- ~ *Implementing Iterable Objects*
- ~ *Generators*
- ~ *Advanced Generator Features*

## => Asynchronous JavaScript :

- ~ *What is Asynchronous JavaScript?*
- ~ *Asynchronous Programming with Callbacks*
- ~ *Promises*
- ~ *Async and await*
- ~ *Asynchronous Iteration*

## => Working with Web Browsers :

- ~ *JavaScript in Web Browsers*
- ~ *Web Programming Basics*
- ~ *Events*
- ~ *Scripting Documents*
- ~ *Scripting CSS*
- ~ *Document Geometry and Scrolling*
- ~ *Web Components*
- ~ *SVG: Scalable Vector Graphics*
- ~ *Audio APIs*
- ~ *Location, Navigation, and History*
- ~ *Networking Concepts*
- ~ *Storage*
- ~ *Worker Threads and Messaging*

## => Node js :

- ~ *What is Node.js?*
- ~ *Client-Server Architecture*
- ~ *Single-Threaded Model*
- ~ *Multi-Threaded Model*
- ~ *Multi-Threaded vs Event-Driven*
- ~ *What is Node.js?*
- ~ *Node.js Features*
- ~ *Node.js Installation*
- ~ *Node.js First Example*
- ~ *Blocking vs Non-blocking*
- ~ *Global Objects*
- ~ *File System*
- ~ *Callbacks*
- ~ *Events*
- ~ *Node.js Architecture*
- ~ *NPM(Node Package Manager)*
- ~ *Node.js Modules*
- ~ *Node.js Modules Types*
- ~ *Core Modules*
- ~ *Local Modules*
- ~ *3rd Party Modules*
- ~ *JSON File*
- ~ *Variables*
- ~ *Operators*
- ~ *Functions*
- ~ *Objects*
- ~ *File Systems*
- ~ *Events*
- ~ *HTTP Module*
- ~ *Creating a Web Server using Node.js*
- ~ *Node.js NPM Tutorial*
- ~ *What is NPM?*
- ~ *Main Functions of NPM*
- ~ *Need For NPM*
- ~ *NPM Packages*
- ~ *NPM Installation*
- ~ *JSON File*
- ~ *Node.js Express Tutorial*
- ~ *Introduction to Express.js*
- ~ *Features of Express.js*
- ~ *Getting Started with Express.js*
- ~ *Routing Methods*
- ~ *Building RESTful API with Node.js*
- ~ *What is REST API?*
- ~ *Features of REST API*
- ~ *Principles of REST API*

- ~ *Methods of REST API*
- ~ *Building REST API with Node.js*
- ~ *Contact List MERN App*

=> **React JS :**

- ~ *Introduction to React*
- ~ *Why should you learn React?*
- ~ *Features of React*
- ~ *React applications*
- ~ *React App & JSX*
- ~ *Functional Components*
- ~ *Applying CSS Styles*
- ~ *Click Events*
- ~ *useState Hook*
- ~ *Lists & Keys*
- ~ *Props & Prop Drilling*
- ~ *Controlled Component Inputs*
- ~ *Project Challenge*
- ~ *useEffect Hook*
- ~ *JSON Server*
- ~ *Fetch API Data*
- ~ *CRUD Operations*
- ~ *Fetch Data Challenge*
- ~ *React Router*
- ~ *Router Hooks*
- ~ *Links*
- ~ *Flexbox Components*
- ~ *Axios API Requests*
- ~ *Custom React Hooks*
- ~ *Context API & useContext Hook*
- ~ *Build & Deploy Your React Apps*

=> **Javascript Projects :**

- ~ *Creating shopping cart app with User Interface*

=> **Bitcoin Blockchain :**

- ~ *History of currencies*
- ~ *Fiat currencies*
- ~ *Disadvantages of fiat currencies*
- ~ *Global financial system*
- ~ *How Central Banks work?*
- ~ *The 2008 Global Financial Crisis*
- ~ *Aftermath of 2008 recession*
- ~ *Creation of Bitcoin- A new decentralised digital currency*
- ~ *Bitcoin message hash implementation in Javascript*
- ~ *Immutable ledger practical implementation*
- ~ *Genesis block*
- ~ *Timestamp server*
- ~ *Merkel trees*
- ~ *Bitcoin as a State Transition System*
- ~ *Unspent Transaction outputs(UTXOs) Javascript implementation*
- ~ *Bitcoin whitepaper*
- ~ *What is a block?*
- ~ *Components of a Bitcoin block*
- ~ *Bitcoin Blockchain live implementation*
- ~ *Distributed Blockchain*
- ~ *Centralized vs Distributed Blockchain*
- ~ *consensus mechanism*
- ~ *Why do we need consensus mechanism in Blockchain networks?*
- ~ *Byzantine generals problem*
- ~ *Byzantine fault tolerance- A solution to Byzantine generals problem*
- ~ *BFT javascript implementation*
- ~ *Bitcoin nodes*
- ~ *Bitcoin miners*
- ~ *Blockchain mining operation*
- ~ *Mempool*
- ~ *Bitcoin difficulty adjustment*
- ~ *Bitcoin halving cycle*
- ~ *Competing chain problem*
- ~ *Maintaining immutability - Longest Chain rule*
- ~ *Block validation*
- ~ *consensus rules*
- ~ *Double Spend Validation*
- ~ *Transaction Input and Output Validation*
- ~ *Coinbase Transaction Reward Validation*
- ~ *Coinbase Maturity*
- ~ *Coinbase Transaction Block Height*
- ~ *Signature Check Counting*
- ~ *SigChecks*
- ~ *Mining incentive*
- ~ *Mining optimized hardware*
- ~ *CPU processing power*
- ~ *GPUs for mining*
- ~ *Application Specific Integrated Circuits(ASIC) miners*
- ~ *CPU vs GPU vs ASIC miners*
- ~ *Distributed peer to peer Blockchain live implementation*
- ~ *Distributed peer to peer Blockchain Javascript implementation*
- ~ *Token transaction live implementation using distributed peer to peer blockchain*
- ~ *Coinbase transaction live implementation using distributed peer to peer blockchain*

- ~ Token and Coinbase transaction Javascript implementation
- ~ Bitcoin public key and private key
- ~ Public key and private key generation
- ~ Bitcoin addresses
- ~ Bitcoin digital signatures
- ~ Signing a peer to peer message with private key- Javascript implementation
- ~ Verifying peer to peer message using public key and digital signature-implementation
- ~ Signing and verifying currency transaction- implementation
- ~ Complete Bitcoin Blockchain implementation with transaction signatures

=> Probable attacks in Bitcoin blockchain :

- ~ Sybil Attack
- ~ Race Attack
- ~ Finney Attack
- ~ Vector76 Attack
- ~ 51% Attack

=> Bitcoin Project :

- ~ Building a Blockchain using Javascript

=> Ethereum Blockchain :

- ~ Module overview
- ~ Understanding the drawbacks of Bitcoin blockchain
- ~ Lack of Turing-completeness
- ~ Value-blindness
- ~ Lack of state
- ~ Blockchain-blindness
- ~ Origin of Ethereum- The programmable currency
- ~ The Decentralized Applications revolution and modern state of blockchain systems
- ~ Decentralized Applications vs Centralized Applications
- ~ Ethereum Accounts overview
- ~ Contract Accounts(CA)
- ~ Externally Owned Accounts(EOA)
- ~ Fields in Ethereum accounts
- ~ Ethereum Account messages
- ~ Ethereum Account transactions
- ~ Ethereum Addresses
- ~ Units of Ether
- ~ Ether Gas
- ~ Computing total gas cost for Ethereum transactions
- ~ Ethereum gas price Javascript implementation
- ~ Ethereum as a State Transition Function
- ~ Ethereum Architecture
- ~ Ethereum Virtual Machine(EVM)
- ~ EVM nodes vs mining nodes
- ~ EVM Bytecode
- ~ EVM Instruction Set
- ~ EVM Opcode
- ~ EVM Storage
- ~ EVM Memory
- ~ EVM Stack
- ~ Geth setup and EVM practical
- ~ Converting bytecode to opcode
- ~ Application Binary Interface(ABI)
- ~ Understanding end-to-end Ethereum Blockchain transaction in Javascript
- ~ Ethereum Smart Contracts architecture

=> Ethereum 2.0 :

- ~ Why was Ethereum 2.0 proposed?
- ~ Energy usage in Proof of Work
- ~ Gas costs in Ethereum 1.0
- ~ Potential scalability issues
- ~ Moving from Proof of Work to Proof of Stake
- ~ Proof of Stake in Ethereum 2.0
- ~ Validators
- ~ Staking
- ~ Attestation
- ~ Crosslinks
- ~ Finality
- ~ consensus clients
- ~ Execution clients
- ~ Sharding
- ~ Shard chains
- ~ Beacon chain
- ~ Data rollup in Ethereum 2.0
- ~ Forking in Blockchain
- ~ Hard Fork
- ~ Soft Fork
- ~ The DAO attack and Ethereum Hard Fork

=> Solidity :

- ~ What is Solidity?
- ~ Why should you learn Solidity programming?
- ~ Introduction to Smart Contracts
- ~ Solidity Installation
- ~ Remix IDE
- ~ Installing Solidity in npm / Node.js
- ~ Layout of a Solidity Source File
- ~ SPDX License Identifier
- ~ Pragmas

- ~ *Comments in Solidity*
- ~ *Structure of a Smart Contract*

#### => Solidity Value Types :

- ~ *Solidity datatypes*
- ~ *Booleans*
- ~ *Integers*
- ~ *Address Type*
- ~ *Address Literals*
- ~ *Contract Types*
- ~ *Byte Type*
- ~ *String Types*
- ~ *Enums in Solidity*

#### => Solidity Reference Types :

- ~ *Data locations- storage, memory and callback*
- ~ *Solidity Arrays*
- ~ *Fixed Arrays*
- ~ *Dynamic Arrays*
- ~ *Bytes and Strings as Arrays*
- ~ *Array Slicing*
- ~ *Structs*
- ~ *Mapping Types*

#### => Solidity Units and Global Variables :

- ~ *Ether Units*
- ~ *Time Units*

#### => Solidity Control Structures :

- ~ *If statement*
- ~ *If/else statement*
- ~ *Nested if/else statements*
- ~ *Solidity Loops*
- ~ *For loop*
- ~ *While loop*
- ~ *Do-while loop*
- ~ *Break statement*
- ~ *Continue statement*

#### => ABI Encoding and Decoding Functions :

- ~ *ABI encoder*
- ~ *ABI decoder*

#### => Cryptographic Functions :

- ~ *Keccak256*
- ~ *SHA256*
- ~ *Ripemd160*
- ~ *Ecrecover*

#### => Smart Contracts :

- ~ *Creating Smart Contracts*
- ~ *Constructor*
- ~ *Scope visibility*
- ~ *State variable visibility*
- ~ *Functions*
- ~ *Function visibility*
- ~ *Getter functions*
- ~ *Setter functions*
- ~ *Function modifiers*
- ~ *Return variables and returning multiple values*
- ~ *Immutable state variables*
- ~ *Payable functions*
- ~ *Fallback functions*
- ~ *View functions*
- ~ *Pure functions*
- ~ *Function overloading*
- ~ *Function overriding*
- ~ *Solidity Events*
- ~ *Block and Transaction details*
- ~ *Solidity Inheritance*
- ~ *Single Inheritance*
- ~ *Multiple Inheritance*
- ~ *Heirarchical Inheritance*
- ~ *Multilevel Inheritance*
- ~ *Abstract Contracts*
- ~ *Solidity Interfaces*
- ~ *Solidity Libraries*

#### => Solidity Programming Applications :

- ~ *Ether Wallet*
- ~ *Multi Sig Wallet*
- ~ *Iterable Mapping*
- ~ *ERC20*
- ~ *ERC721*
- ~ *Uni-directional Payment Channel*
- ~ *Bi-directional Payment Channel*
- ~ *NFT Auction*
- ~ *Crowd Fund*
- ~ *Time Lock*

#### => Common Ethereum Blockchain Hacks and Loopholes :

- ~ Re-Entrancy Attack
- ~ Self Destruct
- ~ Accessing Private Data
- ~ Denial of Service
- ~ Phishing with tx.origin
- ~ Hiding Malicious Code with External Contract
- ~ Honeypot
- ~ Front Running
- ~ Block Timestamp Manipulation
- ~ Signature Replay
- ~ Bypass Contract Size Check

=> Introduction to Blockchain Development Frameworks :

- ~ Introduction to Smart Contract Development in Production
- ~ Web3 libraries for Javascript
- ~ Smart Contract development tools
- ~ Web3 Providers
- ~ Wallets

=> Truffle Suite :

- ~ Truffle overview
- ~ Truffle Installation
- ~ Creatin a new project in Truffle
- ~ Exploring project directories in Truffle
- ~ Compiling Smart Contracts
- ~ Building Artifacts
- ~ Handling Dependencies
- ~ Reading and writing Smart Contract data
- ~ Smart Contract Transactions in Truffle
- ~ Function calls in Truffle
- ~ Abstractions
- ~ Executing Contract functions
- ~ Making Transactions
- ~ Processing Transaction results
- ~ Catching events
- ~ Add a new contract to the network
- ~ Sending ether to a contract
- ~ Invoking overloaded methods
- ~ Using enumerations
- ~ Preserving Files and Content to Storage Platforms
- ~ Inter Planetary File System(IPFS)
- ~ Filecoin
- ~ Textile Buckets
- ~ Running Migrations
- ~ Initial Migration
- ~ Truffle Deployer
- ~ Network considerations
- ~ Truffle Deployer API
- ~ Integrating Truffle with Metamask
- ~ Using Truffle Dashboard
- ~ Using truffle Debugger
- ~ Truffle Develop and Truffle Console
- ~ Writing and executing external scripts
- ~ Testing Smart Contracts
- ~ Writing Automated Tests in Javascript
- ~ Writing Automated Tests in Solidity
- ~ Truffle Build Process
- ~ Truffle Boxes
- ~ Ethereum Name Service
- ~ Truffle Event System
- ~ Network Configuration and Dapp Deployment
- ~ Ganache- Ethereum Client for Truffle Suite
- ~ Installing Ganache
- ~ Ganache Workspaces
- ~ Ganache Ethereum Workspace
- ~ Understanding Workspace Default Configuration in Ganache
- ~ Managing Ganache configurations and settings
- ~ Configuring Truffle to connect to Ganache
- ~ Managing Truffle projects in Ganache
- ~ Exploring the Contracts page
- ~ Exploring the Transactions page
- ~ Linking and unlinking a Truffle project
- ~ Ganache Workspaces
- ~ Creating Workspaces
- ~ Deleting Workspaces
- ~ Editing Workspaces
- ~ Ethereum Workspace
- ~ Loading Existing Workspaces
- ~ Switching Workspaces

=> Hardhat :

- ~ Introduction To Hardhat - Ethereum development environment for professionals
- ~ Hardhat Installation
- ~ Creating a Hardhat project
- ~ Configuring Ethereum Networks
- ~ Configuring the compiler
- ~ Compiling your contracts
- ~ Artifacts
- ~ Writing deployment scripts

- ~ Deploying the Contracts
- ~ Testing Smart Contracts
- ~ Running tests with Ganache
- ~ Running tests on Visual Studio Code
- ~ Running multiple tests in parallel
- ~ Running tasks
- ~ Hardhat Console
- ~ Creating custom tasks
- ~ Hardhat Runtime Environment(HRE)
- ~ Hardhat Plugins
- ~ Optimizing Plugins
- ~ Verbose Logging for debugging
- ~ Solutions to common runtime problems

=> Web3.js :

- ~ Introduction to Web3.js
- ~ Why should you learn Web3.js?
- ~ Applications of Web3.js
- ~ Installing Web3.js using NPM
- ~ Web3 modules
- ~ Creating a new Web3 instance
- ~ Introduction to Web3 Providers
- ~ Setting up a Web3 Provider
- ~ Batch request
- ~ Extending Web3 modules
- ~ Introduction to Web3.eth
- ~ Checksum addresses overview
- ~ Fetching default blockchain details
- ~ Transaction methods
- ~ Block Node methods
- ~ Subscriber Methods
- ~ Web3.js Smart Contract objects and methods
- ~ User wallet and account methods
- ~ Interacting with Ethereum node accounts using web3.eth.personal
- ~ Working with ABI in web3.js
- ~ Commonly used utilities in web3.js
- ~ Hardhat automated testing with Web3.js and Truffle

=> Ethers.js :

- ~ What is Ethers?
- ~ Ethers.js Features
- ~ Installing Ethers.js using NPM
- ~ Connecting to Ethereum: MetaMask
- ~ Connecting to Ethereum: RPC
- ~ Building blocks of Ethers.js- Signers,Providers and Contracts

=> Ethers.js Providers :

- ~ What are Providers?
- ~ Ethers.js provider API overview
- ~ Provider Account methods
- ~ Blocks Methods
- ~ Ethereum Naming Service (ENS) Methods
- ~ EnsResolver
- ~ Logs Methods
- ~ Network Status Methods
- ~ Transactions Methods
- ~ Event Emitter Methods
- ~ Inspection Methods
- ~ BaseProvider
- ~ JsonRpcProvider
- ~ JsonRpcSigner
- ~ JsonRpcUncheckedSigner
- ~ Static.JsonRpcProvider
- ~ Node-Specific Methods
- ~ API Providers
- ~ EtherscanProvider
- ~ InfuraProvider
- ~ AlchemyProvider
- ~ CloudflareProvider
- ~ PocketProvider
- ~ AnkrProvider
- ~ Other Providers
- ~ FallbackProvider
- ~ IpcProvider
- ~ JsonRpcBatchProvider
- ~ UrlJsonRpcProvider
- ~ Web3Provider
- ~ WebSocketProvider

=> Smart Contract Interaction :

- ~ Creating new Smart Contract instance
- ~ Contract Properties
- ~ Contract Methods
- ~ Events
- ~ ContractFactory
- ~ Creating ContractFactory Instances
- ~ ContractFactory Interface Properties
- ~ ContractFactory Methods
- ~ Meta-Class
- ~ Deploying a Contract

- ~ *Connecting to a Contract*

## => Ethereum Blockchain Projects :

- ~ *Building cryptocurrency with ICO*
- ~ *Building decentralized ecommerce website*
- ~ *Building decentralized voting application*
- ~ *Decentralized music sharing app*
- ~ *Token contract swap application*
- ~ *Full stack email dapp*

## => Oracles :

- ~ *What is a Blockchain Oracle?*
- ~ *Solving the Oracle problem*
- ~ *Decentralized Oracles*
- ~ *Types of Blockchain Oracles*
- ~ *Applications of Blockchain oracles*

## => Chainlink overview :

- ~ *Introduction to Chainlink*
- ~ *Understanding the Chainlink Ecosystem*
- ~ *Chainlink Features*
- ~ *Chainlink Applications as Decentralized Oracles*

## => Data Feeds :

- ~ *Introduction to Data Feeds*
- ~ *Using Data Feeds*
- ~ *Fetchin Historical Cryptocurrency Price Data*
- ~ *Chainlink Feed Registry*
- ~ *Using ENS with Data Feeds*

## => Custom Data Feeds :

- ~ *Using any API*
- ~ *Make a GET Request*
- ~ *Multi-Variable Responses*
- ~ *Large Responses*
- ~ *Make an Existing Job Request*
- ~ *Find Existing Jobs*
- ~ *Contract Addresses*

## => Oracle Projects :

- ~ *Live cryptocurrency trading using chainlink*
- ~ *Insurance Dapp using chainlink*

## => The Graph :

- ~ *The Graph Protocol*
- ~ *The Graph architecture*
- ~ *Edge and Node*
- ~ *Everest Registry*
- ~ *Graph Protocol*
- ~ *The Graph vs Etherscan*
- ~ *Graph-cli Installation*
- ~ *Creating new subgraphs*
- ~ *Writing subgraphs*
- ~ *Publishing a Subgraph to the Decentralized Network*

## => GraphQL API :

- ~ *Queries*
- ~ *Sorting*
- ~ *Pagination*
- ~ *Filtering*
- ~ *Time-travel queries*
- ~ *Fulltext Search Queries*
- ~ *Validation*
- ~ *Schema*
- ~ *Entities*
- ~ *Signalling*
- ~ *Curation*
- ~ *Delegators*
- ~ *Consumers*
- ~ *Deploying subgraphs*
- ~ *Subgraph logging*
- ~ *Graph protocol testnet using docker compose*
- ~ *Ethereum node monitoring using The Graph, Prometheus and Grafana*

## => The Graph Networking :

- ~ *Introduction to indexers*
- ~ *Revenue streams*
- ~ *Distribution*
- ~ *Allocation life cycles*
- ~ *Querying and indexing subgraphs*
- ~ *IPFS Hash convertor*

## => AssemblyScript API for The Graph :

- ~ *Installing AssemblyScript API*
- ~ *API Reference*
- ~ *Versions*
- ~ *Built-in Types*
- ~ *Store API*
- ~ *Ethereum API*
- ~ *Logging API*
- ~ *IPFS API*
- ~ *Crypto API*



- ~ JSON API
- ~ Type Conversions Reference
- ~ Data Source Metadata
- ~ Entity and DataSourceContext

=> Project :

- ~ Building a Full-stack Blockchain Application using Ethereum, Polygon, Next.js and GraphQL

=> Decentralized Autonomous Organisations(DAO) :

- ~ What are DAOs?
- ~ Why do we need DAOs?
- ~ DAO membership
- ~ Token-based membership
- ~ Share-based membership
- ~ How do DAOs work?
- ~ Properties of DAOs
- ~ Ethereum and DAOs
- ~ Understanding Governance Mechanisms
- ~ DAOs and the principal-agent problem
- ~ Building Decentralized Autonomous Organisations
- ~ Defining the DAO purpose
- ~ Building the DAO voting mechanism
- ~ Creating the governance token
- ~ DAO fund management
- ~ Initial Coin Offering (ICO)
- ~ Creating a DAO on Aragon
- ~ Creating a DAO using Snapshot
- ~ Building a DAO using DAOstack Alchemy

=> Creating a Custom DAO Project :

- ~ Understanding custom DAOs
- ~ Finding the purpose for our Custom DAO
- ~ Designing the voting architecture
- ~ Implementing the voting architecture in Solidity
- ~ Designing the components of the governance token(DAO cryptocurrency)
- ~ Creating the governance token in Solidity
- ~ Fund Management for our custom DAO
- ~ Designing the Multi-signature wallet for Fund Management
- ~ Creating the Multi-signature wallet in Solidity
- ~ Testing DAO Smart Contracts
- ~ Deploying the DAO to testnet

=> NFT Platforms :

- ~ What are NFT Platforms/Marketplaces?
- ~ CryptoKitties
- ~ Opensea
- ~ Rarible
- ~ Decentraland
- ~ Binance NFT
- ~ Enjin Marketplace
- ~ Axie Marketplace
- ~ Foundation
- ~ Nifty Gateway
- ~ Mintable
- ~ Theta Drop

=> NFT Transaction Fees :

- ~ Gas Fees in NFT
- ~ What are one-time Gas Fees NFT?
- ~ Recurring Gas Fees
- ~ Actions in Gas Fees
- ~ Check Ethereum Gas Fee
- ~ Create and Sell NFTs without Gas Fees
- ~ NFT Marketplace Fees

=> NFT project :

- ~ Building a complete NFT Marketplace with User Interface

=> Polygon Blockchain(MATIC) :

- ~ Introduction to Polygon Blockchain
- ~ Why should you use Polygon network?
- ~ Layer 1 vs Layer 2 Blockchains
- ~ Features of Polygon Blockchain
- ~ Polygon Architecture
- ~ Zero-Knowledge cryptography
- ~ Zero-Knowledge rollups

=> Polygon Projects :

- ~ Retail supply chain Application using Polygon Network
- ~ Building a Social media Dapp on Polygon

=> Polkadot :

- ~ Polkadot Overview
- ~ Polkadot Whitepaper
- ~ Polkadot Architecture
- ~ Parachains
- ~ Parathreads
- ~ Substrate Installation

=> Substrate Fundamentals :

- ~ Runtime environment and setup
- ~ Exinsics

- ~ *Account Abstractions*
- ~ *Transaction Pool*
- ~ *Session Keys*
- ~ *Transaction Weight*
- ~ *Execution*
- ~ *Off-Chain Features*

=> Runtime Development :

- ~ *Frames*
- ~ *Macros*
- ~ *Metadata*
- ~ *Storage*
- ~ *Origins*
- ~ *Events and Errors*
- ~ *Weights and Fees*
- ~ *Benchmarking*
- ~ *Debugging*
- ~ *Testing*
- ~ *Randomness*
- ~ *Chain Specification*
- ~ *Upgrades*
- ~ *Pallet Coupling*
- ~ *Custom RPCs*
- ~ *Smart Contract Toolkits*

=> Development Integration :

- ~ *Polkadot-JS*
- ~ *Client Libraries*
- ~ *Substrate Connect*

=> Development Tools :

- ~ *SR tool*
- ~ *Subxt*
- ~ *Tx Wrapper*
- ~ *Sub Flood*
- ~ *Substrate Archive*
- ~ *Sidecar*
- ~ *Polkadot Launch*

=> Advanced topics in Polkadot :

- ~ *Account Info*
- ~ *SCALE Codec for Substrate*
- ~ *Consensus*
- ~ *Block Import*
- ~ *Executor*
- ~ *Cryptography*
- ~ *Storage*
- ~ *SS58 Address Format*
- ~ *Hash Collections*

# AI Operations

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : <https://ineuron.ai/course/AI-Operations>

## Course Description :-

Artificial Intelligence Operations (AIOps) is the most in demand technical skill these days. It helps to incorporate DevOps principle in AI product development. It's a live instructor-led certification program provided by iNeuron intelligence private limited. Here you will learn various methods to implement AIOps methodology in the ML and DL projects which includes implementation on various clouds like AWS, Azure, GCP and DigitalOcean.

## Course Features :-

- => AIOps certification
- => Online Instructor-led learning: Live teaching by instructors
- => Hands-on project implementation
- => 120+ hours of live interactive classes
- => Every week doubt clearing session after the live classes
- => Lifetime Dashboard access
- => Doubt clearing session
- => Doubt clearing through e-mail
- => Assignments in all the module
- => Resume building
- => Career guidance
- => Interview Preparation
- => Regular assessment
- => Live class recordings and materials
- => Interview Questions

## What you will learn :-

- => AIOps
- => Linux foundation
- => GIT foundation
- => GitHub
- => Gitlab
- => Data version control DVC
- => MLFlow
- => Docker foundation
- => Kubernetes Foundation
- => Tensorflow Extend (TFX)
- => Kubeflow
- => AWS AIOps
- => Azure AIOps
- => GCP AIOps
- => Digital Ocean

## Requirements :-

- => Minimum System requirement: Intel Core i3 processor and 4GB RAM or higher
- => Decent internet connection
- => Your Dedication

## Instructors :-

- => Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

=> Introduction to AI Ops

=> Linux Foundation :

- ~ Why Linux? Linux types? How to access Linux env in different system
- ~ Installation of virtual box, WSL, sandbox for windows user
- ~ Free tier EC2 ubuntu instance
- ~ SSH and SSH tools
- ~ Putty
- ~ Filezilla
- ~ WinSCP
- ~ Course Introduction
- ~ Working with the Shell - I
- ~ Introduction to Shell
- ~ Basic Linux Commands: ls, cat, cd, rm, chmod...etc
- ~ Help for command line
- ~ Type of Shell: bash, zsh etc
- ~ Bash Shell
- ~ Linux Core Concepts
- ~ Linux Kernel and types
- ~ Linux file system
- ~ Linux Boot Sequence
- ~ Runlevels
- ~ File Types
- ~ Filesystem Hierarchy
- ~ Package Management
- ~ Package Management Introduction and configuration
- ~ Linux type based package manager
- ~ RPM and YUM
- ~ DPKG and APT
- ~ Working with the Shell - II
- ~ File Compression and Archival
- ~ Searching for Files and Patterns using grep/wildcards etc
- ~ VI, Nano Editor
- ~ Security and File Permissions
- ~ The Security Incident (story)
- ~ Linux Accounts
- ~ User Management
- ~ Access Control Files
- ~ Account Management
- ~ File Permissions and Ownership
- ~ Cronjobs
- ~ Service management with systemd
- ~ Working overtime (story)
- ~ Creating a systemd Service
- ~ systemd Tools
- ~ Lab - systemd services

=> GIT Foundation :

- ~ What? Why? When? Type? Vendor? Pricing? Industry wise uses of GIT
- ~ Creation of Github/Gitlab/bitbucket account
- ~ Local GitHub UI installation, setup with VSCode and Pycharm
- ~ Local and Remote Repositories installation and configuration
- ~ GIT Repository initialization
- ~ command: git log
- ~ Git Branches
- ~ What is branching in Git and why we need it?
- ~ Master/main branch and user-defined branch
- ~ Checkout and pushing to a branch
- ~ Merging of branches
- ~ Project control and management
- ~ In Remote Repositories
- ~ Initialization of Remote Repositories
- ~ Pushing code to the remote repositories
- ~ Cloning of the remote repositories to local
- ~ PR (Pull Requests)
- ~ Fetch and Pull
- ~ Handling conflict on merging branch
- ~ Forking of repository
- ~ Rebasing
- ~ Resetting and Reverting
- ~ Stashing

=> Data Version Control (DVC) :

- ~ DVC
- ~ What is DVC?
- ~ Installation
- ~ Mac OS
- ~ Windows
- ~ Linux
- ~ Get Started
- ~ Data Versioning
- ~ Model Versioning
- ~ Data Access
- ~ Model Access
- ~ Data Pipelines
- ~ Metrics, Parameters, Plots
- ~ Run, Queue, Compare, Persisting, and Sharing Experiments

- ~ Clean up
- ~ DVC Uses
- ~ Versioning Data and Models
- ~ Sharing Data and Model Files
- ~ Data Registries
- ~ Shared Development Server
- ~ Project Structure
- ~ Experiment Management
- ~ Setup Google Drive Remote
- ~ Large Dataset Optimization
- ~ External Dependencies
- ~ Managing External Data
- ~ Automate Pipelines with DVC
- ~ Pipelines & Experiment Automation
- ~ Common issues with ML experiments
- ~ Build automated pipelines
- ~ Build automated pipeline
- ~ Experiments Management
- ~ Experimenting with reproducible pipelines
- ~ Tracking metrics and plots
- ~ Compare experiment results
- ~ Build, Test & Deploy
- ~ Introduction to CI/CD in Machine Learning
- ~ Build CI/CD pipeline
- ~ Install GitLab Runner and Trigger CI/CD pipeline
- ~ Build Machine Learning pipeline
- ~ Build CI/CD pipeline
- ~ Trigger CI/CD pipeline
- ~ Making Continuous Integration work with ML
- ~ DVC Integration with Project
- ~ Build a model Prototype
- ~ Build a prototype with Jupyter Notebook
- ~ Start to version your code with Git
- ~ Version your code with Git
- ~ Create pipelines
- ~ Automate pipelines and data versioning with DVC
- ~ Create CI pipeline to build, test, experiment
- ~ Experimenting with DVC and CML
- ~ Deploy your model

=> MLFlow :

- ~ What is MLFlow?
- ~ Installation
- ~ MLflow Tracking
- ~ Where Runs Are Recorded
- ~ How Runs and Artifacts are Recorded
- ~ Scenario 1: MLFlow on localhost
- ~ Scenario 2: MLFlow on localhost with SQLite
- ~ Scenario 3: MLFlow on localhost with Tracking Server
- ~ Scenario 4: MLFlow with remote Tracking Server, backend and artifact stores
- ~ Logging Data to Runs
- ~ Logging Functions
- ~ Launching Multiple Runs in One Program
- ~ Performance Tracking with Metrics
- ~ Visualizing Metrics
- ~ Automatic Logging
- ~ Scikit-learn
- ~ TensorFlow and Keras
- ~ Gluon
- ~ XGBoost
- ~ Pytorch
- ~ MLFlow Tracker
- ~ Organizing Runs in Experiments
- ~ Managing Experiments and Runs with the Tracking Service API
- ~ Tracking UI
- ~ Querying Runs Programmatically
- ~ MLFlow Tracking Servers
- ~ Storage
- ~ Networking
- ~ Logging to a Tracking Server
- ~ MLflow Projects
- ~ Overview
- ~ Specifying Projects
- ~ Running Projects
- ~ Iterating Quickly
- ~ Building Multi Step Workflows
- ~ MLFlow Models
- ~ Storage Format
- ~ Model Signature And Input Example
- ~ Model API
- ~ Built-In Model Flavors
- ~ Model Customization
- ~ Built-In Deployment Tools
- ~ Deployment to Custom Targets
- ~ Model Registry
- ~ Model Registry Workflows
- ~ UI Workflow
- ~ Registering a Model
- ~ Using the Model Registry

- ~ API Workflow
- ~ Adding an MLFlow Model to the Model Registry
- ~ Fetching an MLFlow Model from the Model Registry
- ~ Serving an MLFlow Model from Model Registry
- ~ Adding or Updating an MLFlow Model Descriptions
- ~ Renaming an MLFlow Model
- ~ Transitioning an MLFlow Models Stage
- ~ Listing and Searching MLFlow Models
- ~ Archiving an MLFlow Model
- ~ Deleting MLFlow Models

=> Docker Foundation :

- ~ Setup
- ~ Why? What? Where? Problem it can solve? Docker types? Cloud based docker containers
- ~ Installation of specific docker editions based on your system
- ~ Installing Docker
- ~ Create and Use
- ~ Docker Install, Configuration and verify
- ~ Container VS
- ~ Windows Containers unlike Linux
- ~ Inside Containers - Process Monitoring with Command Line Interface(CLI)
- ~ Private and Public Communication in Containers
- ~ CLI Management of Virtual Networks
- ~ Domain Name System(DNS) for Containers can find each other
- ~ Containers
- ~ Docker Image
- ~ Docker Hub Registry predefined Images
- ~ Images and Their Layers: Discover the Image Cache
- ~ Image Tagging and Pushing to Docker Hub
- ~ Create images
- ~ Using Dockerfile Basics
- ~ Run Docker Builds
- ~ Extend Official Images
- ~ Container Lifetime & Persistent Data
- ~ Persistent Data: Data Volumes
- ~ Shell Differences for Path Expansion
- ~ Persistent Data: Bind Mounting
- ~ Docker Compose
- ~ What is Docker Compose ?
- ~ Docker-compose.yml
- ~ Compose Commands
- ~ Add Image Building to Compose Files
- ~ docker project: Deploy ML model and services using Docker

=> Kubernetes Foundation

=> TFX

=> Kubeflow :

- ~ What is Kubeflow?
- ~ Core Kubeflow components
- ~ How to set up Kubeflow on Kubernetes
- ~ How to develop basic ML models in Kubeflow Notebooks
- ~ How to train and deploy models in Kubeflow
- ~ How to use Kubeflow Pipelines
- ~ How to use KFServing to deploy models
- ~ How to manage logs with Kubeflow Metadata component
- ~ Katib Hyper Parameter Tuning
- ~ Kubeflow Pipelines to KFServing

=> GitLab Foundation :

- ~ GitLab Triggers
- ~ AWS S3 storage
- ~ GitLab CI/CD Pipelines
- ~ Pipelines definition
- ~ MongoDB cloud Atlas
- ~ Heroku
- ~ Logdata
- ~ Coral for Monitoring

=> AWS MLOps :

- ~ Amazon Sagemaker
- ~ Amazon s3
- ~ AWS Codebuild
- ~ AWS Codecommit
- ~ Sagemaker Training Job
- ~ Sage Maker Endpoint
- ~ Amazon Api Gateway
- ~ Sagemake Model Monitoring
- ~ Cloudwatch Synthetics
- ~ Cloudwatch Alarm

=> Azure MLOps :

- ~ Create an Azure Machine Learning workspace
- ~ Setup a new project in Azure DevOps
- ~ Import existing YAML pipeline to Azure DevOps
- ~ Declare variables for CI/CD pipeline
- ~ Create training compute
- ~ Train ML model
- ~ Register model
- ~ Deploy model in AKS

=> GCP MLOps :

- ~ *Creating Flask application using Python*
- ~ *Best practices building Flask App*
- ~ *Understanding Docker files and Dependencies*
- ~ *Creating container image*
- ~ *Walkthrough of different deployment options*
- ~ *Serverless deep dive*
- ~ *Deploying on GCP App Engine*
- ~ *Deploying on Serverless Framework*
- ~ *Hosted Kubeflow Pipelines*
- ~ *Start Hosted Pipelines*
- ~ *cluster permissions*
- ~ *Development environment*
- ~ *Launch AI Platform notebook*
- ~ *CI/CD Production Environment*
- ~ *Set up Continuous Integration (CI)*
- ~ *Verify CD*

=> Digital Ocean :

- ~ *Droplets*
- ~ *File Transfers*
- ~ *Gitops*
- ~ *Jenkins*
- ~ *Creating Jobs*
- ~ *Creating pipelines in Jenkins*
- ~ *Docker Images*
- ~ *Kubernetes Flow*
- ~ *Creating Clusters*
- ~ *Load testing*

# Power BI Foundations

---

Topic Name : DATA ANALYTICS

Sub-topic Name : POWER BI

Course link : <https://ineuron.ai/course/Power-BI-Foundations>

## Course Description :-

Power BI is a luxury tool in the hands of businesses overwhelmed by the amount of data they have on hand, and we don't have any other cost-effective way to pull insights than it until now. As a result, power BI swiftly establishes itself as the world's most powerful self-service business intelligence platform and an indispensable tool for both data pros and beginners.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Creating Reports
- => Visualization
- => Real-time insights
- => Dashboarding
- => Business intelligence workflow

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jayant Topnani :
  - ~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

- => Introduction :
  - ~ Difference between Business Analyst & a data analyst
  - ~ Business Understanding & Data Understanding
  - ~ Data Analysis & Data Visualization
  - ~ Data Cleaning & Preparation & methods used with some examples
  - ~ Tools for Data Analysis & Visualization
  - ~ Some charts & best practice related to them
  - ~ Some data connectors in Power BI (Excel/txt, csv)
  - ~ Basic elements of power BI their differences & use
  - ~ Ways of PBI desktop App installation & system requirements
  - ~ PBI desktop App interface Explanation
- => Data Modeling :
  - ~ Modeling Basics
  - ~ Creating Relationships
  - ~ Normalization-Denormalization
  - ~ Dimension & Fact Tables
  - ~ Relationships (Autodetect, Manual & Autodetection settings)
  - ~ Cardinality
  - ~ Active & Inactive Relationships
- => Power Query Editor :
  - ~ Some Basic Data Cleaning Operations
  - ~ Data Transformation
  - ~ Merge & Append Queries
  - ~ Interview Questions related to query Editor
- => DAX :
  - ~ DAX Basics
  - ~ Row & Filter Context
  - ~ Measures & Calculated Columns
  - ~ Some DAX Functions



=> Reporting & Dashboarding :

- ~ Various Types of Filters
- ~ Various Types of Visualizations & Formatting Options
- ~ Story telling & Dashboarding

=> Some More Connectors :

- ~ SQL server
- ~ Odata feed
- ~ Blank Query
- ~ MS Access
- ~ Real Time Data Streaming
- ~ Web
- ~ Pdf
- ~ Folder
- ~ OLE DB

=> PBI service, Mobile App :

- ~ Login PBI service
- ~ User Interface PBI service
- ~ Natural Language query & Quick Insights feature
- ~ Visual Interactions
- ~ Publishing a report to PBI service
- ~ Sharing reports with others, PBI service
- ~ How can we access reports & dashboards through PBI mobile App ?
- ~ Schedule a refresh

# Data Science Industry Ready Projects

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/Data-Science-Industry-Ready-Projects>

## Course Description :-

Ready to use end-to-end data science projects for real-world business use cases. We will be discussing projects from very scratch such as understanding problem statements, capturing requirements, and various aspects of project design using different documentation such as High-Level Design, Low-Level Design, and Architecture Design. Practical use of MLOPS practices using tools such as MLFLOW, Wandb. Pipeline implementation for training, retraining, and inferencing. Designing dashboard to present important KPIs to monitor system and model performance and generate alert to notify the appropriate parties to address serious problems if it is about to occur.

## Course Features :-

- => Online Instructor-led learning
- => Doubt Clearing
- => Proper Roadmap for building AI projects
- => Lifetime Dashboard access
- => Recording of Live Class
- => Material
- => Interview Questions
- => Resume Building
- => Career Guidance
- => Quiz in every module - Based on Real Time Questions
- => Certificate
- => Industry Level Projects and Case studies
- => Capstone Projects

## What you will learn :-

- => System Architecture
- => High Level Design
- => Component Selection
- => Low Level Design
- => Core utility design
- => Deployment Architecture
- => Multistage pipeline for CI/CD
- => ML Pipeline Understanding
- => Training Pipeline Implementation
- => Inference Pipeline Implementation
- => Retraining Pipeline Implementation
- => Deployment of ML Pipeline on Cloud
- => Monitoring of System and Model Performance

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication
- => Knowledge of Python
- => Knowledge of Machine Learning

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Avnish Yadav :

~ 3+ years of experience in various domains such as data scientist, data analyst, database developer, and .net developer. Implemented various sophisticated business requirements, performed an analysis of various data to capture insights and hidden patterns. Fine and tuned various regression and classification-based algorithms for prediction. Implemented various ETL pipelines to fulfil the business requirement. Automated various machine learning pipelines such as data loading, data cleaning, data validation, model selection, model tuning, and model monitoring pipeline. Implemented machine learning pipeline in azure machine learning studio. I have a keen interest to solve complicated machine learning problems to fulfil business requirements.

=> Ketan Gangal :

~ I have worked in data science for more than two years, and I have a track record of successfully implementing data science pipelines in production with practical expertise using ML-Ops, deep learning & machine learning. I also love sequence processing because it is deeply inspired by humans as our feeling, thoughts, emotions, sensations, language are sequential in nature if we can enable machine to understand sequence of information and act accordingly we can make significant progress towards true artificial intelligence.

## Curriculum details :-

=> Project - Sensor Fault Detection :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pandas, Sklearn, Mlflow, Cloud, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Customize Model Training
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation
- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud
- ~ Schedule and Orchestrate Training Pipeline
- ~ Deployment of Inference Pipeline as an API
- ~ Deployment of Retraining Pipeline
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

=> Project - Financial Product Complaint :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pytorch, Cloud, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation

- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud
- ~ Schedule and Orchestrate Training Pipeline
- ~ Deployment of Inference Pipeline as an API on Elastic Container Serving
- ~ Deployment of Retraining Pipeline
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

#### => Project - Face Authenticator :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, MongoDB, Deepface, Flask, Docker, EC2 Instance, Git, Github, SQL
- ~ Face Authenticator Pipeline
- ~ Understanding Face Authenticator mechanism
- ~ Face Registration Pipeline
- ~ Face Identification Pipeline
- ~ Face Registration Pipeline
- ~ Capturing Images of a Person
- ~ Generating Embedding of Facial Image
- ~ Save Embedding in Database
- ~ Face Identification Pipeline
- ~ Detecting face of a Person at login portal
- ~ Generate embedding of captured face
- ~ Search Generated Embedding in DB using similarity metrics Triplet Loss
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

#### => Project - Embedding based search engine :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pytorch, Hugging Face, Transformer, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation
- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud

- ~ Schedule and Orchestrate Training Pipeline
- ~ Deployment of Inference Pipeline as an API
- ~ Deployment of Retraining Pipeline
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

=> Project - AI Based Hybrid Recommender System :

- ~ Project Introduction
- ~ Project Business Use case
- ~ System Architecture
- ~ High Level Design
- ~ Component Selection
- ~ Low Level Design
- ~ Core utility design
- ~ Deployment Architecture
- ~ Multistage pipeline for CI/CD
- ~ Technology Stack
- ~ Python, Pytorch, Transformer, Prometheus and Grafana, Docker, RDBMS, Cloud Storage, Flask, Git, GitHub, Microsoft library
- ~ ML Pipeline Understanding
- ~ Type of ML Pipeline
- ~ Training Pipeline
- ~ Inferencing Pipeline
- ~ Retraining Pipeline
- ~ Training Pipeline Implementation
- ~ Introduction to Training Pipeline
- ~ Data Ingestion From Data Source
- ~ Data Validation
- ~ EDA, Data Preprocessing, Feature Engineering Model Selection
- ~ Model Training, Selection and Hyperparameter Tuning
- ~ Model Analysis and Evaluation
- ~ Model Push/ Export
- ~ Inference Pipeline Implementation
- ~ Introduction to Inference Pipeline
- ~ Understanding of the use of Artifact Generated by Training Pipeline
- ~ Data Validation
- ~ Data Preprocessing and Feature Engineering
- ~ Prediction using preprocessed data
- ~ Retraining Pipeline Implementation
- ~ Introduction to Retraining Pipeline
- ~ Model Analysis and Performance Monitoring of Prediction Pipeline
- ~ Creating Trigger to Initiate Model Retraining
- ~ Deployment of ML Pipeline on Cloud
- ~ Schedule and Orchestrate Training Pipeline
- ~ Deployment of Inference Pipeline as an API
- ~ Deployment of Retraining Pipeline
- ~ Monitoring of System and Model Performance
- ~ Importance of Monitoring
- ~ Visualization of KPI and Other Indicator
- ~ System and Model Performance Visualization
- ~ Implementation of Alert and Notification to prevent Failure
- ~ Project Conclusion

# Solidity

---

Topic Name : BLOCKCHAIN

Sub-topic Name : SOLIDITY

Course link : <https://ineuron.ai/course/Solidity>

## Course Description :-

Solidity & Solana Blockchain course is designed to provide an in depth knowledge on various aspects & concepts of blockchain & Solidity. A step by step learning will be help to focus on each & every parameter of Blockchain. This course will take you into a deep dive into the state of the art blockchain technology and how to go about writing smart contracts in the ethereal platform. Moreover, this is a project-ready course which will help you take whatever you learn and apply it into a real-world portfolio-ready app, which you can showcase to the world.

## Course Features :-

- => Understand the why engineers would want to create an app with Ethereum
- => Build compelling blockchain applications using the Ethereum Blockchain
- => Design, test, and deploy secure Smart Contracts
- => Learn the true purpose and capabilities of Ethereum and Solidity
- => Use the latest version of Ethereum development tools (Web3 v1.0)
- => See practical examples to comprehend what the smart contracts are
- => Learn more about solana Blockchain
- => Learn more about IPFS, NFT's, Oracles and DeFI

## What you will learn :-

- => Solidity Fundamentals
- => Smart Contracts in Solidity
- => Data types and Variables
- => Functions
- => Storage vs Memory
- => Events and logs
- => Factory contract
- => Inheritance
- => Inline Assembly
- => Application Binary Interface
- => Smart Contracts Pitfalls , Testing and Debugging
- => Testing smart Contracts
- => Unit tests
- => Integration Tests
- => Javascript tests
- => Smart Contract Best Practices
- => Creating our own cryptocurrency on Ethereum Network
- => What are ICO and what are tokens
- => Understanding about ERC-20
- => Writing code for our cryptocurrency
- => Safe Math
- => Creating the cryptocurrency
- => Deploying it to the network
- => Solana Blockchain
- => Introduction to solana Blockchain
- => Creating our own cryptocurrency on the Solana Network
- => Understanding about Sol-Tokens
- => Writing code for our cryptocurrency
- => Deploying the currency to the network.

- => Creating our own Solana Token using CLI
- => Installing Virtual Box Ubuntu
- => Downloading the Solana CLI
- => Creating the tokens
- => Web 3.0 & Connecting everything into a project
- => What is Web 3.0 ?
- => iNeuron Marketplace course
- => Installing npm, git and node.js
- => Basic project setup
- => Creating the marketplace contract
- => Adding courses
- => Buying courses
- => Building out the front end
- => Building out the front end part 2
- => Deploying the project
- => A little more about ethereum
- => Ethereum naming service
- => Intro to IPFS
- => Oracles
- => DeFi
- => NFTs
- => What are NFTs and ERC721
- => Creating our project and installing in dependencies
- => Creating the contract
- => Creating the scripts
- => iPFS
- => Deploying the NFT to the network.

#### Requirements :-

- => A computer/laptop
- => Good internet connection
- => Will to learn
- => Beginner level understanding about Javascript, Nodejs and React

#### Curriculum details :-

- => Introduction :
  - ~ *Introduction to course Preview*
- => Solidity Fundamentals :
  - ~ *Smart Contracts in Solidity Preview*
  - ~ *Basic-Smart-Contract-Part1*
  - ~ *Basic Smart Contract Part 2*
  - ~ *Data types and Variables - part 1*
  - ~ *Data types and Variables - part 2*
  - ~ *Functions*
  - ~ *Storage vs Memory*
  - ~ *Events and logs*
  - ~ *Factory contract*
  - ~ *Security Of Smart Contracts*
  - ~ *Inheritance*
  - ~ *Inline Assembly*
  - ~ *Application Binary Interface*
- => Smart Contracts Pitfalls, Testing and Debugging :
  - ~ *Unit tests*
  - ~ *Integration Tests*
  - ~ *Javascript tests*
  - ~ *Smart Contract Best Practices*
- => Creating our own cryptocurrency on Ethereum Network :
  - ~ *What are ICO and what are tokens Preview*
  - ~ *Understanding about ERC-20*
  - ~ *Writing code for our cryptocurrency*
  - ~ *Safe Math*
  - ~ *Creating the cryptocurrency*
  - ~ *Deploying it to the network*
- => Solana Blockchain :

- ~ *Introduction to solana Blockchain Preview*
- ~ *Creating our own cryptocurrency on the Solana Network using CLI - part 1*
- ~ *Creating our own cryptocurrency on the Solana Network using CLI - part 2*
- ~ *Creating our own cryptocurrency on the Solana Network using CLI - part 3*
- ~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 1*
- ~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 2*
- ~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 3*
- ~ *Creating our own cryptocurrency on the Solana Network using Javascript - part 4*

=> Web 3.0 & Connecting everything into a project :

- ~ *What is Web 3.0 ?*
- ~ *iNeuron Marketplace - part1*
- ~ *iNeuron Marketplace - part2*
- ~ *iNeuron Marketplace - part3*
- ~ *iNeuron Marketplace - part4*
- ~ *iNeuron Marketplace - part5*
- ~ *iNeuron Marketplace - part6*
- ~ *iNeuron Marketplace - part7*
- ~ *iNeuron Marketplace - part8*

=> A little more about ethereum :

- ~ *Ethereum naming service*
- ~ *Intro to IPFS*
- ~ *Oracles*
- ~ *DeFI*

=> NFTs :

- ~ *What are NFTs and ERC721*
- ~ *Create Your own NFT part 1*
- ~ *Create Your own NFT part 2*
- ~ *Create Your own NFT part 3*
- ~ *Create Your own NFT part 4*



# Pro Computer Networking

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Computer-Networking>

## Course Description :-

This course is designed mostly for computer science subject Computer Networking test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Computer Networking Theoretical Test

=> Computer Networking Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Computer Networking Test :

~ *Computer Networking Test 1*

~ *Computer Networking Test 2*

~ *Computer Networking Test 3*

~ *Computer Networking Test 4*

# Computer Vision

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/Computer-Vision>

## Course Description :-

Computer Vision is a field of Artificial Intelligence (AI) that enables computers and systems to derive meaningful information from digital images, videos and other visual inputs and take actions or make recommendations based on that information. If AI enables computers to think, Computer vision allows them to see, observe and understand. Computer vision is used in industries ranging from energy and utilities to manufacturing and automotive, and the market is continuing to grow.

## Course Features :-

- => Roadmap
- => Challenges
- => Downloadable resources
- => Quizzes
- => Completion Certificate
- => Hands-on Practicals

## What you will learn :-

- => Basic fundamentals of computer vision.
- => CNN architectures, classification, object detection & segmentation
- => Image formation, motion estimation, tracking & other fundamentals with computer vision.
- => Various architecture usages with Computer vision for advanced-level works.

## Requirements :-

- => Python programming is needed
- => A system with a stable internet connection.
- => Your dedication.

## Instructors :-

- => Sourangshu Pal :
  - ~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

### => Introduction to Course :

- ~ Introduction to Course Preview
- ~ Course Overview
- ~ Course Outcome
- ~ Installing Anaconda, Pycharm & Postman
- ~ Working with Conda Envs
- ~ Pycharm Introduction
- ~ Pycharm with Conda
- ~ Pycharm with venv
- ~ Pycharm with Pipenv

### => CNN Foundations :

- ~ Why CNN? Building an Intuition for CNN Preview
- ~ CNN, Kernels, Channels, Feature Maps, Stride, Padding
- ~ Receptive Fields, Image Output Dimensionality Calculations, MNIST Dataset Explorations with CNN
- ~ MNIST CNN Intuition, Tensorspace.js, CNN Explained, CIFAR 10 Dataset Explorations with CNN
- ~ Dropout & Custom Image Classification Dog Cat Dataset
- ~ Deployment in Heroku, AWS, Azure
- ~ Deployment in GCP, AWS EBS

### => CNN Architectures :

- ~ LeNet-5
- ~ LeNet-5 Practical
- ~ AlexNet
- ~ AlexNet Practical
- ~ VGGNet
- ~ VGG16 Practical
- ~ Inception
- ~ Inception Practical
- ~ ResNet
- ~ Resnet Practical

=> Image Classification Hyper Parameter Tuning :

- ~ Keras Tuner
- ~ Building a simple model
- ~ Tuning with Keras Tuner

=> Data Augmentation :

- ~ What is Data Augmentation?
- ~ Benefits of Data Augmentation
- ~ Exploring Papers like RICAP, Random Erasing, Cutout
- ~ Exploring Augmentor
- ~ Exploring Roboflow

=> Object Detection Basics :

- ~ What is Object Detection?
- ~ Competitions for Object Detection
- ~ Bounding Boxes
- ~ Bounding Box Regression
- ~ Intersection over Union (IoU)
- ~ Precision & Recall
- ~ What is Average Precision?

=> Object Detection Architectures :

- ~ Object Detection Family
- ~ RCNN
- ~ RCNN Network Architecture
- ~ Cons of RCNN
- ~ FAST RCNN
- ~ FAST RCNN Network Architecture
- ~ Cons of FAST RCNN
- ~ FASTER RCNN
- ~ FASTER RCNN Network Architecture
- ~ YOLO
- ~ YOLO Architecture
- ~ YOLO Limitations
- ~ SSD
- ~ SSD Network

=> Practicals Object Detection using Tensorflow 1.x :

- ~ Introduction to TFOD1.x
- ~ Using Google Colab with Google Drive
- ~ Installation of Libraries in Colab
- ~ TFOD1.x Setup in Colab
- ~ Visiting the Model Zoo
- ~ Inferencing in Colab
- ~ Inferencing in Local
- ~ Important Configuration Files
- ~ Webcam Testing

=> Practicals Training a Custom Cards Detector using Tensorflow1.x :

- ~ Custom Model Training in TFOD1.x
- ~ Our Custom Dataset
- ~ Doing Annotations or labeling data
- ~ Selection of Pretrained Model from Model Zoo
- ~ Files Setup for Training
- ~ Let's start Training in Colab
- ~ Export Frozen Inference Graph
- ~ Inferencing with our trained model in Colab
- ~ Training in Local
- ~ Inferencing with our trained model in Local

=> Practicals Creating an Cards Detector Web App with TFOD1 :

- ~ Creating a Pycharm project & Environment Setup
- ~ WebApp Workflow
- ~ Code Understanding
- ~ Prediction with Postman
- ~ Debugging our Application

=> Practicals Object Detection using Tensorflow 2.x :

- ~ Introduction to TFOD2.x
- ~ Using the Default Colab Notebook
- ~ Google Colab & Drive Setup
- ~ Visiting TFOD2.x Model Garden
- ~ Inference using Pretrained Model
- ~ Inferencing in Local with a pretrained model

=> Practicals Training a Custom Chess Piece Detector using Tensorflow2 :

- ~ Custom Model training in TFOD2.x
- ~ Our Custom Dataset
- ~ File Setup for Training
- ~ Let's start Training
- ~ Stop Training or resume Training
- ~ Evaluating the trained model
- ~ Convert CKPT to Saved Model
- ~ Inferencing using the Custom Trained Model in Colab
- ~ Inferencing using the Custom Trained Model in Local PC

=> Practicals Creating an Chess Piece Detector Web App with TFOD2 :

- ~ Creating a Pycharm project & Environment Setup
- ~ Application Workflow
- ~ Code understanding
- ~ Testing our App with PoSTmaN

~ Debugging our Application

=> Practicals Object Detection using Detectron2 :

~ Introduction to Detectron2

~ Detectron2 Colab Setup

~ Visiting Detectron2 Model Zoo

~ Detectron2 Pretrained Model Inferencing

=> Practicals Training a Custom Detector using Detectron2 :

~ Detectron2 Custom Training

~ Exploring the Dataset

~ Registering Dataset for Training

~ Let's start Training

~ Inferencing using the Custom Trained Model in Colab

~ Evaluating the Model

=> Practicals Creating an Custom Detector Web App with Detectron2 :

~ Creating a Pycharm project & Environment Setup

~ Application Workflow

~ Code understanding

~ Testing our App with Postman

~ Debugging our Application

=> Practicals Object Detection using YoloV5 :

~ Introduction to YoloV5

~ YoloV5 Colab Setup

~ Inferencing using Pre Trained Model

=> Practicals Training a Custom Warehouse Apparel Detector using YoloV5 :

~ Custom Training with YoloV5

~ Exploring the Dataset

~ Doing Annotations or labeling data

~ Setting up Google Colab & Drive

~ Let's start Training

~ Inferencing using the Custom Trained Model in Colab

=> Practicals Creating an Warehouse Apparel Detector Web App with YOLOV5 :

~ Creating a Pycharm project & Environment Setup

~ Application Workflow

~ Code understanding

~ Testing our App with Postman

~ Debugging our Application

=> Image Segmentation :

~ Segmentation Introduction

~ From Bounding Box to Polygon Masks

~ What is Image Segmentation?

~ Types of Segmentation

~ MASKRCNN

~ MASK RCNN Architecture

=> MASK RCNN Practicaals with TFOD :

~ Segmentation with TFOD1.x

~ Local Setup

~ Exploring the Dataset

~ Data Annotation

~ Model Selection

~ Files Setup for Training

~ Model Training

~ Export Frozen Inference Graph

~ Model Prediction

=> MASKRCNN parctical with Detectron2 :

~ Introduction to Detectron2

~ Detectron2 Colab Notebook

~ Exploring the Model Zoo

~ Detectron2 Colab Setup

~ Custom Training with Detectron2

~ Exploring our Dataset

~ Data Annotation

~ Data Preparation

~ Setup for Training

~ Let's start Training

~ Inferencing using the Custom Trained Model in Colab

~ Evaluating the Model

=> Face Recognition Project :

~ Introduction to Project

~ Requirement Gathering

~ Techstack Selection

~ Project Installation

~ Project Demo

~ Project Workflow

~ Core Components of the Application

~ Data Collection Module

~ Generate Face Embeddings

~ Training Face Recogniton Module

~ Prediction Pipeline

~ Entrypoint of the Application

~ Application Workflow

~ Debugging Application

~ Data Collection Module

- ~ *Generate Face Embeddings*
- ~ *Training Face Recognition Module*
- ~ *Prediction Pipeline*
- ~ *Entrypoint of the Application*
- ~ *Application Workflow*

=> Object Tracking Project

=> GANS

# ML Interview Prepration

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING INTERVIEW

Course link : <https://ineuron.ai/course/ML-Interview-Prepration>

## Course Description :-

Machine Learning Interview-ready course has been created specifically to familiarise you with the types of questions you may encounter during your interview. Machine learning interviews necessitate a broad understanding of machine learning. We've put together a set of questions to see how well you know machine learning principles and technologies.

## Course Features :-

- => Challenges
- => Quizzes
- => Downloadable resources

## What you will learn :-

- => Profile Building
- => System Designing
- => Domain Understanding

## Requirements :-

- => Prior Knowledge of machine Learning
- => A system with stable internet connection
- => Your dedication

## Instructors :-

- => Sudhanshu Kumar :
  - ~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Interview Questions :
  - ~ ML Question Discussion part 1 Preview
  - ~ ML Question Discussion part 2 Preview

# DSA for FAANG preparation with Python and JavaScript

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA MASTERS

Course link : <https://ineuron.ai/course/DSA-for-FAANG-preparation-with-Python-and-JavaScript>

## Course Description :-

A comprehensive chase to excel any interview for the Data Structures and Algorithms. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

## Course Features :-

- => Free LCO DSA Bundle
- => 18 hrs live support all seven day
- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Analysis in Algorithms
- => Data Structure Introduction
- => Array Data Structure
- => Interview Question on array
- => Recursion in depth
- => Divide and Conquer algorithm
- => Applications of Divide and Conquer
- => Linked List Data Structure
- => Interview Question on Linked List
- => Circular Linked List
- => Doubly Linked List
- => Skip List
- => Stack and Queue
- => Interview Question on Stack and Queue
- => Hashing Data Structure
- => Collision Resolution Techniques
- => Tree Data Structure
- => Tree Traversal
- => Binary Search Tree
- => Height Balanced Tree: AVL Tree

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

=> Anurag Tiwari :

~ Hey, I am Anurag Tiwari, a developer at learncodeonline. We have built a scalable system handled by 300K users on a daily basis. I'm a software developer who constantly seeks innovative solutions to everyday problems. I have been teaching students for the last 24 months.

=> Shishir Chandra :

~ Shishir Chandra currently works with Target Platform engineer team as Director of engineering in distributed processing systems space. Shishir has a wide variety of experience in building and scaling up low latency and high throughput systems in various cloud stacks in companies like Adobe, Apple, Cohesity and Imobi in the past. Shishir's area of expertise is in distributed platforms, databases, transactional and realtime systems, big data processing, Microservices and streaming systems and distributed files systems. Shishir is also an open source contributor with elastic search.

## Curriculum details :-

=> Introduction :

~ Course Overview

=> Analysis in Algorithms :

~ Why we need Data structures and algorithms  
~ Introduction to algorithms and its analysis : Time and Space Complexity  
~ Asymptotic Notation: Big O, Omega and Theta Notation  
~ Recurrence Relation Solving: Substitution, Recursive Tree and Master's Theorem

=> Data Structure Introduction :

~ Memory Process - Stack and Heap  
~ Physical and logical data structures  
~ Abstract data types

=> Array Data Structure :

~ Introduction to arrays  
~ Concept of 1D and 2D array (row major order and column major order)  
~ Searching algorithm: linear, binary, ternary search  
~ Concept of inplace and outplace sorting algorithm  
~ Concept of stable and unstable sorting algorithm  
~ Sorting algorithm: comparison(selection, bubble, insertion, quicksort, mergesort, heapsort, shellsort)  
~ Sorting algorithm: Non-comparison(count sort, bucket sort, radix sort)

=> Interview Question on array :

~ Rotation of an array  
~ Finding of missing number in an array  
~ Division of two integers without using division operator  
~ Search in rotated array  
~ Target triplet  
~ Stock buy sell to maximize profit

=> Recursion in depth :

~ Introduction to recursion  
~ Tracing the recursion tree  
~ Types of recursion  
~ Complex recursion tree  
~ Classic Tower of Hanoi problem

=> Divide and Conquer algorithm :

~ Introduction to Divide and Conquer

=> Applications of Divide and Conquer :

~ Finding of maxima and minima  
~ Finding of power of an element  
~ Binary Search  
~ MergeSort  
~ QuickSort  
~ Selection Procedure  
~ Finding of number of inversions  
~ Strassens' matrix multiplication

=> Linked List Data Structure :

~ Introduction to linked list  
~ Insertion of a node(beginning, end and at any position) in linked list  
~ Deletion of a node(beginning, end and at any position) in linked list  
~ Searching of a node in linked list

=> Interview Question on Linked List :

~ Reversal of a node in linked list  
~ Count of all nodes in linkedlist  
~ Floyd's cycle detection algorithm  
~ Merge two linked list

=> Circular Linked List :

~ Circular Linked List Theory  
~ Insertion of a node in circular linked list  
~ Traversal of a node in circular linked list  
~ Deletion of a node in circular linked list  
~ Count of number of nodes in circular linked list  
~ Conversion of linked list to circular linked list

=> Doubly Linked List :

~ Doubly Linked List Theory  
~ Insertion of a node in doubly linked list  
~ Traversal of a node in doubly linked list  
~ Deletion of a node in doubly linked list

=> Skip List :

~ Introduction to skip list  
~ Build-in skip list  
~ Search in skip list  
~ Insertion in skip list



~ *Deletion in skip list*

=> Stack and Queue :

- ~ *Stack: Push and Pop operation*
- ~ *Implementation of Stack using array and linked list*
- ~ *Queue concept theory*
- ~ *Implementation of Queue using array and linked list*
- ~ *Circular Queue theory*
- ~ *Implementation of Circular Queue*

=> Interview Question on Stack and Queue :

- ~ *Stack using queue conceptual understanding*
- ~ *Implementation of stack using queue*
- ~ *Queue using stack conceptual understanding*
- ~ *Implementation of queue using stack*
- ~ *Valid brackets*
- ~ *Stock Spanning*

=> Hashing Data Structure :

- ~ *Introduction to Hashing Data Structure*
- ~ *Hash Function and its types*

=> Collision Resolution Techniques :

- ~ *Chaining*
- ~ *Open Addressing: Linear Probing, Quadratic Probing, Double Hashing, Perfect Hashing, Consistent Hashing*
- ~ *Application: Bloom Filters*
- ~ *Two Sum Problem*

=> Tree Data Structure :

- ~ *Introduction to Binary Tree*
- ~ *Complete Binary Tree and almost complete binary tree*
- ~ *Full binary tree and representation using array and linked list*

=> Tree Traversal :

- ~ *Introduction to tree traversal*
- ~ *Inorder Traversal*
- ~ *Preorder Traversal*
- ~ *Postorder Traversal*

=> Binary Search Tree :

- ~ *Introduction to Binary Search Tree*
- ~ *Insertion and Deletion in BST*
- ~ *Inorder traversal in BST gives sorted array*
- ~ *Searching in Binary Search Tree*
- ~ *Deletion in Binary Search Tree*

=> Height Balanced Tree: AVL Tree :

- ~ *Introduction: Why AVL Tree?*
- ~ *Creation of an AVL Tree*
- ~ *Insertion in AVL Tree*
- ~ *Searching in AVL Tree*
- ~ *Deletion in AVL Tree*

=> Height Balanced Tree: Red Black Tree :

- ~ *Introduction: Why Red Black Tree?*
- ~ *Properties of Red Black Tree*
- ~ *Creating of Red Black Tree*
- ~ *Insertion Rules in Red Black Tree*
- ~ *Searching in Red Black Tree*
- ~ *Deletion in Red Black Tree*

=> B and B+ Tree: Usage in Databases :

- ~ *Creation of B and B+ Tree*
- ~ *Insertion in B and B+ Tree*
- ~ *Searching in B and B+ Tree*
- ~ *Deletion in B and B+ Tree*

=> Interview Question on Tree :

- ~ *Checking of whether the tree is symmetric or not*
- ~ *Count of number of possible BSTs in a given number of nodes*
- ~ *Catalan number concept to find the number of BST*
- ~ *Level order traversal of a tree*
- ~ *Flip or inverse of a binary tree*
- ~ *Same tree problem*
- ~ *Inorder iterator*
- ~ *Binary Tree Zigzag level order traversal*

=> Graph Traversal Algorithms :

- ~ *Introduction to Graph Traversal Algorithms*
- ~ *Introduction to Depth First Search*
- ~ *DFS Psuedocode and illustration using an example*
- ~ *DFS Coding Implementation*
- ~ *Introduction to Breadth First Search*
- ~ *BFS Psuedocode and illustration using an example*

=> Interview Questions on Graph :

- ~ *Clone of a graph*
- ~ *DFS and Cycle detection with University course problem*
- ~ *Island problem*

=> Heap Data Structure :

- ~ *Introduction to Heap Data Structure*
- ~ *Maxheap and Minheap Overview*

- ~ Insertion in Minheap
- ~ Deletion in Minheap
- ~ Creation of Minheap
- ~ Mathematical derivation to analyse the complexity of creation of minheap
- ~ HeapSort algorithm and why it is not stable algorithm

=> Interview Based Question on Heap Data Structure :

- ~ Maximum Product of three numbers in an array
- ~ Finding of K-closest points from an origin

=> Greedy Algorithm :

- ~ Introduction to greedy algorithm

=> Application of greedy algorithm :

- ~ Fractional Knapsack Problem
- ~ Minimum Spanning Tree: Kruskal and Prim's Algorithm
- ~ Single Source Shortest Path: Dijkstra's algorithm
- ~ Huffman Coding
- ~ Optimal Merge Pattern
- ~ Job Sequencing with Deadline

=> Dynamic Programming :

- ~ Introduction to Dynamic Programming
- ~ Overlapping subproblem in dynamic programming
- ~ Tabulation in dynamic programming
- ~ Memoization in dynamic programming

=> Application of Dynamic Programming :

- ~ Fibonacci Series
- ~ Longest Common Subsequence
- ~ 0/1 Knapsack Problem
- ~ Sum of subset
- ~ All Pair Shortest Path: Floyd Warshall Algorithm
- ~ Bellman Ford Algorithm

=> Interview Problems on Dynamic Programming :

- ~ Knapsack - Coke, Pepsi, Redbull
- ~ Largest sum of subset
- ~ Coin change problem
- ~ Largest sum
- ~ Minimum path to reach target

=> String Matching Algorithms :

- ~ Introduction to String matching algorithms
- ~ Naive String Matching algorithms
- ~ Rabin Karp Algorithm
- ~ Kuth-Morris-Pratt(KMP) Pattern Matching

=> Interview Problems on String :

- ~ Word in a sentence
- ~ Inplace duplicates
- ~ Longest substring
- ~ Palindrome makes and breaks

=> NP-Hard and NP-Complete Problem :

- ~ NP-Hard Problem
- ~ NP-Complete Problem

=> Approaching Design :

- ~ Understanding and clarification
- ~ Business usecase of the problem and knowing the consumers
- ~ Iron out the Functional requirements
- ~ Importance of discussing the trade-offs based on the usecase in picture
- ~ Mastering the art of selling design
- ~ Data model approaches and fitment
- ~ LLD modelling and future readiness of design
- ~ Explaining the features of design like adherence to proper design patterns

=> Introduction to System Design :

- ~ Introduction to system design
- ~ Importance of architecture
- ~ Distinction between HLD and LLD
- ~ Importance of data modelling
- ~ Importance of documentation in design

=> Practicing some real designs :

- ~ Rate limiting
- ~ Uber riders app
- ~ Whatsapp messaging
- ~ food delivery app building
- ~ Booking app building
- ~ Video streaming systems
- ~ Q&A

# Cyber Security Masters

---

Topic Name : CYBER SECURITY

Sub-topic Name : CYBERSECURITY MASTERS

Course link : <https://ineuron.ai/course/Cyber-Security-Masters>

## Course Description :-

One of the most famous FAQ on google is How to get started with Ethical hacking? The perfect way to find the correct opportunity according to your potential is to take a deep dive into a course that has a great variety with a lot of practical practice. This is an ISO 27001 government-certified course that covers Bug Bounty & Web Pentesting, system exploitation, network pentesting, android pentesting, Forensics, etc. From this course, our students have found bugs in big organizations like Harvard, Huawei, Nike, BMW, Doordarshan, Nykaa, GoodRx etc.

## Course Features :-

- => 3 months programme
- => ISO 27001 Certified Cyber Security Master's Certification
- => On-site Internship for Top Scholars
- => 1:1 Personalised Mentorship
- => Live Teaching by Instructors
- => Industry level Testing walkthrough
- => Every week doubt clearing sessions
- => Doubt Clearing through mail and skype support team
- => Assignment and Quizzes in all the modules
- => Resume Building
- => Career Guidance
- => Interview Preparation
- => Job Fair & Internal Hiring

## What you will learn :-

- => How to make money from Bug Bounty?
- => Hand on Kali Linux
- => Linux Fundamentals Crash Course
- => Complete Burpsuite module training
- => Bug Bounty
- => Web Pentesting
- => Owasp top 10
- => 30+ Latest Web Attacks
- => OSI Model
- => Networking Fundamental
- => Wireshark
- => Nmap
- => System Security
- => Exploit and Malwares
- => Forensics
- => Man in the middle attack
- => Anonymity
- => How to crack cyber security interviews

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication
- => I3 Processor or better
- => At least 4GB Ram

## Instructors :-

=> Saksham Choudhary :

~ Hello folks, I am AWS certified Cloud Architect Engineer. With having 5+ years of Experience in Teaching, I am currently providing cloud solutions for various products via my strong hands on DevOps Skills. I am a tech youtuber with 120k + subscriber and has taught 30,000 + students, Narcotics, Custom duty officers, Police officials and Corporate candidates.

## Curriculum details :-

=> Welcome to the world of Penetration Testing :

- ~ Course Introduction
- ~ Why Web Penetration Testing?
- ~ Types of Hackers Preview
- ~ Disclaimer for this course Preview
- ~ What is Vulnerability?
- ~ What is VAPT?
- ~ What is Owasp top 10
- ~ Scope & Duties of Web Pentester in InfoSec Companies
- ~ Goals for Resume Building in Web Pentesting
- ~ Case Study of Reporting bug at Harvard Preview

=> Setting up an environment :

- ~ Things to cover in this section
- ~ Grabbing Required downloadable resources for this section
- ~ Learning Virtualization with Virtual Box
- ~ Setting up & Walkthrough of Vbox modules
- ~ Introduction & History of Linux
- ~ Why Kali Linux?
- ~ Installing Kali Linux
- ~ Tweaks to Run Kali Linux Smoothly Part 1 Preview
- ~ Tweaks to Run Kali Linux Smoothly Part 2
- ~ Updating and Upgrading Kali Linux with Debian packages

=> Linux Fundamentals Crash Course :

- ~ Introduction to command prompt
- ~ Accessing system & Network Related commands
- ~ Ip Config & Bridge network
- ~ Linux file system and Structure
- ~ Introduction to root
- ~ Absolute and relative paths
- ~ Directory listing attributes
- ~ Playing with file and directories
- ~ Different file types in Linux
- ~ Wildcard commands
- ~ Understanding files and Directory permissions
- ~ File permission commands Preview
- ~ Help commands, auto completion and arrow keys
- ~ Piping process
- ~ Linux file editors
- ~ Switching user with sudo module
- ~ System utility commands (Date, Uptime, Hostname, Uname etc)
- ~ Installing softwares
- ~ Github clone to run tools
- ~ Compiling python
- ~ Compiling java

=> Core fundamentals for web pentesting :

- ~ What is an Ip address?
- ~ What is protocol? HTTP & HTTPS
- ~ Subdomain & Domain name
- ~ What are ports?
- ~ Path & Query component in URL
- ~ Parameters and Fragment
- ~ Explaining verbs, What is GET method?
- ~ What is Post Method?
- ~ What is Put Method?
- ~ Delete & Head Method
- ~ Connect & Options
- ~ Trace & Patch
- ~ How does an API works?
- ~ HTTPS Status code part 1
- ~ HTTPS Status code part 2

=> Complete Burpsuite module training :

- ~ What is Burp Suite?
- ~ Burp CA Certificate for SSL/TLS
- ~ Burp Project Type : New, Existing & Temp
- ~ Burp Suite Proxy
- ~ Burpsuite Intruder
- ~ Burpsuite Scanner
- ~ Burp suite Repeater
- ~ Burp Suite Sequencer
- ~ Burp Suite Decoder
- ~ Burp Suite Comparer
- ~ What are Payloads? Simple List, Runtime file, Custom iterator
- ~ Payload type : Character Substitution, Case Modification, Recursive grep
- ~ Payload Type : Illegal Unicode, Character Blocks, Numbers
- ~ Payload Types : Dates, Brute Forcer, Null Payloads, Character Frobber
- ~ Payload types : Bit Flipper, Username Generator, ECB Block Shuffler

- ~ *Burp Suite Extender*
- ~ *Burp Suite Extensions*
- ~ *BApp Store*
- ~ *Burp Suite APIs*
- ~ *Burp Suite Options*
- ~ *Engagement Tools*
- ~ *Http History Analyser*
- ~ *Connect Burp to Android for Testing Android Apps*

#### => Reconnaissance Methodology :

- ~ *DNS Records with Virustotal*
- ~ *HTTP Status Recon*
- ~ *Subdomain enumeration Preview*
- ~ *Aquatone*
- ~ *Shodan Research*
- ~ *Directory Bruteforcing*
- ~ *Digging into the past with WayBack Machine*
- ~ *Certificate Transparency Crt*
- ~ *Wappalyzer for Technology Identification*
- ~ *Netcraft Active Cyber Defence*

#### => Getting started with Testing environment :

- ~ *What is DVWA?*
- ~ *Getting started by Creating Database & User for lab*
- ~ *Configuring DVWA*
- ~ *DVWA Error Solving*

#### => Brute force & Command Injection :

- ~ *Brute force technique part 1*
- ~ *Brute force technique part 2*
- ~ *What is Command Injection & CI Low level breach*
- ~ *Command Injection: Breaching Medium Level Security*
- ~ *Command Injection: Breaching High Level Security*
- ~ *Command Injection Mitigation & Secure Code writing logic*
- ~ *Remote Code Execution Incident Report Study*

#### => Insecure Session Management & Cookie Manipulating Flaw :

- ~ *Insecure Session Management & Cookie Manipulating Flaw*
- ~ *Insecure JSON Parsing*

#### => Cross Site Request Forgery :

- ~ *What is Cross Site Request Forgery? CSRF Part 1*
- ~ *CSRF: Part 2 (Designing Custom CSRF Form)*
- ~ *CSRF: Execution of Custom form and Mitigation Technique*
- ~ *CSRF: Automated form via Burpsuite*
- ~ *CSRF Incident Report Study*

#### => File Upload Vulnerability :

- ~ *What is File Upload Vulnerability? Breaching Low Level*
- ~ *Breaching Medium Level*
- ~ *Breaching High Level & Mitigation*
- ~ *File Upload Incident Report Study*

#### => File Inclusion Vulnerability :

- ~ *Local & Remote File Inclusion (Low Level)*
- ~ *LFI & RFI (Medium & High Level)*
- ~ *LFI & RFI Incident Report Study*

#### => SQL Injection :

- ~ *SQL Injection Master Lab & What is Database?*
- ~ *SQL Fundamentals*
- ~ *What is ID, Joining & Breaking the query in SQL*
- ~ *Selecting Vulnerable Column & Fetching Database Name*
- ~ *Dumping Database*

#### => Boolean Based & SQL Automation :

- ~ *Boolean Based Queries & Fundamentals*
- ~ *Boolean Based demonstration*
- ~ *Automation With SQL Map*

#### => Cross Site Scripting :

- ~ *Reflecting XSS*
- ~ *Stored XSS*
- ~ *Dom Bases XSS*
- ~ *Exploring Innovative method for executing XSS via Case Studies*

#### => Increasing Difficulty with WebGoat :

- ~ *Gathering Pre-Requisites for WebGoat*
- ~ *Configuring WebGoat in Windows*

#### => Token Exploitation :

- ~ *What is JSON Web Token? (JWT)*
- ~ *JWT : JSON Web Token Hijacking with SQL Injection*
- ~ *JWT Payment Gateway Manipulation*

#### => Password Reset EndPoint :

- ~ *Password Reset Endpoint*
- ~ *Creating and Exploiting Password Reset Link*

#### => Path Traversal :

- ~ *Path Traversal - Bypass File Upload Fix 1*
- ~ *Path Traversal - Bypass File Upload Fix 2*
- ~ *Path Traversal - Retrieving Files*

=> SQL String Based :

- ~ *String SQL Injection Part 1*
- ~ *String SQL Injection Part 2*
- ~ *Delete Data & Retrieve Data from Tables*
- ~ *SQL Login Attack*

=> HTML Tempering & XXE :

- ~ *HTML Tempering explained with Execution*
- ~ *XXE : What is XXE Injection?*
- ~ *XXE Injection Content Type Manipulation*
- ~ *Blind XXE Injection*

=> Insecure Direct Object Reference :

- ~ *What is IDOR?*
- ~ *Data Extraction via IDOR*
- ~ *Account Hijacking via IDOR*

=> Advance CSRF & SSRF :

- ~ *Login CSRF*
- ~ *SSRF Explained*
- ~ *SSRF - Request Manipulation to display User*

=> Bonus Attacks :

- ~ *Vulnerable Components - Exploiting CVE*
- ~ *Meta Data Sanitization*
- ~ *Client-Side Filtering*

=> Wireshark :

- ~ *OSI Model Layer*
- ~ *Split of Concentration*
- ~ *Application layer*
- ~ *Presentation Layer*
- ~ *Session layer*
- ~ *Top Layer vs*
- ~ *Transport Layer*
- ~ *Network Layer*
- ~ *Data link Layer*
- ~ *Physical Layer*
- ~ *Host Communication*
- ~ *Encapsulation*
- ~ *TCP/IP vs OSI Model*
- ~ *Wireshark Filters & Data Capture*

=> Nmap :

- ~ *Nmap Basics, Target Specification & Port States*
- ~ *Nmap Scanning & Ping Scanning*
- ~ *Nmap Scan Techniques with SYN, Connect, UDP, SCTP, TCP, ACK & Window*
- ~ *Nmap Scan Techniques Part 2 : Null, Fin, XMAS, Maimon, IDLE Scan & IP Protocol*
- ~ *Nmap Performance, Firewall & IDS Evasion*

=> Exploits :

- ~ *What is metasploit?*
- ~ *How port scanning can help us in exploiting machines?*
- ~ *How to Configure Exploits?*
- ~ *Executing Eternal Blue exploit on Windows Machine*
- ~ *Microsoft Windows 10 (1903/1909) - 'SMBGHOST' SMB3*
- ~ *Microsoft Windows 10 (1903/1909) - 'SMBGHOST' SMB3*

=> Forensics :

- ~ *Analysis - Registry, Email and Browser Artifacts*
- ~ *Analysis - PDF Files and Page Files*
- ~ *Malware File Analysis*
- ~ *USB Forensics - Detection and Investigation*
- ~ *Meta Data Analysis - MS Office Files*
- ~ *Meta Data Analysis - Image Files*
- ~ *Memory Forensics using FTK Imager and Volatility3 tool overview*
- ~ *Volatility3 - Memory File Analysis and Infected system file*

=> Final Module :

- ~ *Pentsting with Automated tools : Owasp Zap*
- ~ *Hittrack & Wpscan*
- ~ *What is Accunetix?*
- ~ *Accunetix Practical Scanning*
- ~ *How to Make POC (Proof of Concept)*
- ~ *How to make a VAPT (Vulnerability Assessment & Penetration Testing Report) report : Part 1*
- ~ *VAPT Part 2*
- ~ *How to get Job Ready and CV guide*
- ~ *What to learn next?*
- ~ *Final Closure*

=> Extra Module :

- ~ *Maintaining Anonymity with Tor Browser & Dark Net Walkthrough*
- ~ *Proxychaining & Masking Mac Address*
- ~ *Man in the middle attacks (MITM) via ARP Spoofing*
- ~ *Phone Hacking*

=> Interview Prep :

- ~ *Mock Interview: Level 1*
- ~ *Mock Test Paper (Practical Skill Based): Level 2*
- ~ *Group Discussion Round: Level 3*
- ~ *Resume Building*

=> Examination :  
~ Final Examination

# Full Stack Data Analytics Tech Neuron

---

Topic Name : DATA ANALYTICS

Sub-topic Name : BUSINESS ANALYTICS MASTERS

Course link : <https://ineuron.ai/course/Full-Stack-Data-Analytics-Tech-Neuron>

## Course Description :-

Data analytics is the process of gathering, transforming, and organizing information in order to draw conclusions, make predictions, and make better decisions. The full stack data analytics course is meant to assist you in becoming a skilled data analyst. Learn how to deal with SQL databases, develop data visualizations, and apply predictive analytics and statistics in a corporate environment using the best analytics tools and methodologies.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Statistics
- => Basic Charts in Power BI
- => Working with Maps
- => Slicers in Power BI
- => Cards and Filters
- => Power Query
- => M Language
- => Tableau
- => SQL
- => Python
- => Excel

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Statistics :
  - ~ Introduction
  - ~ Different types of Statistics
  - ~ Population vs Sample
  - ~ Mean, Median and Mode
  - ~ Variance, Standard Deviation
  - ~ Sample Variance why  $n-1$
  - ~ Standard Deviation
  - ~ Variables
  - ~ Random Variables
  - ~ Percentiles & quartiles
  - ~ 5 number summary
  - ~ Histograms
  - ~ Gaussian - Normal distribution
  - ~ Standard Normal distribution
  - ~ Application Of Zscore
  - ~ Basics Of Probability
  - ~ Addition Rule In Probability
  - ~ Multiplication rule in probability
  - ~ Permutation
  - ~ Combination
  - ~ Log Normal Distribution
  - ~ Central Limit theorem
  - ~ Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median And Mode
  - ~ Covariance



- ~ *Pearson And Spearman Rank Correlation*
- ~ *What is P Value*
- ~ *What is Confidence Intervals*
- ~ *How To Perform Hypothesis Testing - Confidence IntervalZ Test Statistics Derive Conclusion*
- ~ *Hypothesis testing part 2*
- ~ *Hypothesis testing part 3*
- ~ *Finalizing statistics*

#### => Basic Charts in Power BI :

- ~ *2.0 Basic Charts in Power BI Desktop*
- ~ *2.1 Column Chart in Power BI*
- ~ *2.2 Stacked Column Chart in Power BI*
- ~ *2.3 Pie Chart in Power BI*
- ~ *2.4 Donut Chart in Power BI*
- ~ *2.5 Funnel Chart in Power BI*
- ~ *2.6 Ribbon Chart*
- ~ *2.7 Include and Exclude*
- ~ *2.8 Export data from Visual*

#### => Working with Maps :

- ~ *3.1 Creating a Map in Power BI*
- ~ *3.2 Filled Map*
- ~ *3.3 Map with Pie Chart*
- ~ *3.4 Formatting in Map*
- ~ *3.5 Change Background in Map*
- ~ *3.6 Map of India in Power BI*
- ~ *3.7 Map of Australia in Power BI*

#### => Tables and Matrix in Power BI :

- ~ *4.0 Table and Matrix in Power BI*
- ~ *4.1 Creating a Table in Power BI*
- ~ *4.2 Formatting a Table*
- ~ *4.3 Conditional Formatting in Table*
- ~ *4.4 Aggregation in Table*
- ~ *4.5 Matrix in Power BI*
- ~ *4.6 Conditional Formatting in Matrix*
- ~ *4.7 Hierarchy in Matrix*
- ~ *4.8 Sub-Total and Total in Matrix*
- ~ *4.9 Number Formatting in Table*

#### => Other Charts in Power BI :

- ~ *5.0 Other Charts in Power BI*
- ~ *5.1 Line Chart in Power BI*
- ~ *5.2 Drill Down in Line Chart*
- ~ *5.3 Area Chart in Power BI*
- ~ *5.4 Line vs Column Chart in Power BI*
- ~ *5.5 Scatter Plot in Power BI*
- ~ *5.6 Waterfall Chart in Power BI*
- ~ *6.7 TreeMap in Power BI*
- ~ *5.8 Gauge Chart in Power BI*

#### => Cards and Filters :

- ~ *6.0 Cards and Filters in Power BI*
- ~ *6.1 Number Card*
- ~ *6.2 Text Card*
- ~ *6.2.1 Formatting of Text Card*
- ~ *6.3 Date Card*
- ~ *6.3.1 Date Card (Relative Filtering)*
- ~ *6.4 Multi-Row Card*
- ~ *6.5 Filter on Visual*
- ~ *6.6 Filter on This Page*
- ~ *6.7 Filter on All Pages*
- ~ *6.8 Drillthrough in Power BI*

#### => Slicers in Power BI :

- ~ *7.0 Slicers in Power BI*
- ~ *7.1 Text Slicers in Power BI*
- ~ *7.2 Formatting a Text Slicer*
- ~ *7.3 Date Slicers in Power BI*
- ~ *7.4 Formatting a Date Slicer*
- ~ *7.5 Number Slicers in Power BI*

#### => Advanced Charts in Power BI :

- ~ *8.0 Advanced Charts in Power BI*
- ~ *8.1 Animated Bar Chart Race*
- ~ *8.2 Drill down donut Chart*
- ~ *8.3 Drill down Column chart*
- ~ *8.4 Word Cloud in Power BI*
- ~ *8.5 Sankey Chart in Power BI*
- ~ *8.6 Infographic in Power BI*
- ~ *8.7 Play Axis in Power BI*
- ~ *8.8 Scroller in Power BI*
- ~ *8.9 Sunburst Chart in Power BI*
- ~ *8.10 Histogram in Power BI*

#### => Objects in Power BI :

- ~ *9.1 Insert Image in Power BI*
- ~ *9.2 Insert Text in Power BI*
- ~ *9.3 Insert Shapes in Power BI*
- ~ *9.4 Insert Buttons in Power BI*
- ~ *9.5 Web URL Action in Power BI*

~ 9.6 Page Navigation Action in Power BI

~ 9.7 Bookmark Action in Power BI

~ 9.8 Drillthrough Action in Power BI

=> Power BI Service Introduction :

~ 10.1 Create a Superstore Report in Power BI

~ 10.2 Create an Account on Power BI Service

~ 10.3 Publish Report to Power BI Service Account

~ 10.4 Export Power BI Report to PPT, PDF or PBIX

~ 10.5 Comment, Share and Subscribe to Power BI Report

~ 10.6 Create a Dashboard in Power BI Service

~ 10.7 Problem in Power BI Dashboard and its solution

~ 10.8 Automatic Refresh in Power BI using Gateway

=> Power Query - Text Functions :

~ 11.0 Text Functions in Power Query (Power BI)

~ 11.1 Merge Columns in Power Query (Power BI)

~ 11.2 Split and Trim in Power Query (Power BI)

~ 11.3 Upper, Lower and ProperCase in Power Query (Power BI)

~ 11.4 Prefix and Suffix in Power Query (Power BI)

~ 11.5 Left, Right and Mid Functions in Power Query (Power BI)

~ 11.6 Extract Text with Delimiters

=> Power Query - Date Functions :

~ 12.0 Date Functions in Power Query (Power BI)

~ 12.1 Year, Quarter, Month and Day Functions in Power Query (Power BI)

~ 12.2 Find Difference between Dates in Power Query (Power BI)

~ 12.3 Month and Day Name in Power Query (Power BI)

~ 12.4 Day, Week of Month, Year in Power Query (Power BI)

~ 12.5 Extract Date, Time in Power Query (Power BI)

~ 12.6 Calculate Age in Power Query (Power BI)

~ 12.7 Day of Year, Quarter, Month in Power Query (Power BI)

=> Power Query - Number Functions :

~ 13.0 Number Functions in Power Query (Power BI)

~ 13.1 Basic Number Functions in Power Query (Power BI)

~ 13.2 Percentage, Percent Of, Module in Power Query (Power BI)

~ 13.3 Round Functions in Power Query (Power BI)

~ 13.4 IsEven, IsODD, Sign in Power Query (Power BI)

=> Power Query - Append Files :

~ 14.1 Append multiple CSV files in a folder in Power Query (Power BI)

~ 14.2 Append multiple excel sheets, Tables in Power Query (Power BI)

~ 14.3 Append Excel sheets or Tables with different columns in Power BI

~ 14.4 Append multiple Excel files from a folder in Power BI

~ 14.5 Append different data sources in Power BI

=> Power Query - Merge Files :

~ 15.0 Merge Files and Tables in Power BI

~ 15.1 Merge Sheets or Tables in Power Query (Power BI)

~ 15.2 Merge Data from multiple Excel files or Workbooks in Power BI

~ 15.3 Merge data from different data sources in Power Query (Power BI)

~ 15.4 Merge data having multiple criteria in Power BI

=> Power Query - Conditional Columns :

~ 16.0 Conditional Column and Column from example in Power BI

~ 16.1 Column from examples in Power BI - Split Text

~ 16.2 Column from examples in Power BI - Merge Columns

~ 16.3 Column from Examples in Power BI - Date

~ 16.4 Column from Examples in Power BI - Alphanumeric

~ 16.5 Conditional Column in Power BI - One Column

~ 16.6 Conditional Column in Power BI - two columns

~ 16.7 Conditional Column in Power BI - Compare two columns

~ 16.8 Conditional Column in Power BI - on Dates

=> Power Query - - Important Topics :

~ 17.0 Very Important Topics in Power Query (Power BI)

~ 17.1 Fill Down in Power BI

~ 17.2 Grouping in Power Query (Power BI)

~ 17.3 Transpose in Power Query (Power BI)

~ 17.4 Unpivot In Power Query (Power BI)

~ 17.5 Data Types in Power Query (Power BI)

~ 17.6 Replace Errors and Values in Power Query (Power BI)

~ 17.7 Keep and Remove Rows in Power Query (Power BI)

~ 17.8 Add, Remove and Goto Columns in Power Query (Power BI)

=> M Language Introduction :

~ 18.0 M Language in Power Query

~ 18.1 Introduction to M Language

~ 18.2 IsIn Date Functions in M Language - Power BI

~ 18.3 Add and Subtract Date M Functions in Power BI

~ 18.4 Basic Date M Functions in Power BI

~ 18.5 Basic Text M Functions in Power BI

~ 18.6 Simple M Code in Power BI

~ 18.7 Trick to get all 900+ M Functions in Power BI

=> Introduction to tableau :

~ Tableau Introduction

~ Download and Install Tableau

~ Tableau Vs Excel

=> Charts - 1 :

- ~ Column Chart
- ~ Horizontal Bar Chart
- ~ Stacked Column Chart
- ~ Stacked Bar Chart
- ~ Keep Only, Exclude
- ~ Keep Only, Exclude2\_Normal
- ~ Publish to Tableau Public

#### => Charts - 2 :

- ~ Pie Chart
- ~ Multiple Pie Chart
- ~ TreeMap\_Editing
- ~ Packed Bubble Chart
- ~ Word Cloud OR Word Map
- ~ Formatting payal

#### => Charts - 3 :

- ~ Data Types in Tableau
- ~ Filled Map
- ~ Symbol Maps
- ~ India Map
- ~ Histogram

#### => Charts - 4 :

- ~ Text Table
- ~ Text Table with Multiple Measures
- ~ Measure Names and Measure Values
- ~ Line Chart
- ~ Line Chart with Multiple Measures
- ~ Discrete Vs Continuous Line Chart
- ~ Discrete Vs Continuous

#### => Charts - 5 :

- ~ Lollipop Chart
- ~ Line Vs Column Chart
- ~ Dual Axis Chart
- ~ Column vs Shapes
- ~ Bar in Bar Chart

#### => Charts - 6 :

- ~ Calculated fields
- ~ Conditional Column Chart
- ~ Column chart with Shapes based on condition
- ~ Conditional Maps

#### => Charts - 7 :

- ~ Map with Pie Chart
- ~ Map with WMS

#### => Charts - 8 :

- ~ Funnel Chart
- ~ Advanced Funnel Chart
- ~ Calendar
- ~ Dumbbell Chart
- ~ Donut Chart
- ~ Multiple Donut Chart

#### => Charts - 9 :

- ~ Bullet Chart 1
- ~ Bullet Chart 2
- ~ Table Calculations Part 1
- ~ Table Calculations - Compute Using - Part 2
- ~ Table Calculations - Relative - Part 3
- ~ Bump Chart
- ~ Bump Chart with Circle
- ~ 100 Percent Stacked Column Chart

#### => Charts - 10 :

- ~ Scatter Plot
- ~ Scatter Plot with Images OR Shapes
- ~ Bubble Chart
- ~ Animation - Column Chart
- ~ Animation - Line Chart
- ~ Animation - Column vs Line Chart

#### => Charts - 11 :

- ~ Heat Maps
- ~ Heat Map with Shapes
- ~ Heat Map with Conditional Formatting
- ~ Pareto Chart
- ~ Rounded Bar Chart

#### => Introduction and installation of MySQL :

- ~ Introduction to section 1
- ~ MySQL introduction - 5 points to know
- ~ MySQL Installation MAC

#### => Basics of MySQL :

- ~ Introduction to section 2
- ~ Creating and dropping database - Startup
- ~ Resolving the issue for future
- ~ Creating your first table

- ~ Adding values to canon table
- ~ Answering customer question

=> Playing with data :

- ~ Introduction to section 3
- ~ Primary key, default and NULL
- ~ Table with primary key and default values
- ~ Testing the new table
- ~ Adding new values and answering questions
- ~ Update in customers table
- ~ Delete from the customers table

=> More on functions :

- ~ Introduction to section 4
- ~ Understand the new lco user DB
- ~ Task for CONCAT
- ~ Task for REPLACE
- ~ task for SUBSTRING
- ~ Task for reverse and CHAR\_LENGTH
- ~ Task for case conversion and DOCS

=> Answering some DB questions :

- ~ Introduction to section 5
- ~ A task on DISTINCT
- ~ A task for ORDER BY
- ~ A task on LIMIT
- ~ Match the pattern
- ~ A task on COUNT
- ~ SQL MODES and GROUP BY
- ~ MIN MAX and SUBQUERIES
- ~ GROUP BY with MAX and MIN
- ~ SUM and AVERAGE with GROUP BY
- ~ A task on AND OR
- ~ A task in RANGE based selection
- ~ CASE THEN - multiple range selection

=> A pinch of theory :

- ~ Introduction to section 6
- ~ Data type for INTEGER and STRING
- ~ Data type for DATE, DATETIME and JSON
- ~ DATE TIME code Example
- ~ Get the date and time
- ~ Lets join tom and jerry tables
- ~ Types of JOIN

=> FOREIGN KEY and JOINS :

- ~ Introduction to section 7
- ~ Code talk over FOREIGN keys
- ~ Understand a new database
- ~ A task on INNER join
- ~ ONE to MANY and MANY TO MANY
- ~ Join more 3 or more tables
- ~ A task on LEFT JOIN
- ~ A task on RIGHT JOIN
- ~ FULL OUTER join and UNION tasks

=> A pinch of more theory :

- ~ Introduction to section 8
- ~ Database engines - INNODB and more
- ~ ACID in database

=> A 30 Task assignment for movie DB :

- ~ Introduction to section 9
- ~ How practice database works - FILM

=> Final exam - single attempt :

- ~ MYSQL Outro and some free resources

=> Python Basics :

- ~ Python Introduction, Installation and Setup
- ~ Python Basics & Conditionals
- ~ Conditionals & Loops
- ~ Working with Loops
- ~ Working with Strings & Lists
- ~ List manipulation
- ~ Tuple, Set & Dictionary
- ~ Working with Functions

=> Microsoft Excel Fundamentals :

- ~ Launching Excel
- ~ Microsoft Excel Startup Screen
- ~ Introduction to the Excel Interface
- ~ Customizing the Excel Quick Access Toolbar
- ~ More on the Excel Interface
- ~ Understanding the Structure of an Excel Workbook
- ~ Saving an Excel Document
- ~ Opening an Existing Excel Document
- ~ Common Excel Shortcut Keys

=> Entering and editing text and formulas :

- ~ Entering Text to Create Spreadsheet Titles
- ~ Working with Numeric Data in Excel

- ~ *Entering Date Values in Excel*
- ~ *Working with Cell References*
- ~ *Creating Basic Formulas in Excel*
- ~ *Relative Versus Absolute Cell References in Formulas*
- ~ *Understanding the Order of Operation*

=> Working with basic excel functions :

- ~ *The structure of an Excel Function*
- ~ *Working with the SUM() Function*
- ~ *Working with the MIN() and MAX() Functions*
- ~ *Working with the AVERAGE() Function*
- ~ *Working with the COUNT() Function*
- ~ *Adjacent Cells Error in Excel Calculations*
- ~ *Using the AutoSum Command*
- ~ *Excel's AutoSum Shortcut Key*
- ~ *Using the AutoFill Command to Copy Formulas*

=> Modifying an excel worksheet :

- ~ *Moving and Copying Data in an Excel Worksheet*
- ~ *Inserting and Deleting Rows and Columns*
- ~ *Changing the Width and Height of Cells*
- ~ *Hiding and Unhiding Excel Rows and Columns*
- ~ *Renaming an Excel Worksheet*
- ~ *Deleting an Excel Worksheet*
- ~ *Moving and Copying an Excel Worksheet*

=> Formatting data in an excel worksheet :

- ~ *Working with Font Formatting Commands*
- ~ *Changing the Background Color of a Cell*
- ~ *Adding Borders to Cells*
- ~ *Excel Cell Borders Continued*
- ~ *Formatting Data as Currency Values*
- ~ *Formatting Percentages*
- ~ *Using Excel's Format Painter*
- ~ *Creating Styles to Format Data*
- ~ *Merging and Centering Cells*
- ~ *Using Conditional Formatting*
- ~ *Editing Excel Conditional Formatting*

# Text To Speech

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP PROJECT

Course link : <https://ineuron.ai/course/Text-To-Speech>

## Course Description :-

This is a Python project for converting text to speech in English with various accents. We will create a script that will make use of the gTTS library. The code will be written in a modular way, with a good user end point application.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access

## What you will learn :-

- => Real Time Projects
- => Text To Speech
- => Implementation Python oops concepts.
- => Modular Coding
- => gTTS
- => Flask

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Rishav Dash :

~ This is Rishav Dash. I am a Jr. Data Scientist and mentor at INeuron.ai with working experience in computer vision, natural language processing, Machine Learning, and Alops. Hands-on experience leveraging machine learning, deep learning, transfer learning models to challenging real-world problems, and building products to solve peoples problems.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- Text To Speech :
  - ~ Introduction of Instructor
  - ~ Project Overview
  - ~ End Notes
  - ~ Problem Description
  - ~ Understand the application scope
  - ~ Tour to existing solution
  - ~ End Notes
  - ~ Solution Description
  - ~ Notebook Walkthrough
  - ~ Cost involved
  - ~ End Notes
  - ~ Structure overview
  - ~ Utils
  - ~ Pipeline
  - ~ Frontend app design
  - ~ Docker
  - ~ Tour to the cloud and Service Overview
  - ~ Heroku
  - ~ Workflow
  - ~ Conclude the project
  - ~ Points to improve from current project
  - ~ Assignments & External Resources

# Python for Beginner

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON

Course link : <https://ineuron.ai/course/Python-for-Beginner>

## Course Description :-

The goal of this course is to educate everyone on the fundamentals of computer programming using Python. We'll go over the fundamentals of building a programme in Python from a set of simple instructions. There are no prerequisites for this course. Anyone with a basic understanding of computers should be able to grasp the valuable content in this course.

## Course Features :-

- => Quizzes
- => Assignments
- => Hands-on practicals
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Python installation and setup
- => Python basic concepts
- => OOPS concepts

## Requirements :-

- => A system with internet connection.
- => Your dedication

## Instructors :-

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Python :
  - ~ Python installation and setup
  - ~ Python basics
  - ~ While loop
  - ~ For loop
  - ~ String and list manipulation
  - ~ List manipulation
  - ~ Tuple, set, dictionary
  - ~ Function
  - ~ Logging and debugging
  - ~ Modules and exception
  - ~ Class and object
  - ~ Abstraction and inheritance

# Mathematics using Python

---

Topic Name : K12

Sub-topic Name : CLASS9

Course link : <https://ineuron.ai/course/Mathematics-using-Python>

## Course Description :-

Many people are unaware that Python is a fantastic tool for studying mathematics. Python may be used as a basic calculator, but did you know that it can also be used to master more difficult concepts in algebra, geometry, and matrix analysis? That's precisely what this course will teach you. This course introduces Python programming to students with the help of various mathematical concepts taught in schools. This course is designed in order to help students in understanding the different mathematical subjects and concepts with the help of practical programming and hands-on practice in the Python programming language.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with tech
- => Real-time project
- => Live class recording
- => Doubt clearing
- => Assignment in all the module
- => Quiz in every module
- => Career Counselling
- => Completion certificate

## What you will learn :-

- => Linear algebra
- => Vector operations
- => Matrix
- => Eigen Vectors and Eigen Values
- => Matrix Operations in Machine Learning

## Requirements :-

- => Interest to learn
- => Dedication
- => System with good internet connection

## Curriculum details :-

- => Introduction :
  - ~ Course introduction
  - ~ Who is this course for?
  - ~ Course overview
  - ~ Course outcome
- => Linear Algebra :
  - ~ Introduction to Linear Algebra
  - ~ Vectors
  - ~ Matrices
  - ~ Tensors
- => Vector Operations :
  - ~ Transposition and Norm of a Vector
  - ~ Dot Product
  - ~ Orthogonal Vectors
  - ~ Projection of Vectors
  - ~ Line, Plane and Hyperplane
- => Matrix :
  - ~ Transposition of Matrix
  - ~ Arithmetic Operation
  - ~ Hadamard Operations and Reduction of Matrix
  - ~ Hands-on Code demo with Python
  - ~ Solving system of Linear Equations
  - ~ Types of Solutions
  - ~ Plotting Equation
  - ~ Hands-on Plotting equations
  - ~ Matrix Norms and Properties



- ~ *Linear Transformation*
- ~ *Matrix Multiplication*
- ~ *Matrix Inversion*
- ~ *Identity Matrix*
- ~ *Diagonal Matrix*
- ~ *Symmetric Matrix*
- ~ *Determinant of a Matrix*

=> Eigen Vectors and Eigen Values :

- ~ *Eigen Vectors and Eigen Values*
- ~ *Properties of Eigen Values*
  
- ~ *Properties of Eigen Values*

=> Matrix Operations in Machine Learning :

- ~ *Affine Transforamtions*
- ~ *Singular Vector Decomposition*
- ~ *Image Compression*
- ~ *Moore-Penrose Pseudoinverse*
- ~ *Application of Pseudoinverse*
- ~ *Principle Component Analysis*

# End to End Object Detection

---

Topic Name : DATA SCIENCE

Sub-topic Name : COMPUTER VISION

Course link : <https://ineuron.ai/course/End-to-End-Object-Detection>

## Course Description :-

Become an Object Detection Guru with the latest frameworks available like Tensorflow, Detectron2 and Yolo. In this course you will be learning to create four different object detector using multiple frameworks from scratch.

## Course Features :-

- => Lifetime Dashboard Access
- => Certificate
- => End to End Project
- => Self paced classes

## What you will learn :-

- => Python Basics
- => Flask Development
- => Pycharm Basics
- => Debug Applications
- => Tensorflow1.x Object Detection
- => Tensorflow2.x Object Detection
- => Detectro2 Object Detection/Segmentation
- => Yolo Object Detection
- => Working with Images
- => Working with Videos

## Requirements :-

- => Computer with Internet Connectivity
- => Basic Python Knowledge
- => 8GB RAM preferred
- => Intel Core i5 preferred
- => Windows/Linux/MAC Preferred

## Instructors :-

=> Sourangshu Pal :

~ Visual Computing Engineer and instructor at iNeuron.ai having 3 years of diverse experience in the discipline of visual computing with specialization in Deep Learning and Computer Graphics. Loves to analyze, process, and model visual data then interpret the insights to create actionable plans for solving challenging business problems.

## Curriculum details :-

=> Introduction to Course :

- ~ Introduction to Course Preview
- ~ Who is this Course for? Preview
- ~ Course Overview
- ~ Course Outcome
- ~ Installing Anaconda, Pycharm & Postman
- ~ Working with Conda Envs
- ~ Pycharm Introduction
- ~ Pycharm with Conda
- ~ Pycharm with venv
- ~ Pycharm with Pipenv

=> Covering Python Basics :

- ~ Introduction
- ~ Building a Calculator
- ~ Working with Command Line Arguments
- ~ Building the Flask Application
- ~ Testing our App in POSTMAN
- ~ Learn to Debug with Pycharm
- ~ Adding an UI to our Web App

=> Understand Object detection theoretically :

- ~ Introduction
- ~ What is Object Detection?
- ~ What are Bounding Boxes?

- ~ Metrics used in Object Detection
- ~ Applications of Object Detection

=> Object Detection using Tensorflow 1.x :

- ~ Introduction
- ~ Introduction to TFOD1.x
- ~ Using Google Colab with Google Drive
- ~ Installation of Libraries in Colab
- ~ TFOD1.x Setup in Colab
- ~ Visiting the Model Zoo
- ~ Inferencing in Colab
- ~ Inferencing in Local
- ~ Important Configuration Files
- ~ Webcam Testing

=> Training a Custom Mask Detector using Tensorflow1.x :

- ~ Introduction
- ~ Our Custom Dataset
- ~ Doing Annotations or labeling data
- ~ Preparing the Dataset for Training
- ~ Selection of Pretrained Model from Model Zoo
- ~ Files Setup for Training
- ~ Let's start Training
- ~ Resume or Stop Training
- ~ Converting CKPT to Frozen Inference Graph
- ~ Inferencing with our trained model

=> Creating an End To End Mask Detector Web Application with TFOD1 :

- ~ Introduction
- ~ Creating a Pycharm project & Environment Setup
- ~ Debugging our Application
- ~ Testing our App with PoSTmaN
- ~ Adding an UI to our Web APP

=> Object Detection using Tensorflow 2.x :

- ~ Introduction
- ~ Introduction to TFOD2.x
- ~ Installation of Libraries in Colab
- ~ Visting TFOD2.x Model Garden
- ~ Inference using Pretrained Model
- ~ Important Configuration Files
- ~ Inferencing in Local with a pretrained model

=> Training a Custom Chess Piece Detector using Tensorflow2 :

- ~ Introduction
- ~ Our Custom Dataset
- ~ Doing Annotations or labeling data
- ~ Preparing the Dataset for Training
- ~ Selection of Pretrained Model from Model Zoo
- ~ File Setup for Training
- ~ Let's start Training
- ~ Stop Training or resume Training
- ~ Convert CKPT to Saved Model
- ~ Inferencing using the Custom Trained Model in Colab
- ~ Inferencing using the Custom Trained Model in Local PC

=> Creating an End To End Chess Piece Detector Web Application with TFOD2 :

- ~ Introduction
- ~ Creating a Pycharm project & Environment Setup
- ~ Building a Flask Application
- ~ Debugging our Application
- ~ Testing our App with PoSTmaN
- ~ Adding an UI to our Web APP

=> Object Detection using Detectron2 :

- ~ Introduction
- ~ Introduction to Detectron2
- ~ Installing libraries in Google Colab
- ~ Visiting the Model Zoo
- ~ Inferencing using Pre Trained Model

=> Training a Custom Detector using Detectron2 :

- ~ Introduction
- ~ Our Custom Dataset
- ~ Doing Annotations or labeling data
- ~ Registering Dataset for Training
- ~ Selection of Pretrained Model from Model Zoo
- ~ Let's start Training
- ~ Stop Training or resume Training
- ~ Inferencing using the Custom Trained Model in Colab
- ~ Evaluating the Model

=> Creating an End To End Custom Detector Web Application with Detectron2 :

- ~ Introduction
- ~ Creating a Pycharm project & Environment Setup
- ~ Building a Flask Application
- ~ Debugging our Application
- ~ Testing our App with PoSTmaN
- ~ Adding an UI to our Web APP

=> Object Detection using YoloV5 :

- ~ Introduction

- ~ *Introduction to YoloV5*
- ~ *Inferencing using Pre Trained Model*

=> Training a Custom Warehouse Apparel Detector using YoloV5 :

- ~ *Introduction*
- ~ *Our Custom Dataset*
- ~ *Doing Annotations or labeling data*
- ~ *Preparing the Dataset for Training*
- ~ *Let's start Training*
- ~ *Inferencing using the Custom Trained Model in Colab*

=> Creating an End To End Warehouse Apparel Detector Web Application with YOLOV5 :

- ~ *Introduction*
- ~ *Creating a Pycharm project & Environment Setup*
- ~ *Building a Flask Application*
- ~ *Debugging our Application*
- ~ *Testing our App with PoSTmaN*
- ~ *Adding an UI to our Web APP*

### Project details :-

=> Mask detector

=> Chess Piece detector

=> Mixed Classes detector

=> Warehouse Apparel detector

# Tech Awareness

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Tech-Awareness>

## Course Description :-

We will teach you about latest technology changes, news, how-tos and more.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => New technology
- => Social Media Awareness
- => Technologies Roles in Career

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Shivan Kumar :
  - ~ Associate Data Scientist, Mentor, and Kaggle Master with 2 Years of Experience. Experience in building models that translate data points into business insights. Highly accurate at collecting, analyzing, and interpreting large datasets, developing machine learning, deep learning, Chatbots models, and deploying the solutions on the cloud. I also love to participate in Hackathons. In my free time, I like to mentor students to learn Data Science and achieve their goals.

## Curriculum details :-

- => Course Introduction :
  - ~ Welcome to Tech Awareness course
  - ~ What you will learn from this course
  - ~ Course pre-requisites
  - ~ Why tech awareness is important?
  - ~ Who is this course for?
  - ~ What you will get from this course?
  - ~ How to get access to course materials?
  - ~ What career path you can follow after completion of this course?
- => Tech Awareness :
  - ~ Introduction to Technology
  - ~ Digital Technology vs Traditional Technology
  - ~ Impact of Technology on Kids
  - ~ Internet Technology
  - ~ Social Media Awareness
  - ~ Negative Impact vs Positive Impact
  - ~ When and How to Set Technology Limits for Kids
  - ~ A balanced approach to use technology
  - ~ Technology's advantages and drawbacks
  - ~ Technologies Roles in Career
  - ~ Why Programming came into existence?

# MongoDB Course

---

Topic Name : DATABASE

Sub-topic Name : MONGODB

Course link : <https://ineuron.ai/course/MongoDB-Course>

## Course Description :-

This course is designed for database administrators, database architects, software developers, software architects, database professionals, project managers, IT developers, testers, analytics professionals, research professionals, and system administrators who want to work with NoSQL databases and MongoDB.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => What is mongoDB
- => How does mongoDB works
- => What is mocha and need of mocha in mongodb
- => Big umbrella of MongoDB
- => How to install mongoDB on MAC
- => How to install mongoDB on Windows
- => Create and Read operation in MongoDB
- => ObjectId and BSON in mongoDB
- => Triple A and CRUD operations in mongoDB
- => UpdateOne and DeleteOne in #mongoDB
- => UpdateMany and deleteMany in mongoDB
- => Database issues with Update in mongodb

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> MongoDB :

- ~ What is mongoDB
- ~ How does mongoDB works
- ~ What is mocha and need of mocha in mongodb
- ~ Big umbrella of MongoDB
- ~ How to install mongoDB on MAC
- ~ How to install mongoDB on Windows
- ~ Create and Read operation in MongoDB
- ~ ObjectId and BSON in mongoDB
- ~ Triple A and CRUD operations in mongoDB
- ~ UpdateOne and DeleteOne in #mongoDB
- ~ UpdateMany and deleteMany in mongoDB
- ~ Database issues with Update in mongodb
- ~ Getting more data in #mongodb
- ~ Save bandwidth while querying in mongoDB
- ~ Understanding objects structure in mongoDB
- ~ Understanding Arrays in mongoDB
- ~ What is schema in mongoDB
- ~ Database modeling mongoDB series

- ~ *Relation in database mongoDB*
- ~ *One to one relation in mongo database with id*
- ~ *One to many relation in mongo database*
- ~ *Many to Many relation in mongoDB*
- ~ *Exploring mocha, mongo and mongoose*
- ~ *Creating file structure and installing dependencies*
- ~ *Creating our first student schema*
- ~ *Connecting with mongoddb with mongoose*
- ~ *BlueBird and Q promises*
- ~ *Hooks in mocha*
- ~ *Describe and it blocks for mocha test*
- ~ *Your first create test using mocha*
- ~ *Using beforeEach in mocha tests*
- ~ *A read test in mongoDB*
- ~ *A delete test for mongoDB*
- ~ *An update test in mongoDB*

# Cyber Security Masters Tech Neuron

---

Topic Name : CYBER SECURITY

Sub-topic Name : CYBERSECURITY MASTERS

Course link : <https://ineuron.ai/course/Cyber-Security-Masters-Tech-Neuron>

## Course Description :-

One of the most famous FAQ on google is How to get started with Ethical hacking? The perfect way to find the correct opportunity according to your potential is to take a deep dive into a course that has a great variety with a lot of practical practice. This is an ISO 27001 government certified course that covers Bug Bounty & Web Pentesting, system exploitation, network pentesting, android pentesting, Forensics etc. From this course, our students have found bugs in big organizations like Harvard, Huawei, Nike, BMW, Doordarshan, Nykaa, GoodRx etc.

## Course Features :-

- => ISO 27001 Certified Cyber Security Master's Certification
- => On-site Internship for Top Scholars
- => 1:1 Personalised Mentorship
- => Revision Classes
- => Live Teaching by Instructors
- => Industry level Testing walkthrough
- => Every week doubt clearing sessions
- => Doubt Clearing through mail and skype support team
- => Assignment and Quizes in all the modules
- => Resume Building
- => Career Guidance
- => Interview Preparation
- => Job Fair & Internal Hiring

## What you will learn :-

- => How to make money from Bug Bounty?
- => Hand on Kali Linux
- => Linux Fundamentals Crash Course
- => Complete Burpsuite module training
- => Bug Bounty
- => Web Pentesting
- => Owasp top 10
- => 30+ Latest Web Attacks
- => OSI Model
- => Networking Fundamental
- => Wireshark
- => Nmap
- => System Security
- => Exploit and Malwares
- => Forensics
- => Man in the middle attack
- => Anonymity
- => How to crack cyber security interviews

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication
- => I3 Processor or better
- => At least 4GB Ram



## Curriculum details :-

=> Welcome to the world of Penetration Testing :

- ~ Course Introduction
- ~ Why Web Penetration Testing?
- ~ Types of Hackers
- ~ Disclaimer for this course
- ~ What is Vulnerability?
- ~ What is VAPT?
- ~ What is Owasp top 10
- ~ Scope & Duties of Web Pentester in InfoSec Companies
- ~ Goals for Resume Building in Web Pentesting
- ~ How much and where can I make money from bug bounty?

=> Setting up an environment :

- ~ Things to cover in this section
- ~ Grabbing Required downloadable resources for this section
- ~ Learning Virtualization with Virtual Box
- ~ Setting up & Walkthrough of Vbox modules
- ~ Introduction & History of Linux
- ~ Why Kali Linux?
- ~ Installing Kali Linux
- ~ Tweaks to Run Kali Linux Smoothly Part 1
- ~ Tweaks to Run Kali Linux Smoothly Part 2
- ~ Updating and Upgrading Kali Linux with Debian packages

=> Linux Fundamentals Crash Course :

- ~ Introduction to command prompt
- ~ Accessing system & Network Related commands
- ~ Ip Config & Bridge network
- ~ Linux file system and Structure
- ~ Introduction to root
- ~ Absolute and relative paths
- ~ Directory listing attributes
- ~ Playing with file and directories
- ~ Different file types in Linux
- ~ Wildcard commands
- ~ Understanding files and Directory permissions
- ~ File permission commands
- ~ Help commands, auto completion and arrow keys
- ~ Piping process
- ~ Linux file editors
- ~ Switching user with sudo module
- ~ System utility commands (Date, Uptime, Hostname, Uname etc)
- ~ Installing softwares
- ~ Github clone to run tools
- ~ Compiling python
- ~ Compiling java

=> Core fundamentals for web pentesting :

- ~ What is an Ip address?
- ~ What is protocol? HTTP & HTTPS
- ~ Subdomain & Domain name
- ~ What are ports?
- ~ Path & Query component in URL
- ~ Parameters and Fragment
- ~ Explaining verbs, What is GET method?
- ~ What is Post Method?
- ~ What is Put Method?
- ~ Delete & Head Method
- ~ Connect & Options
- ~ Trace & Patch
- ~ How does an API works?
- ~ HTTPS Status code part 1
- ~ HTTPS Status code part 2

=> Complete Burpsuite module training :

- ~ What is Burp Suite?
- ~ Burp CA Certificate for SSL/TLS
- ~ Burp Project Type : New, Existing & Temp
- ~ Burp Suite Proxy
- ~ Burpsuite Intruder
- ~ Burpsuite Scanner
- ~ Burp suite Repeater
- ~ Burp Suite Sequencer
- ~ Burp Suite Decoder
- ~ Burp Suite Comparer
- ~ What are Payloads? Simple List, Runtime file, Custom iterator
- ~ Payload type : Character Substitution, Case Modification, Recursive grep
- ~ Payload Type : Illegal Unicode, Character Blocks, Numbers
- ~ Payload Types : Dates, Brute Forcer, Null Payloads, Character Frobber
- ~ Payload types : Bit Flipper, Username Generator, ECB Block Shuffler
- ~ Burp Suite Extender
- ~ Burp Suite Extensions
- ~ BApp Store
- ~ Burp Suite APIs
- ~ Burp Suite Options
- ~ Engagement Tools
- ~ Http History Analyser

- ~ *Connect Burp to Android for Testing Android Apps*
- => **Reconnaissance Methodology :**
  - ~ *DNS Records with Virustotal*
  - ~ *HTTP Status Recon*
  - ~ *Subdomain enumeration*
  - ~ *Aquatone*
  - ~ *Shodan Research*
  - ~ *Directory Bruteforcing*
  - ~ *Digging into the past with WayBack Machine*
  - ~ *Certificate Transparency Crt*
  - ~ *Wappalyzer for Technology Identification*
  - ~ *Netcraft Active Cyber Defence*
- => **Getting started with Testing environment :**
  - ~ *What is DVWA?*
  - ~ *Getting started by Creating Database & User for lab*
  - ~ *Configuring DVWA*
  - ~ *DVWA Error Solving*
- => **Brute force & Command Injection :**
  - ~ *Brute force technique part 1*
  - ~ *Brute force technique part 2*
  - ~ *What is Command Injection & CI Low level breach*
  - ~ *Command Injection: Breaching Medium Level Security*
  - ~ *Command Injection: Breaching High Level Security*
  - ~ *Command Injection Mitigation & Secure Code writing logic*
  - ~ *Remote Code Execution Incident Report Study*
- => **Insecure Session Management & Cookie Manipulating Flaw :**
  - ~ *Insecure Session Management & Cookie Manipulating Flaw*
  - ~ *Insecure JSON Parsing*
- => **Cross Site Request Forgery :**
  - ~ *What is Cross Site Request Forgery? CSRF Part 1*
  - ~ *CSRF: Part 2 (Designing Custom CSRF Form)*
  - ~ *CSRF: Execution of Custom form and Mitigation Technique*
  - ~ *CSRF: Automated form via Burpsuite*
  - ~ *CSRF Incident Report Study*
- => **File Upload Vulnerability :**
  - ~ *What is File Upload Vulnerability? Breaching Low Level*
  - ~ *Breaching Medium Level*
  - ~ *Breaching High Level & Mitigation*
  - ~ *File Upload Incident Report Study*
- => **File Inclusion Vulnerability :**
  - ~ *Local & Remote File Inclusion (Low Level)*
  - ~ *LFI & RFI (Medium & High Level)*
  - ~ *LFI & RFI Incident Report Study*
- => **SQL Injection :**
  - ~ *SQL Injection Master Lab & What is Database?*
  - ~ *SQL Fundamentals*
  - ~ *What is ID, Joining & Breaking the query in SQL*
  - ~ *Selecting Vulnerable Column & Fetching Database Name*
  - ~ *Dumping Database*
- => **Boolean Based & SQL Automation :**
  - ~ *Boolean Based Queries & Fundamentals*
  - ~ *Boolean Based demonstration*
  - ~ *Automation With SQL Map*
- => **Cross Site Scripting :**
  - ~ *Reflecting XSS*
  - ~ *Stored XSS*
  - ~ *Dom Bases XSS*
  - ~ *Exploring Innovative method for executing XSS via Case Studies*
- => **Increasing Difficulty with WebGoat :**
  - ~ *Gathering Pre-Requisites for Webgoat*
  - ~ *Configuring Webgoat in Windows*
- => **Token Exploitation :**
  - ~ *What is JSON Web Token? (JWT)*
  - ~ *JWT : JSON Web Token Hijacking with SQL Injection*
  - ~ *JWT Payment Gateway Manipulation*
- => **Password Reset EndPoint :**
  - ~ *Password Reset Endpoint*
  - ~ *Creating and Exploiting Password Reset Link*
- => **Path Traversal :**
  - ~ *Path Traversal - Bypass File Upload Fix 1*
  - ~ *Path Traversal - Bypass File Upload Fix 2*
  - ~ *Path Traversal - Retrieving Files*
- => **SQL String Based :**
  - ~ *String SQL Injection Part 1*
  - ~ *String SQL Injection Part 2*
  - ~ *Delete Data & Retrieve Data from Tables*
  - ~ *SQL Login Attack*
- => **HTML Tempering & XXE :**

- ~ *HTML Tampering explained with Execution*
- ~ *XXE : What is XXE Injection?*
- ~ *XXE Injection Content Type Manipulation*
- ~ *Blind XXE Injection*

=> **Insecure Direct Object Reference :**

- ~ *What is IDOR?*
- ~ *Data Extraction via IDOR*
- ~ *Account Hijacking via IDOR*

=> **Advance CSRF & SSRF :**

- ~ *Login CSRF*
- ~ *SSRF Explained*
- ~ *SSRF - Request Manipulation to display User*

=> **Bonus Attacks :**

- ~ *Vulnerable Components - Exploiting CVE*
- ~ *Meta Data Sanitization*
- ~ *Client-Side Filtering*

=> **Wireshark :**

- ~ *OSI Model Layer*
- ~ *Split of Concentration*
- ~ *Application layer*
- ~ *Presentation Layer*
- ~ *Session layer*
- ~ *Top Layer vs*
- ~ *Transport Layer*
- ~ *Network Layer*
- ~ *Data link Layer*
- ~ *Physical Layer*
- ~ *Host Communication*
- ~ *Encapsulation*
- ~ *TCP/IP vs OSI Model*
- ~ *Wireshark Filters & Data Capture*

=> **Nmap :**

- ~ *Nmap Basics, Target Specification & Port States*
- ~ *Nmap Scanning & Ping Scanning*
- ~ *Nmap Scan Techniques with SYN, Connect, UDP, SCTP, TCP, ACK & Window*
- ~ *Nmap Scan Techniques Part 2 : Null, Fin, XMAS, Maimon, IDLE Scan & IP Protocol*
- ~ *Nmap Performance, Firewall & IDS Evasion*

=> **Exploits :**

- ~ *What is metasploit?*
- ~ *How port scanning can help us in exploiting machines?*
- ~ *How to Configure Exploits?*
- ~ *Executing Eternal Blue exploit on Windows Machine*
- ~ *Microsoft Windows 10 (1903/1909) - 'SMBGhost' SMB3*
- ~ *Microsoft Windows 10 (1903/1909) - 'SMBGhost' SMB3*

=> **Forensics :**

- ~ *Analysis - Registry, Email and Browser Artifacts*
- ~ *Analysis - PDF Files and Page Files*
- ~ *Malware File Analysis*
- ~ *USB Forensics - Detection and Investigation*
- ~ *Meta Data Analysis - MS Office Files*
- ~ *Meta Data Analysis - Image Files*
- ~ *Memory Forensics using FTK Imager and Volatility3 tool overview*
- ~ *Volatility3 - Memory File Analysis and Infected system file*

=> **Final Module :**

- ~ *Pentsting with Automated tools : Owasp Zap*
- ~ *Httrack & Wpscan*
- ~ *What is Accunetix?*
- ~ *Accunetix Practical Scanning*
- ~ *How to Make POC (Proof of Concept)*
- ~ *How to make a VAPT (Vulnerability Assessment & Penetration Testing Report) report : Part 1*
- ~ *VAPT Part 2*
- ~ *How to get Job Ready and CV guide*
- ~ *What to learn next?*
- ~ *Final Closure*

=> **Extra Module :**

- ~ *Maintaining Anonymity with Tor Browser & Dark Net Walkthrough*
- ~ *Proxychaining & Masking Mac Address*
- ~ *Man in the middle attacks (MITM) via ARP Spoofing*
- ~ *Phone Hacking*

=> **Interview Prep :**

- ~ *Mock Interview: Level 1*
- ~ *Mock Test Paper (Practical Skill Based): Level 2*
- ~ *Group Discussion Round: Level 3*
- ~ *Resume Building*

=> **Examination :**

- ~ *Final Examination*

# Deep Learning With Computer Vision and Advanced NLP

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/Deep-Learning-With-Computer-Vision-and-Advanced-NLP>

## Course Description :-

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks. It is a function that imitates the workings of the human brain in processing data and creating patterns for use in decision making. Learn Deep Learning, Transfer Learning and Neural Networks using the latest frameworks. Become a Deep Learning Guru!

## Course Features :-

- => Deep Learning
- => Natural Language processing
- => Computer Vision
- => Course Certificate
- => One to One Resume Discussion
- => Doubt Clearing session
- => Email Support
- => All 7 Days in a week Skype Support
- => Career Guidance

## What you will learn :-

- => Advance NLP with deep-learning overview.
- => TensorFlow Installation.
- => Pytorch.
- => Neural Network.
- => CNN overview
- => Advance Computer Vision Part 1.
- => Advance computer Vision Part 2.
- => ChatBot.
- => Text processing
- => Spacy.
- => NLP terminology.
- => RNN
- => Attention Based model.
- => Hardware Setup GPU.
- => Transfer Learning in NLP.
- => Mini NLP Project.
- => Deployment of Model and Performance tuning.
- => NLP Transfer learning project with deployment and integration with UI.
- => NLP end to end project with architecture and deployment.
- => NLP project end to end with deployment in various cloud and UI integration.
- => Computer Vision Project.

## Requirements :-

- => Dedication
- => Computer with i3 processor and internet

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

=> Introduction

=> Advance NLP with deep-learning overview :

- ~ *Computational Linguistic*
- ~ *History of NLP*
- ~ *Why NLP*
- ~ *Use of NLP*

=> TensorFlow Installation :

- ~ *Tensorflow Installation 2.0*
- ~ *Tensorflow Installation 1.6 with virtual environment*
- ~ *TensorFlow 2.0 function*
- ~ *Tensorflow 2.0 neural network creation*
- ~ *Tensorflow 1.6 functions*
- ~ *Tensorflow 1.6 neural network and its functions*
- ~ *Keras Introduction*
- ~ *Keras in-depth with neural network creation*
- ~ *Mini project in Tensorflow*

=> Pytorch :

- ~ *Pytorch installation*
- ~ *Pyrotorch functional overview*
- ~ *Pytorch neural network creation*

=> Neural Network :

- ~ *A Simple Perception Preview*
- ~ *Neural Network overview and its use case Preview*
- ~ *Various Neural Network architect overview*
- ~ *Use case of Neural Network in NLP and computer vision*
- ~ *Multilayer Network*
- ~ *Loss Functions*
- ~ *The Learning Mechanism*
- ~ *Optimizers*
- ~ *Forward and Backward Propagation*
- ~ *Gradient Descent*

=> CNN overview :

- ~ *CNN definition and various CNN based architecture*
- ~ *End to End CNN network training*
- ~ *Deployment in Azure*
- ~ *Cloud performance tuning of CNN network*

=> Advance Computer Vision Part 1. :

- ~ *GAN*
- ~ *Generative Model Using GAN*
- ~ *BERT*
- ~ *Semi-Supervised learning using GAN*
- ~ *Restricted Boltzmann Machine (RBM) and Autocoders*
- ~ *CNN Architectures*
- ~ *LeNet-5*
- ~ *AlexNet*
- ~ *GoogleNet*
- ~ *VGGNet*
- ~ *ResNet*
- ~ *SSD*
- ~ *SSD lite*
- ~ *Faster R CNN*

=> Advance computer Vision Part 2. :

- ~ *SCNN*
- ~ *Masked R-CNN*
- ~ *Xception*
- ~ *SENet*
- ~ *Facenet*
- ~ *Implementing a ResNet 34 CNN using Keras*
- ~ *Pretrained Models from Keras*
- ~ *Pretrained Models for Transfer Learning*

=> ChatBot :

- ~ *Intents and Entities*
- ~ *Fulfillment and integration*
- ~ *Chatbot using Microsoft bot builder and LUIS, development to Telegram, Skype*
- ~ *Chatbot using Microsoft bot builder and LUIS, development to Telegram, Skype*
- ~ *Chatbot using Amazon Lex, deployment to Telegram, Skype*
- ~ *Chatbot using RASA NLU, deployment to Telegram , Skype*
- ~ *Semantic Segmentation*
- ~ *Classification and Localisation*
- ~ *TensorFlow Object Detection*
- ~ *You Only Look Once (YOLO)*

=> Text processing :

- ~ *Importing Text*
- ~ *Web Scrapping*
- ~ *Text Processing*
- ~ *Understanding Regex*
- ~ *Text Normalisation*
- ~ *Word Count*
- ~ *Frequency Distribution*
- ~ *Text Annotation*

- ~ Use of Anotator
- ~ String Tokenization
- ~ Annotator Creation
- ~ Sentence processing
- ~ Lemmatization in text processing
- ~ POS
- ~ Named Entity Recognition
- ~ Dependency Parsing in text
- ~ Sentimental Analysis

=> Spacy :

- ~ Spacy Overview
- ~ Spacy function
- ~ Spacy function implementation in text processing
- ~ POS tagging, challenges and accuracy
- ~ Entities and named entity Recognition, interpolation, Language models

=> NLP terminology :

- ~ Morphology and Diversity
- ~ Ambiguity and Paradigms
- ~ Structures and meanings
- ~ Lexical Knowledge, Network Metaphors and co-references
- ~ Lexical Ambiguity
- ~ Polysemy and homonymy
- ~ Coreference Resolution
- ~ Anaphora and cataphora resolution
- ~ Multi-sentential resolution
- ~ Humans and Ambiguity
- ~ Machines and ambiguity
- ~ Co-occurrence and distributional similarity
- ~ Similarity and relatedness
- ~ Knowledge graphs and repositories
- ~ Computational Linguistics
- ~ Word embeddings and co-occurrence vectors
- ~ Word Sim353 Dataset examples
- ~ Word2vec
- ~ Part of speech tagging

=> RNN :

- ~ Recurrent Neural Networks
- ~ Long Short Term Memory (LSTM)
- ~ Bi LSTM
- ~ GRU implementation
- ~ Building a Story writer using character level RNN

=> Attention Based model :

- ~ Seq 2 Seq
- ~ Encoders and Decoders
- ~ Attention Mechanism
- ~ Attention Neural Networks
- ~ Self Attention

=> Hardware Setup GPU :

- ~ GPU Introduction
- ~ Various type of GPU configuration
- ~ GPU provider and its pricing
- ~ Paperspace GPU setup
- ~ Running model in GPU

=> Transfer Learning in NLP :

- ~ Introduction to transformers
- ~ BERT Model
- ~ ELMo Model
- ~ GPT1 Model.
- ~ GPT2 Model
- ~ ALBERT Model
- ~ DistilBERT Model

## Project details :-

=> NLP project end to end with deployment in various cloud and UI integration :

- ~ Topic Modeling
- ~ Word sense disambiguation
- ~ Text to speech
- ~ Keyword Spotting
- ~ Document Ranking
- ~ Text Search (with Synonyms)
- ~ Language Modeling
- ~ Spam Detector
- ~ Image Captioning

=> Mini NLP project :

- ~ Machine Translation
- ~ Abstractive text summarization
- ~ Keyword spotting
- ~ Language modelling
- ~ Document summarization

=> Deployment of model and performance tuning :

- ~ Deep learning model deployment strategies
- ~ Deep learning project architecture

- ~ Deep learning model deployment phase
- ~ Deep learning model retraining phase
- ~ Deep learning model deployment in aws
- ~ Deep learning model deployment in azure
- ~ Deep learning model deployment in gcloud

=> Nlp transfer learning project :

- ~ Deployment and integration with ui machine translation
- ~ Question answering (like chat bot)
- ~ Sentiment analysis imdb
- ~ Text search (with synonyms)
- ~ Text classifications
- ~ Spelling corrector
- ~ Entity (person, place or brand) recognition
- ~ Text summarization
- ~ Text similarity (paraphrase)
- ~ Topic detection
- ~ Language identification
- ~ Document ranking
- ~ Fake news detection
- ~ Plagiarism checker
- ~ Text summarization extractive
- ~ Text summarization abstractive

=> NLP end to end project with architecture and deployment :

- ~ Movie review using bert
- ~ Ner using bert
- ~ Pos bert
- ~ Text generation gpt 2
- ~ Text summarization xlnet
- ~ Abstract bert
- ~ Machine Translation
- ~ Nlp text summarization custom
- ~ Keras/tensorflow
- ~ Language identification
- ~ Text classification using fast bert
- ~ Neuralcore
- ~ Detecting fake text using gltr with bert and gpt2
- ~ Fake news detector using gpt2
- ~ Python plagiarism checker type a message
- ~ Question answering

=> Computer Vision Project :

- ~ Traffic Surveillance System
- ~ Object identification
- ~ Object tracking
- ~ Object classification
- ~ Tensorflow object detection
- ~ Image to text processing
- ~ Speech to speech analysis
- ~ Vision based attendance system

# Reinforcement Learning

---

Topic Name : DATA SCIENCE

Sub-topic Name : REINFORCEMENT LEARNING

Course link : <https://ineuron.ai/course/Reinforcement-Learning>

## Course Description :-

You will learn the basics of reinforcement learning through this course. Adaptive learning systems are needed for artificial intelligence to reach its full potential. Implementing a complete RL solution will teach you how Reinforcement Learning (RL) solutions assist in solving real-world problems through trial-and-error interaction.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Basics of Reinforcement Learning
- => Foundations of Reinforcement Learning
- => OpenAI GYM Cartpole Experiment

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Sunny Bhaveen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

- => Introduction to Reinforcement Learning :
  - ~ Introduction to the course
  - ~ RL Introduction Part 1
  - ~ RL Introduction Part 2
  - ~ OpenAI GYM Cartpole Experiment



# Full Stack Django and React

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Full-Stack-Django-and-React>

## Course Description :-

ReactJS makes it simple to create dynamic and interactive online apps by providing elegant answers to some of front-end programming's most vexing problems. It is quick, scalable, adaptable, and powerful, and it has a thriving development community that is expanding quickly. The time is now to become familiar with React.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Django and Django REST framework
- => Django tour - quick
- => Django admin and migration
- => Storing our images in django
- => Setting Django API structure
- => Handling API root route in django
- => Image serialization and product views in django
- => Setting up URL for products in Django
- => Custom User serlization in Django
- => Generating the token in django
- => Custom user signin in Django
- => Creating base component
- => Bring in card component
- => Pulling up text info from backend
- => Conditional rendering of button

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with project :

- ~ Section 1 intro
- ~ Course Goals
- ~ Create a project structure from client requirements
- ~ How to handle such project
- ~ Django and Django REST framework

=> Installation for backend :

- ~ Section 2 intro
- ~ Python installation and virtual env
- ~ Learn to manage virtual environment
- ~ Django installation
- ~ Django tour - quick
- ~ Django admin and migration

=> Getting structure ready :

- ~ Section 3 intro
- ~ Cross origin requests in django
- ~ Django REST framework installation
- ~ Storing our images in django
- ~ Setting Django API structure
- ~ Handling API root route in django

=> Setting up categories in admin :

- ~ Section 4 intro
- ~ Setting up category model and admin
- ~ Serialize the data from DB in Django
- ~ Category API routing and views in Django
- ~ Testing with postman for category in Django

=> Setting up Products in admin :

- ~ Section 5 intro
- ~ Add model for Product in Django
- ~ Image serialization and product views in django
- ~ Setting up URL for products in Django

=> Setting up Custom User in admin :

- ~ Section 6 intro
- ~ Custom User model in admin in Django
- ~ Let 27s get some errors intentionally in django
- ~ Custom User serlization in Django
- ~ Generating the token in django
- ~ Custom user signin in Django
- ~ Signout user from django
- ~ Permissions template for you in django
- ~ Fixing bugs and urls for user in django
- ~ Classic super admin issue in django
- ~ Detailed testing with postman for user Django

=> Getting orders in admin Django :

- ~ Section 7 intro
- ~ Creating model for Orders in django
- ~ Add serialization for orders
- ~ Is user authentication Django
- ~ Adding order in admin in django
- ~ URL for order and auth token in Django

=> Payment Gateway backend :

- ~ Section 8 intro
- ~ Understand the payment gateway
- ~ Generate braintree token for user in Django
- ~ Proocess the payment from backend in django
- ~ Setup payment urls and debug in Django
- ~ giving django some test

=> Moving to front end :

- ~ Section 9 intro

=> React app for frontend :

- ~ Action plan for handling front end
- ~ Config for backend connection and structure
- ~ Application is up and running now
- ~ getting all products from backend
- ~ Getting all tshirt name in frontend
- ~ Image helper for card

=> Reusable card component :

- ~ Creating base component
- ~ Bring in card component
- ~ Pulling up text info from backend
- ~ Conditional rendering of button

=> Cart and auth helper :

- ~ Add item to cart helper
- ~ load and remove item from cart
- ~ degugging remove from cart
- ~ Signin and signup helper files
- ~ signout and authenticated user
- ~ Handling private routes in react

=> Signin and Signup compenents :

- ~ Load up the signup page in route
- ~ Create signup form template
- ~ Handle signup of user
- ~ Handling error and success message
- ~ User dashboard and private route debugging
- ~ Setup routes for signin
- ~ Getting the form ready for signin
- ~ Handle signin route in formdata
- ~ Handling error and success message

=> Navigation bar :

- ~ Get your nav bar here
- ~ Styling your navbar
- ~ Make user signout
- ~ Resolving signout bug and conditional rendering

=> Cart and purchase :

- ~ *Get the card ready for cart*
- ~ *Load all products and bugs*
- ~ *Force mounting of component*
- ~ *Order helper file*
- ~ *Payment helper file*
- ~ *Getting client token method*
- ~ *Load credit card handler*
- ~ *A long method for final purchase*
- ~ *Expecting bugs in this*
- ~ *A final debug walk through*

# React Native

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : REACT NATIVE

Course link : <https://ineuron.ai/course/React-Native>

## Course Description :-

There is no need to learn Java, Android Development, Swift programming, or anything else, all you need is React and JavaScript to build great native mobile applications including both Android and iOS. This is undoubtedly why Instagram, Uber, Skype, and plenty of other major brands use it to create mobile applications. You may join this league if you complete this course. You'll learn all there is to know about React Native's philosophy and fundamental principles, as well as how to create responsive designs that function on a variety of device sizes, how to navigate, and much more.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => React elements
- => React styles
- => Touchable and props
- => Handling images in React
- => Handling sound in React
- => Handling multiple users
- => Camera integration
- => Redux

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with React Native :

- ~ What is React native
- ~ Tools we will need
- ~ Installation of tools in MAC
- ~ First Hello World in Android emulator
- ~ How to install chocolatey in Windows
- ~ Android installation and setup on WINDOWS
- ~ Running your first react native app on Windows and AVD
- ~ Run app on real Android device

=> Getting started with First app :

- ~ Strategy for this app
- ~ What is metro
- ~ File structure of react native app
- ~ Get it done from scratch
- ~ 3 ways to style elements in react native
- ~ More styling of text
- ~ State in react Native
- ~ Touchable and props in react native

=> App to handle Images :

- ~ Strategy for dice roller app
- ~ Basic structure of dice roller
- ~ Writing CSS for dice Roller
- ~ Dice roller complete and 2 assignments

=> App to handle sound :

- ~ Strategy to handle Spanish number app
- ~ Adding third party code in react native
- ~ ScrollViews and sound import in react native
- ~ Unique keys for iterable in React Native
- ~ More on React native flexbox
- ~ Playing sound in react native app

=> App to handle user Input :

- ~ Getting started with currency App
- ~ Basic currency values
- ~ Formatting text input and result container
- ~ Buttons and keyboard dismiss
- ~ Handling error in input

=> Camera App - Profile Picture :

- ~ Permissions in Camera App
- ~ Having image in state
- ~ Click a photo
- ~ Load image from state

=> handle third party lib with Game app :

- ~ Getting started with native base
- ~ Installation and configuration of native base
- ~ Your first custom component
- ~ Having a plan of action for ticTacToe
- ~ Win message and reset game
- ~ Finishing game and debugging

=> Reactive native router and local storage :

- ~ Overview of Netflix store app
- ~ Going with all installation and screens
- ~ Getting navigation setup ready
- ~ Adding a FAB and moving to add screen
- ~ UI for add season list
- ~ Async Storage in react native
- ~ Home screen setup for seasons
- ~ React native hooks
- ~ Custom hooks in react native
- ~ Checked and delete seasons
- ~ Update and fix the bugs

=> Handling API and web requests :

- ~ Getting started with API
- ~ How to read API
- ~ Fetch a web request
- ~ Pass info to another component
- ~ making card for user

=> Redux in React Native :

- ~ Getting started with Redux
- ~ Redux Three Principles
- ~ action in REDUX
- ~ Reducer in REDUX
- ~ Store in REDUX
- ~ Provider in REDUX
- ~ Dispatch props in REDUX
- ~ state to props in redux

=> Instagram clone - Mega Section :

- ~ Instagram challenges in app
- ~ Having a firebase project
- ~ Connect firebase to RN project
- ~ Auth and Database basics in RN
- ~ Enable MultiDex and action
- ~ Signup a new user
- ~ Register a user in database
- ~ Signin and signout a user
- ~ Dispatch and database events
- ~ Handling all reducers
- ~ Lets create a lot of files
- ~ Store provider and new workflow
- ~ Ask Permission to user
- ~ Redux hooks and firebase subscribers
- ~ screen options to Custom header
- ~ Custom header component
- ~ Midway debugging session
- ~ Prep work for image upload
- ~ Upload image to firebase from mobile
- ~ Testing signup in firebase
- ~ Handle signin screen and testing
- ~ UI and config for addPost
- ~ Adding post to firebase
- ~ sending post to firebase and resolving bugs
- ~ Handling home page
- ~ Iron out bugs and prepare post component

~ Like and dislike feature

# Julia Programming

---

Topic Name : PROGRAMMING

Sub-topic Name : JULIA

Course link : <https://ineuron.ai/course/Julia-Programming>

## Course Description :-

Julia is a high-level, high-performance, dynamic programming language. Julia is also a general-purpose language it can be used to create applications, and many of its features are well suited for numerical analysis and computational science. Upon completion of this course, you will be able to perform Julia programming and you will be able to get a kickstart on how to use Julia for Data Science.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Julia Basics
- => Julia Data Science
- => Julia Projects

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Jaydeep Dixit :

~ Jaydeep Dixit is a data scientist and Blockchain Developer working at iNeuron having 1.5+ years of total experience. He specializes in Machine Learning and Blockchain. He has worked on various end-to-end projects in both machine learning and Blockchain. In addition to his primary job function, he has been recognized for his problem-solving skills.

## Curriculum details :-

=> Introduction :

- ~ Introduction
- ~ Who is this course for ?
- ~ Course prerequisite
- ~ What is Julia programming language ?
- ~ Julia vs Other Programming

=> Installation :

- ~ Installation

=> Julia Basics :

- ~ Variables
- ~ Integers Floating Point Numbers
- ~ Mathematical Operators and Elementary functions
- ~ Complex Numbers and Rational numbers
- ~ Strings
- ~ Functions
- ~ Compound Expression
- ~ Conditional Evaluation
- ~ Short Circuit Evaluation
- ~ Loops

=> Julia Data Science Basics :

- ~ Data Basics
- ~ Plotting
- ~ Julia Project

=> Summary :

- ~ Course Summary
- ~ Future Learning Path

# Graphics Design with PhotoShop

---

Topic Name : K12

Sub-topic Name : CLASS7

Course link : <https://ineuron.ai/course/Graphics-Design-with-PhotoShop>

## Course Description :-

In this course, you will learn everything you need to know about Photoshop. Photoshop is a very powerful tool that can do a lot of different things with images, including making them move. This means that you will be able to make posters as well as thumbnails and digital paintings. You can not only change existing photos or artworks, but you can also start from scratch and make new things.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => One to One Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to the course
- => Photoshop tools
- => Layers in Photoshop
- => Filters in Photoshop
- => Layer effects in Photoshop
- => Graphic Design works
- => Create Projects
- =>
- =>

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ Course Introduction
  - ~ Who is this course for?
  - ~ Course Overview & Course outcome
  - ~ Course Pre-requisite
  - ~ What is Graphic Designing?
  - ~ What is Photoshop?
  - ~ Why Photoshop?
- => Assignment1 :
  - ~ What do you know about Photoshop?
- => Installation :
  - ~ Installation of Adobe Photoshop
  - ~ Photoshop overview
- => Photoshop basics :
  - ~ How to create a new project in Photoshop?
  - ~ How to import images into Photoshop?
  - ~ Exploring the Photoshop Workspace and Customizing it.
  - ~ Write steps to create a project in Photoshop.
- => Photoshop tools :
  - ~ What is a move tool?
  - ~ What is a marquee tool?
  - ~ What is a lasso tool?



- ~ What is a selection tool?
- ~ What is a crop tool?
- ~ What is an eyedropper tool?
- ~ What is a healing tool?
- ~ What is a brush tool?
- ~ What is a clone tool?
- ~ What is the history brush tool?
- ~ What is an eraser tool?
- ~ What is a gradient tool?
- ~ What is a blur tool?
- ~ What is a dodger tool?
- ~ What is a type tool?
- ~ What is a pen tool?
- ~ What is direct / path selection tool?
- ~ What is a shape tool?
- ~ What is a hand tool?
- ~ What is a zoom tool?
- ~ What is a edit toolbar?
- ~ What is a selecting colour?

=> Assignment2 :

- ~ Write down 10 tools and their shortcut keys.

=> Layers :

- ~ What are layers?
- ~ How to create layers?
- ~ How to create a group for layers?
- ~ How to create a layer masking?
- ~ How to delete layers?
- ~ How to merge layers?
- ~ How to change opacity in the fill of layer?
- ~ How to turn off layers and mark with colors?
- ~ How to change the smart layer into the rasterized layer?
- ~ How to add layers style?

=> Assignment3 :

- ~ Write a step to create a new layer.

=> Assignment4 :

- ~ Write a step to merge layers in photoshop.

=> Assignment5 :

- ~ Write a step to create a layer masking.

=> Filter :

- ~ What is 3d?
- ~ What is a Blur?
- ~ What is Distort?
- ~ What is noise?
- ~ What is Pixelate?
- ~ What is sharpen?
- ~ What is stylize?

=> Assignment6 :

- ~ Write briefly about blur filters.

=> Assignment7 :

- ~ Write briefly about any file layer effects.

=> Layer effects :

- ~ What is Solid color?
- ~ What is the Gradient?
- ~ What is a Pattern?
- ~ What is Brightness/Contrast?
- ~ What are Levels?
- ~ What are Curves?
- ~ What is Exposure?
- ~ What is Vibrance?
- ~ What is Hue/Saturation?
- ~ What is the Color balance?
- ~ What is Black & White?
- ~ What is a Photo filter?
- ~ What is a Channel mixer?
- ~ What is Invert?
- ~ What is Posterize?
- ~ What is the threshold?
- ~ What is selective color?

=> Assignment8 :

- ~ Write briefly about any five-layer effects.

=> Graphic designing works :

- ~ How to create a Digital painting?
- ~ How to create a Matte painting?
- ~ How to create a Logo design?
- ~ How to create an icon?
- ~ How to create a thumbnail?
- ~ How to create posters?
- ~ How to create a banner?
- ~ How to create a business card?
- ~ How to create a brochure?
- ~ How to create social media posts?

- ~ *How to create sliding posts?*
- ~ *What is Typography?*
- ~ *How to import vectors from Adobe Illustrator into PS?*

=> Assignment9 :

- ~ *Create a new logo for iNeuron.*

=> Assignment10 :

- ~ *Create a thumbnail on your favorite topic and add your images.*

=> Final output :

- ~ *How to save projects in Photoshop?*
- ~ *How to save the project in PSD?*
- ~ *How to save the project in PNG?*
- ~ *How to save the project in JPEG?*
- ~ *How to save the project in PDF?*
- ~ *How to export projects in high quality?*

=> Assignment11 :

- ~ *Write a step to export your project in high quality in JPEG.*

=> Project :

- ~ *Poster Design*

=> Project Assignments :

- ~ *Digital Painting*

=> Course Summary :

- ~ *Course Outro*
- ~ *Future Learning Path*

# Android Programming with Machine Learning Apps

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : ANDROID

Course link : <https://ineuron.ai/course/Android-Programming-with-Machine-Learning-Apps>

## Course Description :-

Learning Android Development with Machine Learning will look great on any Android developer's CV. Machine Learning is a kind of Artificial Intelligence (AI) that allows the software to learn, explore, and predict outcomes without the need for human intervention. Machine learning has been employed in a variety of industries, and it is currently being actively used in the creation of mobile applications. Machine learning algorithms can analyse specific user activity patterns and respond to search queries with ideas and recommendations. This course will teach you how to use Android with Machine Learning .

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Android Studio fundamentals
- => Theme customization
- => Buttons and toasts
- => Fully customized Gradles
- => Android elements and components
- => SQLite database
- => JSON and APIs
- => Firebase
- => Machine Learning in Android
- => Various projects

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

- => Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to Android P development :
  - ~ Pep talk - Do not skip
  - ~ Tools that we will need
  - ~ Android History
- => Windows installation and setup :
  - ~ Installation of Android studio in WINDOWS
  - ~ AVD configuration and Hello world for WINDOWS
- => MAC setup and installation :
  - ~ Installation of Android - MAC
  - ~ Setting up Android Virtual device and config
- => Tour theme and App icons :
  - ~ Creating a project - API levels
  - ~ Exploring files in directory structure
  - ~ A tour of Android studio and customization - part 1
  - ~ A tour of Android studio and customization - part 2

- ~ Theme customisation and app on real device
- ~ Problems in App icon - Customization

#### => Buttons and toasts :

- ~ Button Customization
- ~ Click events for buttons
- ~ Assignment Solution
- ~ Methods and buttons
- ~ Basics of Toast and assignment
- ~ Shorter toasts

#### => Fully Customized Toasts and Gradles :

- ~ Basics setup for custom layouts
- ~ Preparing custom layouts
- ~ Customized layout inflation
- ~ Designing Elements in Linear layout
- ~ Gradle documentation
- ~ Final customization with gradle

#### => Components Tour of Android elements :

- ~ Components tour
- ~ Exploring text fields
- ~ Buttons and widgets in android
- ~ Understand layouts in Android

#### => Dice Roller app :

- ~ Designing assets for dice game
- ~ UI for DiceRoller
- ~ Writing code for diceRoller
- ~ Your assignment for this section

#### => Fun Background app :

- ~ Fun Background Design
- ~ Code part - fun background app

#### => Animated Login App :

- ~ Design assets for project AnimatedLogin
- ~ Applying animations in layout
- ~ Button Customization for app
- ~ Everything about button Customization
- ~ 1 more thing about buttons

#### => Truth Dare Game :

- ~ Setting up UI for Truth dare game
- ~ Code for Game and assignment

#### => Components of Android App :

- ~ Country Selector App - UI
- ~ Country Selector App - Code
- ~ Quick Change App
- ~ Burger Rating app - UI
- ~ Burger Rating app - code and assignment
- ~ Seekbar implementation
- ~ Uploader App UI
- ~ Uploader App Code with thread
- ~ Date Time picker in Android

#### => Currency Converter app :

- ~ Design of currency Converter app
- ~ Design of currency Converter app part 2
- ~ Handling Empty input and Assignment

#### => 3 Apps - Drumpad, examTimer, Music Player :

- ~ Going to a new screen

#### => 4 Apps - Drumpad, examTimer, Music Player :

- ~ Passing multiple values from intent

#### => 5 Apps - Drumpad, examTimer, Music Player :

- ~ MediaPlayer Class

#### => 6 Apps - Drumpad, examTimer, Music Player :

- ~ Setting layout for DrumPad App

#### => 7 Apps - Drumpad, examTimer, Music Player :

- ~ DrumApp code and assignment

#### => 8 Apps - Drumpad, examTimer, Music Player :

- ~ Exam Timer App design

#### => 9 Apps - Drumpad, examTimer, Music Player :

- ~ MediaPlayer App UI

#### => 10 Apps - Drumpad, examTimer, Music Player :

- ~ Exam Timer App code and sound

#### => 11 Apps - Drumpad, examTimer, Music Player :

- ~ Finishing Music Player app and Rockers

#### => Recycler and Card Views :

- ~ Recycler and Card Views Introduction
- ~ Custom layouts and getters
- ~ ArrayList for views
- ~ 10 Step guide for custom adapters
- ~ Main config for Insta cards

- ~ Refactoring the data
- ~ Add and remove Cards

=> SQLiteDatabase App - Student Record :

- ~ Introduction to database - UI setup
- ~ Database Helper introduction
- ~ Insert and Update data using helper
- ~ CRUD helper in Sqlite
- ~ Helper for showing messages
- ~ Adding data in sqlite
- ~ Getting data and handling cursor
- ~ Getting all data at once
- ~ Update and deletion of data

=> Jason and API apps :

- ~ What is API and JSON
- ~ Converting regular objects in JSON
- ~ Json to regular objects and Serialized name
- ~ Objects inside an object
- ~ Array in an object
- ~ Volley and API Introduction
- ~ Fetching an API request
- ~ Singleton in Volley

=> Firebase - Amazing Online database :

- ~ Section Intro
- ~ What is Firebase?
- ~ Exploring Firebase for Android
- ~ Setting layout for login system
- ~ User Registration System
- ~ User login & logout
- ~ Firebase Database - Rock Paper Scissor Online Game
- ~ Understanding Firebase Database
- ~ Running game on multiuser
- ~ Setting user registration system to database UI
- ~ Setting user registration system to database - code
- ~ Getting complex user data from database
- ~ Firebase Image Uploader Part 1
- ~ Firebase Image Uploader Part 2

=> Machine learning - Face and Smile detection app :

- ~ Machine Learning KIT in Firebase
- ~ Connecting with MLKIT online
- ~ Custom assets and gradle
- ~ Firebase app initializer
- ~ Inflating result dialog box
- ~ Open a camera on a REAL device
- ~ Final code for Face and smile detection

=> Machine Learning - Text Detection app :

- ~ Text Recognition app
- ~ How to download exercise files
- ~ Adding Custom Assets
- ~ Firebase initializer
- ~ Result Activity
- ~ Firecamera in our app
- ~ Text Recognition and Debug

=> How to publish app on store :

- ~ How to publish app on store

# Tibco Business Works

---

Topic Name : DATA ANALYTICS

Sub-topic Name : DASHBOARDING

Course link : <https://ineuron.ai/course/Tibco-Business-Works>

## Course Description :-

The principles and overview of TIBCO Active Matrix Business Works are covered in depth in this TIBCO Business Works course.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction to TIBCO AMX BusinessWorks
- => Features of TIBCO AMX BusinessWorks
- => TIBCO AMX BusinessWorks Installation
- => What is TIBCO EMS and JMS
- => Messaging Concepts
- => Destinations (Queue/Topic)
- => Destination Properties
- => Introduction to TIBCO AMX BusinessWorks Debugger
- => Debug Job Data
- => Using Breakpoints
- => Configuring Debugging Environment

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to EAI :
  - ~ Introduction to EAI
- => Introduction to TIBCO BusinessWorks :
  - ~ Introduction to TIBCO AMX BusinessWorks
  - ~ Features of TIBCO AMX BusinessWorks
  - ~ TIBCO AMX BusinessWorks Installation
  - ~ Introduction to TIBCO Business Studio
  - ~ Using Workspace
- => TIBCO AMX BusinessWorks Design and Development :
  - ~ Introduction to Application, Module, Processes and Packages
  - ~ Introduction to Activities/Palette/Transitions/Shared Resources
  - ~ Mapping and Transformations
  - ~ Building Web-services(SOAP HTTP/JMS)
- => Working With Properties/Variables :
  - ~ Introduction to Process Properties
  - ~ Introduction to Module Properties
  - ~ Introduction to Application Properties
  - ~ Scope of Properties
  - ~ Introduction to Application Configuration Profiles
  - ~ Working With Properties
- => Working with Groups :
  - ~ Introduction to Groups
  - ~ Understanding Group Options
  - ~ Implementing Loops Using Groups
  - ~ Exception Handling
  - ~ Transaction Handling
  - ~ Development Of TIBCO AMX BW Application Using Groups For A Real-time Scenario

=> Working With JDBC Activities/Drivers :

- ~ Introduction to JDBC Activities
- ~ Configuring JDBC Drivers At Design-time and Run-time
- ~ Working With JDBC Activities
- ~ Development of TIBCO AMX BW Application With JDBC Activities
- ~ For A Real-time Scenario

=> Working with SOAP Activities :

- ~ Introduction to Soap
- ~ Creation of WSDL(Abtract and concrete WSDLs)
- ~ Creation of webservices
- ~ Development of Soap based webservice with a Realtime Scenario and test it with
- ~ external clients like Soap UI

=> Working with REST Activities :

- ~ Introduction to REST/JSON Palette
- ~ Invoking REST API
- ~ JSON to XML and XML to JSON Conversion
- ~ Development Of REST Client For A Real-time Scenario

=> Tibco EMS Enterprise messaging service :

- ~ What is TIBCO EMS and JMS
- ~ Messaging Concepts
- ~ Destinations (Queue/Topic)
- ~ Destination Properties
- ~ Delivery Modes
- ~ Types of Acknowledgement
- ~ Bridge/Route
- ~ EMS Administration Commands
- ~ Configuring GEMS Tool For Monitoring/Management

=> Debugging Application :

- ~ Introduction to TIBCO AMX BusinessWorks Debugger
- ~ Debug Job Data
- ~ Using Breakpoints
- ~ Configuring Debugging Environment
- ~ Testing TIBCO AMX BusinessWorks Application
- ~ Endpoint details

=> Application Packaging and Deployment :

- ~ Validating Project and Building EAR File
- ~ Introduction to TIBCO AMX BW Runtime Components
- ~ TIBCO AMX BW Runtime Architecture
- ~ Domain/Appnode/Appspace/Application
- ~ Introduction to various Deployment Modes
- ~ Introduction to TIBCO Enterprise Administrator (TEA)/Administrator GUI

# Statistics Live Class

---

Topic Name : DATA SCIENCE

Sub-topic Name : STATS

Course link : <https://ineuron.ai/course/Statistics-Live-Class>

## Course Description :-

Statistics is the discipline that concerns the collection, organization, analysis, interpretation and presentation of data. Statistics is the foundation behind all the work you want to do regarding Data Science. So, you must know all the statistical concepts to learn data science well. In this course, you will learn all the statistical concepts in detail that will be highly beneficial for various fields of Data Science.

## Course Features :-

- => Online classes
- => Doubt Clearing
- => Live-Class Recording
- => Real-time Project
- => Assignment in all modules
- => Quiz in every module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Understand what a Normal Distribution is.
- => Explain the difference between continuous and discrete variables
- => Understand the Central Limit Theorem
- => Use the Z-Score and Z-Tables
- => Understand the difference between normal distribution and t-distribution
- => Create confidence intervals
- => Understand standard deviations
- => Understand what a sampling distribution is
- => Apply Hypothesis Testing for Proportions
- => Use the t-Score and t-Tables

## Requirements :-

- => Basic understanding of Maths
- => A system with internet connection
- => Your dedication

## Instructors :-

=> Bharath J P V :

~ Enthusiast Data Scientist with a strong background in Mathematics and Statistics. Completed My Master in Statistics. Have experience teaching Mathematics and Statistics for more than a year. I thought for more than 1000 students and helped them make their careers in their respective fields. I believe in "we rise by lifting others". Following this principle, I hope to make your life easier.

## Curriculum details :-

- => Course introduction :
  - ~ Introduction
- => Stats Fundamental :
  - ~ Statistics
  - ~ Inferential Statistics
  - ~ Descriptive Statistics
  - ~ Mean, Median and Mode
  - ~ Population vs Sample
  - ~ Guassian or Normal Distribution
  - ~ Log Normal Distribution
  - ~ Covariance
  - ~ Central Limit Theorem
  - ~ Chebyshev's inequality
  - ~ Pearson Correlation Coefficient
  - ~ Spearman's Rank Correlation Coefficient
  - ~ Standardization vs Normalization
- => Python :



~ *Use of Python in Statistics*

=> Representation and interaction with Data :

~ *Data as a table*

~ *Pandas DataFrame*

=> Hypothesis testing: Comparing two groups :

~ *Student's T-test*

~ *Paired test*

=> Linear models, multiple factors, and analysis of variance :

~ *Python formulas for specifying statistical models*

~ *Multiple Regression*

~ *Analysis of variance(ANOVA)*

=> Visualization: Statistical exploration using Seaborn :

~ *Pairplot: scatter matrices*

~ *Implot: plotting a univariate regression*

# Full Stack JavaScript Bootcamp 2.0

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : FULL STACK WEB DEVELOPMENT

Course link : <https://ineuron.ai/course/Full-Stack-JavaScript-Bootcamp-2.0>

## Course Description :-

Learn everything about JavaScript with React, Nodejs, MongoDB, React Native, Tailwind, etc. Build industry-ready projects for web and mobile.

## Course Features :-

- => Full stack Web Developer certification
- => Job guarantee Program
- => One year of internship
- => Revision Classes
- => Online Instructor-led learning: Live teaching by instructors
- => 50+ hands-on industry real-time projects.
- => 100+ hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing through mail and discussion forum
- => Assignment in all the module
- => Quiz in every module
- => A live project with real-time implementation
- => Resume building
- => Career guidance
- => Interview Preparation
- => Regular assessment
- => Job Fair and Internal Hiring
- => Mock Interview
- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => HTML and DOM
- => Starting with CSS
- => Working on coming soon template
- => Mobile responsive webpages
- => Register a new account
- => CSS animation and libraries
- => Flexbox in CSS
- => Projects- HTML & CSS
- => JavaScript Refresher
- => Project - JavaScript
- => JavaScript interview questions
- => Create a react template
- => Getting friendly with states
- => Building a Tic Tac Toe

=> Learn React Context API with projects

=> Making Project with Nodejs

=> Getting Started with Reactive Native

=> Making Project in React Native

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

=> Anurag Tiwari :

*~ Hey, I am Anurag Tiwari, a developer at learncodeonline. We have built a scalable system handled by 300K users on a daily basis. I'm a software developer who constantly seeks innovative solutions to everyday problems. I have been teaching students for the last 24 months.*

## Curriculum details :-

=> Getting started :

- ~ Introduction to web dev
- ~ Get the development tools
- ~ your first hello world
- ~ Goal of this course and instructions
- ~ Tools for backend developer
- ~ Mongodb MAC install
- ~ Mongodb WIN install
- ~ MongoDB in cloud - Atlas
- ~ Mongo GUI - compass

=> We do not write without emmet :

- ~ Getting started with emmet
- ~ Emmet and speedy html
- ~ Parent child and grouping
- ~ emmet in css

=> HTML and DOM :

- ~ An old style blog
- ~ Why DOM is important
- ~ Inline vs Block and bring in images
- ~ Lists and interlinking pages
- ~ Getting a video on service page
- ~ 3 Plans in a table
- ~ GET and POST forms
- ~ Types of input forms

=> Starting with CSS :

- ~ Secret to learn CSS
- ~ Explore and bring in fonts
- ~ Bring in colors and styles
- ~ Transition and box shadow DOCS
- ~ Margin and padding
- ~ Button gets all and assignment

=> Working on coming soon template :

- ~ Introduction to CSS variables and new project
- ~ Browser defaults and variables
- ~ Getting more control over elements

=> Mobile responsive webpages :

- ~ What are media queries
- ~ Media query in action
- ~ App landing page - setup
- ~ Navigation bar for website
- ~ Bring content in columns
- ~ Cover image in css
- ~ Start with flexbox
- ~ Buttons and columns in flexbox
- ~ Absolute position in CSS
- ~ Media query for 2 screens

=> Register a new account :

- ~ Handle conflict in CSS
- ~ Strategy and placing html
- ~ Class and ID for testers
- ~ Where to use z index
- ~ Bootstrap style of CSS
- ~ Fixing CSS on form
- ~ Doing your assignment

=> CSS animation and libraries :

- ~ Animation and keyframes
- ~ Third party animation library
- ~ Razorpay style clipping

~ Not a payment gateway integration

=> Flexbox in CSS :

- ~ What is flexbox
- ~ Get to know the power of flexbox
- ~ Flexbox series-Axis and Flex direction
- ~ Flexbox series -justify content
- ~ Align items in flexbox
- ~ Flexbox series ordering the elements
- ~ Flex grow in flexbox

=> Projects- HTML & CSS :

- ~ Mock up with design files to code files -1
- ~ Mock up with design files to code files -2
- ~ Mock up with design files to code files -3
- ~ Mock up with design files to code files -4
- ~ Mock up with design files to code files -5

=> JavaScript Refresher :

- ~ Welcome to JavaScript Course
- ~ What are JavaScript engines
- ~ What ES version of JavaScript is good for us
- ~ Variable and datatypes in JavaScript
- ~ Our first User Signup
- ~ Operators in JavaScript Calculate discount
- ~ Type and Operator precedence in JavaScript
- ~ What are conditionals in JavaScript
- ~ Logical conditional Login in JavaScript
- ~ Ternary operator in JavaScript
- ~ Switch for role-based access in JavaScript
- ~ Coercion and falsy values in JavaScript
- ~ Basics of functions in JavaScript
- ~ Functions in variable User Role in JavaScript
- ~ Understand the context in JavaScript
- ~ Code hoisting in JavaScript
- ~ Scope chaining in JavaScript
- ~ Light intro to THIS in JavaScript
- ~ Introduction to Array in JavaScript
- ~ Callback and arrow function introduction in array
- ~ Fill and Filter in Array in JavaScript
- ~ Slice and Splice in JavaScript
- ~ Objects in JavaScript
- ~ Methods and objects in JavaScript
- ~ For loop basics in JavaScript
- ~ While and do while loops in JavaScript
- ~ For Each loop in JavaScript
- ~ For in and for of loop in JavaScript
- ~ Confusing part of THIS in JavaScript
- ~ What is DOM
- ~ How to grab web elements in JavaScript
- ~ A counter project in JavaScript
- ~ Get Computed properties in JavaScript
- ~ Event listener in JavaScript
- ~ New keyword in JavaScript
- ~ What is proto in JavaScript
- ~ Better code with object chain in JavaScript
- ~ Objects from MDN docs
- ~ Self-Executing Anonymous Function and functional programming
- ~ Lexical scoping in JavaScript
- ~ Closure in JavaScript
- ~ Borrow a method using bind
- ~ Get to know node Elements in JavaScript
- ~ Generating elements and Text node in DOM
- ~ Solution of Scope problem in JavaScript
- ~ Template literals in JavaScript
- ~ Maps in JavaScript
- ~ Destructure the data in JavaScript
- ~ Spread and REST operators in JavaScript
- ~ Classes and module exports in JavaScript
- ~ Private props getters and setters in JavaScript
- ~ Inheritance in JavaScript
- ~ Event loop Will JavaScript wait
- ~ Promise async and await in JavaScript
- ~ How to Handle API in JavaScript
- ~ Get to know game files
- ~ Logic of game JavaScript
- ~ Fixing the bug in game JavaScript
- ~ What is new in JavaScript 2021
- ~ Why iiife appears in JavaScript interviews
- ~ Quirky Behavior of JavaScript

=> Project - Javascript :

- ~ Project overview
- ~ Project business case
- ~ Project Mockup
- ~ Project setup
- ~ Project code
- ~ Project Deployment
- ~ Project hyper care

=> JavaScript interview questions :

~ *JavaScript industry ready interview questions - 30*

=> Introduction :

~ *Into to REACTJS course*

=> Getting started with ReactJS :

~ *How to use exercise files*

~ *What is react and myths*

~ *Tools that we need*

=> Going All classic :

~ *Section 2 introduction*

~ *Finishing the hello world task*

~ *Delete and recreate everything*

~ *Adding CSS to our Hello World*

~ *Everything in its own file*

~ *Reusable components*

=> Create a react template :

~ *Section 3 introduction*

~ *Understand the existing template*

~ *Move navbar and understand the errors*

~ *Convert the HTML template into React App*

~ *Reusable Card and Assignment*

=> Getting friendly with states :

~ *Section 4 introduction*

~ *What are props and states*

~ *Preparing the state based applications*

~ *Complete counter application*

~ *Assignment for counter app*

=> Building a Tic Tac Toe :

~ *section 5 Introduction*

~ *Your need to study first*

~ *Preparing the Tic Tac Toe*

~ *Sending icons from components*

~ *Setup layout for tictactoe*

~ *Game is almost working*

~ *Finishing tictactoe and assignment*

=> Learn React Context API with projects :

~ *Section 6 Introduction*

~ *The problem that contextAPI solves*

~ *Detail on Context and Provider*

~ *Detail on Consumer in contextAPI*

~ *Understand the working of dark and light mode*

~ *Creating a theme Toggler with Context API*

~ *Finishing the theme switcher app*

=> App with Context API with reducers and actions :

~ *Section 7 introduction*

~ *What are we building here*

~ *Create brain of the application*

~ *useReducer for our app*

~ *Add an input form*

~ *Sending a dispatch*

~ *Display the context data and dispatch*

=> Local storage and useEffect hooks :

~ *Section 8 introduction*

~ *Introducing the Effect hook*

~ *A form to submit the data*

~ *Looping through all the values*

~ *Hooks and local storage in action*

=> Learn to handle API :

~ *Section 9 introduction*

~ *Learn to read docs for API*

~ *lets read Axios docs*

~ *Drill down the API*

~ *Extracting information from API*

=> Designing a shopping cart API :

~ *Section 10 introduction*

~ *A walk through Pexels and JSON*

~ *Add item to the cart*

~ *Buy item and remove item*

~ *Fetching photos from API*

~ *Store everything is state*

~ *Card for every product*

~ *Create cart section*

~ *Bring the shop together*

~ *Removing the duplicate*

=> Firebase with Github App :

~ *Section 11 introduction*

~ *What we are about to build*

~ *React Router crash course*

~ *Your tour to configure firebase*

~ *Read firebase docs with me*

- ~ Creating components for firebase app
- ~ Bring in the react router
- ~ Headers and Footers
- ~ Conditional rendering in Navbar
- ~ Adding firebase configuration
- ~ User Signup in firebase
- ~ Logout and signin user
- ~ User card component
- ~ Repo component
- ~ Home page and finish the app

=> Firebase real time database :

- ~ Section 12 introduction
- ~ A challenge application
- ~ Firebase real time database
- ~ Setting context and actions
- ~ Creating reducers for contact
- ~ Header and Footer tasks
- ~ How to upload image in firebase storage
- ~ Add and update contact in firebase
- ~ Add or update finder
- ~ Update star and delete contact
- ~ Use dispatch and FIXME
- ~ Get all data from firebase
- ~ Loop through firebase object
- ~ Firebase finale and assignment

=> Bonus-Redux App :

- ~ 3 Principles of redux
- ~ Bring in the central state
- ~ Actions make redux simpler
- ~ Reducer - brain part of app
- ~ Component dispatching the info
- ~ 2 most important method for Redux
- ~ Provider to give access of store
- ~ Finally creating that store

=> React project :

- ~ Chat app with react
- ~ Project overview
- ~ Project business case
- ~ Project Mockup
- ~ Project setup
- ~ Project code
- ~ Project Deployment
- ~ Project hyper care

=> React interview question :

- ~ React industry interview question - 30

=> Resume Discussion :

- ~ Key points for your resume
- ~ Templates for resume
- ~ Project for your resume
- ~ prepare your GIT
- ~ prepare your social media profile
- ~ prepare your demo for your resume
- ~ Detail project report
- ~ resume verification
- ~ place where you can apply for job
- ~ Final touch
- ~ Go get your Dream job
- ~ Key points for your resume
- ~ Templates for resume
- ~ Project for your resume
- ~ prepare your GIT
- ~ prepare your social media profile
- ~ prepare your demo for your resume
- ~ Detail project report
- ~ Resume verification
- ~ Place where you can apply for job
- ~ Final touch
- ~ Go get your Dream job
- ~ Resume Discussion

=> MongoDB :

- ~ What is mongoDB
- ~ How does mongoDB works
- ~ What is mocha and need of mocha in mongodb
- ~ Big umbrella of MongoDB
- ~ How to install mongoDB on MAC
- ~ How to install mongoDB on Windows
- ~ Create and Read operation in MongoDB
- ~ ObjectId and BSON in mongoDB
- ~ Triple A and CRUD operations in mongoDB
- ~ UpdateOne and DeleteOne in #mongoDB
- ~ UpdateMany and deleteMany in mongoDB
- ~ Database issues with Update in mongodb
- ~ Getting more data in #mongodb
- ~ Save bandwidth while querying in mongoDB

- ~ Understanding objects structure in mongoDB
- ~ Understanding Arrays in mongoDB
- ~ What is schema in mongoDB
- ~ Database modeling mongoDB series
- ~ Relation in database mongoDB
- ~ One to one relation in mongo database with id
- ~ One to many relation in mongo database
- ~ Many to Many relation in mongoDB
- ~ Exploring mocha, mongo and mongoose
- ~ Creating file structure and installing dependencies
- ~ Creating our first student schema
- ~ Connecting with mongoddb with mongoose
- ~ BlueBird and Q promises
- ~ Hooks in mocha
- ~ Describe and it blocks for mocha test
- ~ Your first create test using mocha
- ~ Using beforeEach in mocha tests
- ~ A read test in mongoDB
- ~ A delete test for mongoDB
- ~ An update test in mongoDB

=> Project :- Take it up to Heroku - Production :

- ~ Things you need to deploy on Heroku
- ~ Plan your application
- ~ Types of web request
- ~ Framework - Express, Koa, Hapi
- ~ Starting with package JSON file
- ~ Your first express app
- ~ Request Response and Status code
- ~ All social routes
- ~ Handle the date situation
- ~ Parameters and bugs in route
- ~ Pushing app to HEROKU
- ~ Debug social app in production

=> Project:-Swagger Docs :

- ~ What is swagger and api docs
- ~ Nodemon ext and YAML docs
- ~ Authentication token for swagger docs
- ~ Docs for HTTP methods swagger
- ~ A new documentation centric project
- ~ Setup information - swagger
- ~ Authentication and Authorization - swagger
- ~ String based GET request - swagger
- ~ handling objects - swagger
- ~ handling array in Swagger docs
- ~ Sending data in URL - swagger
- ~ managing request body in swagger
- ~ handle url query in swagger
- ~ handling images in swagger
- ~ handling header tokens in swagger

=> Project:-Authentication :

- ~ What we have done till section 3 - backend
- ~ Hiding secrets in backend
- ~ Picking up a database for backend
- ~ Why we need mongoose - ODM
- ~ Pro db modeling tools
- ~ Creating model for auth system
- ~ Creating basic structure for auth system
- ~ Creating user schema and dotenv
- ~ Registering a user in auth system
- ~ Database connection in auth system
- ~ What is a middleware
- ~ Handling password situation
- ~ What is JWT and creating token
- ~ Register route in auth app
- ~ Login flow for auth app
- ~ Web vs Mobile
- ~ Writing custom middleware
- ~ Setting up secure cookies

=> Project :-File, image and form handling :

- ~ Why people face issue in image upload
- ~ Cloudinary and EJS
- ~ How GET works and postman issues
- ~ Using template engines
- ~ Biggest confusion in front end forms
- ~ Handling images in forms
- ~ Handling images in forms part 2
- ~ upload image to cloudinary or other providers
- ~ Handling multiple files and uploading them

=> Project:- Theory and Razorpay :

- ~ File structure for production app
- ~ Getting a logger - MORGAN
- ~ Error handler and Promises
- ~ Sending emails using nodemailer
- ~ Why mongoose docs are important
- ~ Razorpay project

~ Razorpay front end integration

=> Project:- Big Ecommerce app starts :

~ Project requirement

~ User modeling and file structure

~ Product model discussion

~ Order Model discussion

~ How forgot password feature work

~ Functions in user model and hooks

=> Project:- Basic Config and imports :

~ Getting files and folders ready

~ Preparing basic express app

~ Routes and controllers in dummy

~ Injecting docs and middleware

~ Custom error handlers

~ The big Promise

=> Project:- User model and signup :

~ Creating a user model and validator

~ password encryption and mongoose prototypes

~ Validating the password

~ creating JWT tokens

~ forgot password and crypto hashing

~ User routes and postman

~ Signup a user and cookies

~ Database connection

~ Testing the user signup with postman

~ Handling image upload

~ Testing photo upload and user signup

~ yes, we know about postman files

=> Project:- User controllers and routes :

~ Login route and controller

~ logout controller and route

~ Send email from node

~ Forgot password controller

~ Reset password controller and routes

~ Middleware - injecting information

~ User dashboard controller and routes

~ Update the password for a user

~ Updating the user profile

~ User, admin, manager and more roles

~ Manager only routes

~ Admin get a single user

~ Admin can update any user

~ Admin can delete a user now

=> Project:- Working on Product Model :

~ Product middleware setup for routes

~ Product Model and refs

~ A long talk on URL replace and mongo operators

~ Creating a product

~ Where clause in search

~ Where clause Pager

~ Aggregation filter in Where Clause

~ Get all products with WHERE and pager

~ Debugging and testing of product add and get

=> Project:-More routes in Products :

~ Single product route

~ Update the product with photos

~ Delete a product and minor bug

~ Testing and debugging

~ Add a review

~ Delete a review and requested routes

~ Configure routes for reviews

=> Project:- Razorpay and Stripe :

~ Stripe Docs

~ Stripe controllers

~ Razorpay payments and order

~ Setup payment routes

=> Project:- Processing Orders :

~ Order model in action

~ Creating a order and BSON

~ Testing create order and routes

~ Populate fields in order

~ Order of routes is important

~ Updating the stock

~ Delete order and push to git

~ Pushing code to production server

=> Project :- Auth and Social Logins :

~ Social login foundation and demo app

~ Consent screen and API keys

~ Why passport js

~ Package installation

~ Home routes and EJS

~ Preparing routes for login



- ~ Showing consent screen of google
- ~ Getting information and email from google
- ~ Moving google data to database
- ~ Serialize and deserialize user
- ~ Protect the Home

=> Web Development interview question :

- ~ industry ready interview question - 100

=> Resume Discussions & Preparation :

- ~ Key points for your resume
- ~ Templates for resume
- ~ Project for your resume
- ~ prepare your GIT
- ~ prepare your social media profile
- ~ prepare your demo for your resume
- ~ Detail project report
- ~ resume verification
- ~ place where you can apply for job
- ~ Final touch
- ~ go get your Dream job

=> Getting started with React Native :

- ~ What is React native
- ~ Tools we will need
- ~ Installation of tools in MAC
- ~ First Hello World in Android emulator
- ~ How to install chocolatey in Windows
- ~ Android installation and setup on WINDOWS
- ~ Running your first react native app on Windows and AVD
- ~ Run app on real Android device

=> Getting started with First app :

- ~ Strategy for this app
- ~ What is metro
- ~ File structure of react native app
- ~ Get it done from scratch
- ~ 3 ways to style elements in react native
- ~ More styling of text
- ~ State in react Native
- ~ Touchable and props in react native

=> App to handel Images :

- ~ Strategy for dice roller app
- ~ Basic structure of dice roller
- ~ Writing CSS for dice Roller
- ~ Dice roller complete and 2 assignments

=> App to handle sound :

- ~ Strategy to handle Spanish number app
- ~ Adding third party code in react native
- ~ ScrollViews and sound import in react native
- ~ Unique keys for iterable in React Native
- ~ More on React native flexbox
- ~ Playing sound in react native app

=> App to handle user Input :

- ~ Getting started with currency App
- ~ Basic currency values
- ~ Formating text input and result container
- ~ Buttons and keyboard dismiss
- ~ Handling error in input

=> Camera App - Profile Picture :

- ~ Permissions in Camera App
- ~ Having image in state
- ~ Click a photo
- ~ Load image from state

=> Handel third party lib with Game app :

- ~ Getting started with native base
- ~ Installation and configuration of native base
- ~ Your first custom component
- ~ Having a plan of action for ticTacToe
- ~ Win message and reset game
- ~ Finishing game and debugging

=> Reactive native router and local storage :

- ~ Overview of netflix store app
- ~ Going with all installation and screens
- ~ Getting navigation setup ready
- ~ Adding a FAB and moving to add screen
- ~ UI for add season list
- ~ Async Storage in react native
- ~ Home screen setup for seasons
- ~ React native hooks
- ~ Custom hooks in react native
- ~ Checked and delete seasons
- ~ Update and fix the bugs

=> Handling API and web requests :

- ~ Getting started with API

- ~ How to read API
- ~ Fetch a web request
- ~ Pass info to another component
- ~ making card for user

=> Redux in React Native :

- ~ Getting started with Redux
- ~ Redux Three Principles
- ~ action in REDUX
- ~ Reducer in REDUX
- ~ Store in REDUX
- ~ Provider in REDUX
- ~ Dispatch props in REDUX
- ~ state to props in redux

=> Instagram clone - Mega Section :

- ~ Instagram challenges in app
- ~ Having a firebase project
- ~ Connect firebase to RN project
- ~ Auth and Database basics in RN
- ~ Enable MultiDex and action
- ~ Signup a new user
- ~ Register a user in database
- ~ Signin and signout a user
- ~ Dispatch and database events
- ~ Handling all reducers
- ~ Lets create a lot of files
- ~ Store provider and new workflow
- ~ Ask Permission to user
- ~ Redux hooks and firebase subscribers
- ~ screen options to Custom header
- ~ Custom header component
- ~ Midway debugging session
- ~ Prep work for image upload
- ~ Upload image to firebase from mobile
- ~ Testing signup in firebase
- ~ Handle signin screen and testing
- ~ UI and config for addPost
- ~ Adding post to firebase
- ~ sending post to firebase and resolving bugs
- ~ Handling home page
- ~ Iron out bugs and prepare post component
- ~ Like and dislike feature

=> Mega Project :

- ~ Food Delivery Application
- ~ Ecommerce
- ~ Stackoverflow Clone

=> Mock Interviews :

- ~ Mock Interviews

# Cyber Security Foundations

---

Topic Name : CYBER SECURITY

Sub-topic Name : CYBERSECURITY MASTERS

Course link : <https://ineuron.ai/course/Cyber-Security-Foundations>

## Course Description :-

One of the most famous FAQ on google is How to get started with Ethical hacking? The perfect way to find the correct opportunity according to your potential is to take a deep dive into a course that has a great variety with a lot of practical practice.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Maintaining anonymity and exploring darknet
- => How to get started with bug bounty and core fundamentals building
- => Information gathering via OSINT
- => Intercepting web layers for bug hunting

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Saksham Choudhary :

~ Hello folks, I am AWS certified Cloud Architect Engineer. With having 5+ years of Experience in Teaching, I am currently providing cloud solutions for various products via my strong hands on DevOps Skills. I am a tech youtuber with 120k + subscriber and has taught 30,000 + students, Narcotics, Custom duty officers, Police officials and Corporate candidates.

## Curriculum details :-

- => Day1 :
  - ~ Maintaining anonymity and exploring darknet
- => Day2 :
  - ~ How to get started with bug bounty and core fundamentals building
- => Day3 :
  - ~ Information gathering via OSINT
- => Day4 :
  - ~ Intercepting web layers for bug hunting
- => NaN :
  - ~ NaN
  - ~ NaN
  - ~ NaN

# ELK Stack

---

Topic Name : DEVOPS

Sub-topic Name : DEVOPS MASTERS

Course link : <https://ineuron.ai/course/ELK-Stack>

## Course Description :-

This program is specialized in providing theoretical as well as practical understanding of the ELK stack or elastic stack which allows you to learn about Elastic search, Kibana and logstash and how they can work together. Course curriculum includes ELK stack, elastic search, kibana, logstash, architecture, practical implementation and much more!

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => ELK Stack
- => Elastic search
- => Kibana
- => Logstash
- => Installation
- => Architecture
- => Handson

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Course Introduction :

~ Syllabus overview Preview

=> ELK Stack :

~ What is ELK Stack Preview

~ Use cases of ELK Stack

=> Elastic Search :

~ What is Elastic search

~ Why Elastic search

=> Kibana :

~ What is Kibana

~ Roles of kibana in ELK Stack

=> Logstash :

~ What is Logstash?

~ Why use logstash?

~ How elastic search, kibana, logstash work together

=> Architecture of Elastic search :

~ Architecture of Elastic search

~ Sharding

~ Replica sharding

=> Installation :

~ Installation kibana

~ Elasticsearch & Logstash installation

=> Practical implementation :

~ Hands-on kibana

~ Elastic search cloud service

~ CRUD Hands-on

- ~ *Fine tuning concept*
- ~ *Hands-on fine tuning*

# Full Stack Data Analytics 2.0

---

Topic Name : DATA ANALYTICS

Sub-topic Name : BUSINESS ANALYTICS MASTERS

Course link : <https://ineuron.ai/course/Full-Stack-Data-Analytics-2.0>

## Course Description :-

The full stack data analytics course is meant to assist you in becoming a skilled data analyst. Learn how to deal with SQL databases, develop data visualizations, and apply predictive analytics and statistics in a corporate environment using the best analytics tools and methodologies.

## Course Features :-

- => Full stack Data Analytics certification
- => Internship Anytime
- => Online Instructor-led learning: Live teaching by instructors
- => 15+ hands-on industry real-time projects.
- => 100 hours live interactive classes.
- => Every week doubt clearing session after the live classes.
- => Lifetime Dashboard access
- => Doubt clearing one to one
- => Doubt clearing through mail & support team
- => Live project with real-time implementation
- => Resume building Anytime
- => Career guidance Anytime
- => Interview Preparation
- => Regular assessment
- => Mock Interview

## What you will learn :-

- => MySQL
- => Basic Charts in Power BI
- => Working with Maps
- => Slicers in Power BI
- => Cards and Filters
- => Power Query
- => M Language
- => Tableau
- => SQL
- => Python
- => Statistics
- => Excel
- => Informatica Cloud (IICS)

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Sudhanshu Kumar :

~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

=> Anand Jha :

~ Experienced developer with 6+ years of overall IT experience in Software Development Life Cycle. Vast experience in helping clients in their business

requirements delivery through data acquisition, analysis, and driving insights through machine learning model developments, KPI reporting, and driving business to a higher level. Data Story Teller using MS-Excel, Tableau & Powerbi by analyzing raw data and building KPI Reports. Solid understanding of exploratory data analysis using SQL, R and PYTHON Advanced understanding of statistical, algebraic, and other analytical techniques Good understanding of Cloud Technologies : SNOWFLAKE , AWS , AZURE & GOOGLE CLOUD Experience working in an Agile Environment. Strong Communication, Presentation, and Interpersonal skills with excellent problem-solving capabilities.

## Curriculum details :-

### => MySQL :

- ~ MySQL
- ~ SQL
- ~ SQL Query
- ~ SQL Queries Part 2
- ~ SQL Queries
- ~ SQL function and procedure
- ~ SQL primary and foreign key
- ~ Window Function
- ~ Partition
- ~ Joins ,Union , Indexing , CTE
- ~ Triggers & Case
- ~ NF & Pivote
- ~ Snowflake
- ~ Snowflake Fundamentals
- ~ SQL Project
- ~ Sql Date Functions
- ~ Staging in SNOWFLAKE ( loading data from AWS to snowflake )
- ~ CIA\_Factbook\_Population\_Datasets
- ~ SNOWFALKE CONTINUOUS DATA LOAD USING SNOWPIPE
- ~ Snowflake continuous data loading

### => Informatica Cloud (IICS) Introduction :

- ~ What is Cloud Computing
- ~ The Different Types of cloud computing

### => IICS-Informatica Intelligent Cloud Services :

- ~ What is PasS?
- ~ Power Center on premise vs Informatica cloud
- ~ Power Center on cloud vs Informatica cloud
- ~ Advantages of using Informatica cloud
- ~ Informatica Cloud Architecture
- ~ Informatica Cloud Connectivity and Secure Agent
- ~ Informatica Documentation

### => Getting Started with IICS :

- ~ How to Create a informatica cloud account
- ~ What are the different services available in informatica cloud
- ~ Quick Overview of Data Integration Service
- ~ Quick Overview of Administration Service
- ~ Quick Overview of Monitor Service
- ~ Steps to download and install secure agent

### => Data Integration Service :

- ~ Folder Creation under Explore Option
- ~ Mapping Designer-Layout
- ~ How to Configure a reusable Flat File Connection
- ~ How to Configure a reusable Relational connection for an on-premise database
- ~ Steps to Create the mapping and execute it
- ~ Copy to and move to options
- ~ How are the environments setup work in the cloud
- ~ What are sub organization and how to link sub organizations
- ~ Sub organization option

### => Connections :

- ~ Setup salesforce connection
- ~ Setup big query connection
- ~ Setup snowflake connection
- ~ Setup Amazon S3 connection

### => Transformations :

- ~ Creating Customer Table in Oracle Database
- ~ Set up Oracle connection
- ~ Create Mapping
- ~ Connect with Target Table
- ~ Expression Transformation
- ~ Overview in Expression Transformation
- ~ Router Transformation
- ~ Filter Transformation
- ~ Aggregator Transformation

### => Tasks :

- ~ Mapping Task Properties in IICS
- ~ Mapping Task Demo
- ~ Synchronization Task in IICS
- ~ Replication Task in IICS
- ~ Data Transfer Task in IICS

### => Taskflows :

- ~ Create a Taskflow in IICS
- ~ Tasks inside a Taskflow in IICS
- ~ Schedule a Taskflow in IICS
- ~ Unschedule a Taskflow in IICS

- ~ File Listener in IICS
- ~ Start Taskflow from File Listener in IICS
- ~ Start Taskflow from Faulted step in IICS
- ~ Blackout Period in IICS
- ~ Copy,Rename,move,Delete Assets in IICS
- ~ Deploy the assets from one environment to another environment

#### => Informatica Projects :

- ~ Build an ETL Pipeline for Bank Transactions
- ~ Develop Informatica Mappings, Sessions, Worklets, Workflows for Sales Dataset
- ~ Create data warehouse for multiple subject

#### => PowerBI Live Classes :

- ~ Power Bi
- ~ Snowflake Integration with Power Bi + Donut Charts
- ~ Line Chart (All types) + Slicers + Managing Relationship

#### => Basic Charts in Power BI :

- ~ 2.0 Basic Charts in Power BI Desktop
- ~ 2.1 Column Chart in Power BI
- ~ 2.2 Stacked Column Chart in Power BI
- ~ 2.3 Pie Chart in Power BI
- ~ 2.4 Donut Chart in Power BI
- ~ 2.5 Funnel Chart in Power BI
- ~ 2.6 Ribbon Chart
- ~ 2.7 Include and Exclude
- ~ 2.8 Export data from Visual

#### => Working with Maps :

- ~ 3.1 Creating a Map in Power BI
- ~ 3.2 Filled Map
- ~ 3.3 Map with Pie Chart
- ~ 3.4 Formatting in Map
- ~ 3.5 Change Background in Map
- ~ 3.6 Map of India in Power BI
- ~ 3.7 Map of Australia in Power BI

#### => Tables and Matrix in Power BI :

- ~ 4.0 Table and Matrix in Power BI
- ~ 4.1 Creating a Table in Power BI
- ~ 4.2 Formatting a Table
- ~ 4.3 Conditional Formatting in Table
- ~ 4.4 Aggregation in Table
- ~ 4.5 Matrix in Power BI
- ~ 4.6 Conditional Formatting in Matrix
- ~ 4.7 Hierarchy in Matrix
- ~ 4.8 Sub-Total and Total in Matrix
- ~ 4.9 Number Formatting in Table

#### => Other Charts in Power BI :

- ~ 5.0 Other Charts in Power BI
- ~ 5.1 Line Chart in Power BI
- ~ 5.2 Drill Down in Line Chart
- ~ 5.3 Area Chart in Power BI
- ~ 5.4 Line vs Column Chart in Power BI
- ~ 5.5 Scatter Plot in Power BI
- ~ 5.6 Waterfall Chart in Power BI
- ~ 6.7 TreeMap in Power BI
- ~ 5.8 Gauge Chart in Power BI

#### => Cards and Filters :

- ~ 6.0 Cards and Filters in Power BI
- ~ 6.1 Number Card
- ~ 6.2 Text Card
- ~ 6.2.1 Formatting of Text Card
- ~ 6.3 Date Card
- ~ 6.3.1 Date Card (Relative Filtering)
- ~ 6.4 Multi-Row Card
- ~ 6.5 Filter on Visual
- ~ 6.6 Filter on This Page
- ~ 6.7 Filter on All Pages
- ~ 6.8 Drillthrough in Power BI

#### => Slicers in Power BI :

- ~ 7.0 Slicers in Power BI
- ~ 7.1 Text Slicers in Power BI
- ~ 7.2 Formatting a Text Slicer
- ~ 7.3 Date Slicers in Power BI
- ~ 7.4 Formatting a Date Slicer
- ~ 7.5 Number Slicers in Power BI

#### => Advanced Charts in Power BI :

- ~ 8.0 Advanced Charts in Power BI
- ~ 8.1 Animated Bar Chart Race
- ~ 8.2 Drill down donut Chart
- ~ 8.3 Drill down Column chart
- ~ 8.4 Word Cloud in Power BI
- ~ 8.5 Sankey Chart in Power BI
- ~ 8.6 Infographic in Power BI
- ~ 8.7 Play Axis in Power BI
- ~ 8.8 Scroller in Power BI



- ~ 8.9 Sunburst Chart in Power BI
- ~ 8.10 Histogram in Power BI

#### => Objects in Power BI :

- ~ 9.1 Insert Image in Power BI
- ~ 9.2 Insert Text in Power BI
- ~ 9.3 Insert Shapes in Power BI
- ~ 9.4 Insert Buttons in Power BI
- ~ 9.5 Web URL Action in Power BI
- ~ 9.6 Page Navigation Action in Power BI
- ~ 9.7 Bookmark Action in Power BI
- ~ 9.8 Drillthrough Action in Power BI

#### => Power BI Service Introduction :

- ~ 10.1 Create a Superstore Report in Power BI
- ~ 10.2 Create an Account on Power BI Service
- ~ 10.3 Publish Report to Power BI Service Account
- ~ 10.4 Export Power BI Report to PPT, PDF or PBIX
- ~ 10.5 Comment, Share and Subscribe to Power BI Report
- ~ 10.6 Create a Dashboard in Power BI Service
- ~ 10.7 Problem in Power BI Dashboard and its solution
- ~ 10.8 Automatic Refresh in Power BI using Gateway

#### => Power Query - Text Functions :

- ~ 11.0 Text Functions in Power Query (Power BI)
- ~ 11.1 Merge Columns in Power Query (Power BI)
- ~ 11.2 Split and Trim in Power Query (Power BI)
- ~ 11.3 Upper, Lower and ProperCase in Power Query (Power BI)
- ~ 11.4 Prefix and Suffix in Power Query (Power BI)
- ~ 11.5 Left, Right and Mid Functions in Power Query (Power BI)
- ~ 11.6 Extract Text with Delimiters

#### => Power Query - Date Functions :

- ~ 12.0 Date Functions in Power Query (Power BI)
- ~ 12.1 Year, Quarter, Month and Day Functions in Power Query (Power BI)
- ~ 12.2 Find Difference between Dates in Power Query (Power BI)
- ~ 12.3 Month and Day Name in Power Query (Power BI)
- ~ 12.4 Day, Week of Month, Year in Power Query (Power BI)
- ~ 12.5 Extract Date, Time in Power Query (Power BI)
- ~ 12.6 Calculate Age in Power Query (Power BI)
- ~ 12.7 Day of Year, Quarter, Month in Power Query (Power BI)

#### => Power Query - Number Functions :

- ~ 13.0 Number Functions in Power Query (Power BI)
- ~ 13.1 Basic Number Functions in Power Query (Power BI)
- ~ 13.2 Percentage, Percent Of, Module in Power Query (Power BI)
- ~ 13.3 Round Functions in Power Query (Power BI)
- ~ 13.4 IsEven, IsODD, Sign in Power Query (Power BI)

#### => Power Query - Append Files :

- ~ 14.1 Append multiple CSV files in a folder in Power Query (Power BI)
- ~ 14.2 Append multiple excel sheets, Tables in Power Query (Power BI)
- ~ 14.3 Append Excel sheets or Tables with different columns in Power BI
- ~ 14.4 Append multiple Excel files from a folder in Power BI
- ~ 14.5 Append different data sources in Power BI

#### => Power Query - Merge Files :

- ~ 15.0 Merge Files and Tables in Power BI
- ~ 15.1 Merge Sheets or Tables in Power Query (Power BI)
- ~ 15.2 Merge Data from multiple Excel files or Workbooks in Power BI
- ~ 15.3 Merge data from different data sources in Power Query (Power BI)
- ~ 15.4 Merge data having multiple criteria in Power BI

#### => Power Query - Conditional Columns :

- ~ 16.0 Conditional Column and Column from example in Power BI
- ~ 16.1 Column from examples in Power BI - Split Text
- ~ 16.2 Column from examples in Power BI - Merge Columns
- ~ 16.3 Column from Examples in Power BI - Date
- ~ 16.4 Column from Examples in Power BI - Alphanumeric
- ~ 16.5 Conditional Column in Power BI - One Column
- ~ 16.6 Conditional Column in Power BI - two columns
- ~ 16.7 Conditional Column in Power BI - Compare two columns
- ~ 16.8 Conditional Column in Power BI - on Dates

#### => Power Query - Important Topics :

- ~ 17.0 Very Important Topics in Power Query (Power BI)
- ~ 17.1 Fill Down in Power BI
- ~ 17.2 Grouping in Power Query (Power BI)
- ~ 17.3 Transpose in Power Query (Power BI)
- ~ 17.4 Unpivot In Power Query (Power BI)
- ~ 17.5 Data Types in Power Query (Power BI)
- ~ 17.6 Replace Errors and Values in Power Query (Power BI)
- ~ 17.7 Keep and Remove Rows in Power Query (Power BI)
- ~ 17.8 Add, Remove and Goto Columns in Power Query (Power BI)

#### => M Language Introduction :

- ~ 18.0 M Language in Power Query
- ~ 18.1 Introduction to M Language
- ~ 18.2 IsIn Date Functions in M Language - Power BI
- ~ 18.3 Add and Subtract Date M Functions in Power BI
- ~ 18.4 Basic Date M Functions in Power BI
- ~ 18.5 Basic Text M Functions in Power BI

- ~ 18.6 Simple M Code in Power BI
- ~ 18.7 Trick to get all 900+ M Functions in Power BI

=> Introduction to tableau :

- ~ Tableau Introduction
- ~ Download and Install Tableau
- ~ Tableau Vs Excel

=> Charts - 1 :

- ~ Column Chart
- ~ Horizontal Bar Chart
- ~ Stacked Column Chart
- ~ Stacked Bar Chart
- ~ Keep Only, Exclude
- ~ Keep Only, Exclude2\_Normal
- ~ Publish to Tableau Public

=> Charts - 2 :

- ~ Pie Chart
- ~ Multiple Pie Chart
- ~ TreeMap\_Editing
- ~ Packed Bubble Chart
- ~ Word Cloud OR Word Map
- ~ Formatting payal

=> Charts - 3 :

- ~ Data Types in Tableau
- ~ Filled Map
- ~ Symbol Maps
- ~ India Map
- ~ Histogram

=> Charts - 4 :

- ~ Text Table
- ~ Text Table with Multiple Measures
- ~ Measure Names and Measure Values
- ~ Line Chart
- ~ Line Chart with Multiple Measures
- ~ Discrete Vs Continuous Line Chart
- ~ Discrete Vs Continuous

=> Charts - 5 :

- ~ Lollipop Chart
- ~ Line Vs Column Chart
- ~ Dual Axis Chart
- ~ Column vs Shapes
- ~ Bar in Bar Chart

=> Charts - 6 :

- ~ Calculated fields
- ~ Conditional Column Chart
- ~ Column chart with Shapes based on condition
- ~ Conditional Maps

=> Charts - 7 :

- ~ Map with Pie Chart
- ~ Map with WMS

=> Charts - 8 :

- ~ Funnel Chart
- ~ Advanced Funnel Chart
- ~ Calendar
- ~ Dumbell Chart
- ~ Donut Chart
- ~ Multiple Donut Chart

=> Charts - 9 :

- ~ Bullet Chart 1
- ~ Bullet Chart 2
- ~ Table Calculations Part 1
- ~ Table Calculations - Compute Using - Part 2
- ~ Table Calculations - Relative - Part 3
- ~ Bump Chart
- ~ Bump Chart with Circle
- ~ 100 Percent Stacked Column Chart

=> Charts - 10 :

- ~ Scatter Plot
- ~ Scatter Plot with Images OR Shapes
- ~ Bubble Chart
- ~ Animation - Column Chart
- ~ Animation - Line Chart
- ~ Animation - Column vs Line Chart

=> Charts - 11 :

- ~ Heat Maps
- ~ Heat Map with Shapes
- ~ Heat Map with Conditional Formatting
- ~ Pareto Chart
- ~ Rounded Bar Chart

=> Introduction and installation of MySQL :

- ~ Introduction to section 1

- ~ MySQL introduction - 5 points to know
- ~ Mysql Installation MAC

#### => Basics of MySQL :

- ~ Introduction to section 2
- ~ Creating and dropping database - Startup
- ~ Resolving the issue for future
- ~ Creating your first table
- ~ Adding values to canon table
- ~ Answering customer question

#### => Playing with data :

- ~ Introduction to section 3
- ~ Primary key, default and NULL
- ~ Table with primary key and default values
- ~ Testing the new table
- ~ Adding new values and answering questions
- ~ Update in customers table
- ~ Delete from the customers table

#### => More on functions :

- ~ Introduction to section 4
- ~ Understand the new lco user DB
- ~ Task for CONCAT
- ~ Task for REPLACE
- ~ task for SUBSTRING
- ~ Task for reverse and CHAR\_LENGTH
- ~ Task for case conversion and DOCS

#### => Answering some DB questions :

- ~ Introduction to section 5
- ~ A task on DISTINCT
- ~ A task for ORDER BY
- ~ A task on LIMIT
- ~ Match the pattern
- ~ A task on COUNT
- ~ SQL MODES and GROUP BY
- ~ MIN MAX and SUBQUERIES
- ~ GROUP BY with MAX and MIN
- ~ SUM and AVERAGE with GROUP BY
- ~ A task on AND OR
- ~ A task in RANGE based selection
- ~ CASE THEN - multiple range selection

#### => A pinch of theory :

- ~ Introduction to section 6
- ~ Data type for INTEGER and STRING
- ~ Data type for DATE, DATETIME and JSON
- ~ DATE TIME code Example
- ~ Get the date and time
- ~ Lets join tom and jerry tables
- ~ Types of JOIN

#### => FOREIGN KEY and JOINS :

- ~ Introduction to section 7
- ~ Code talk over FOREIGN keys
- ~ Understand a new database
- ~ A task on INNER join
- ~ ONE to MANY and MANY TO MANY
- ~ Join more 3 or more tables
- ~ A task on LEFT JOIN
- ~ A task on RIGHT JOIN
- ~ FULL OUTER join and UNION tasks

#### => A pinch of more theory :

- ~ Introduction to section 8
- ~ Database engines - INNODB and more
- ~ ACID in database

#### => A 30 Task assignment for movie DB :

- ~ Introduction to section 9
- ~ How practice database works - FILM

#### => Final exam - single attempt :

- ~ MYSQL Outro and some free resources

#### => Python Basics :

- ~ Python Introduction, Installation and Setup
- ~ Python Basics & Conditionals
- ~ Conditionals & Loops
- ~ Working with Loops
- ~ Working with Strings & Lists
- ~ List manipulation
- ~ Tuple, Set & Dictionary
- ~ Working with Functions

#### => Statistics :

- ~ Introduction
- ~ Different types of Statistics
- ~ Population vs Sample
- ~ Mean, Median and Mode
- ~ Variance, Standard Deviation

- ~ Sample Variance why n-1
- ~ Standard Deviation
- ~ Variables
- ~ Random Variables
- ~ Percentiles & quartiles
- ~ 5 number summary
- ~ Histograms
- ~ Gaussian - Normal distribution
- ~ Standard Normal distribution
- ~ Application Of Zscore
- ~ Basics Of Probability
- ~ Addition Rule In Probability
- ~ Multiplication rule in probability
- ~ Permutation
- ~ Combination
- ~ Log Normal Distribution
- ~ Central Limit theorem
- ~ Statistics - Left Skewed And Right Skewed Distribution And Relation With Mean, Median And Mode
- ~ Covariance
- ~ Pearson And Spearman Rank Correlation
- ~ What is P Value
- ~ What is Confidence Intervals
- ~ How To Perform Hypothesis Testing - Confidence IntervalZ Test Statistics Derive Conclusion
- ~ Hypothesis testing part 2
- ~ Hypothesis testing part 3
- ~ Finalizing statistics

#### => Microsoft Excel Fundamentals :

- ~ Launching Excel
- ~ Microsoft Excel Startup Screen
- ~ Introduction to the Excel Interface
- ~ Customizing the Excel Quick Access Toolbar
- ~ More on the Excel Interface
- ~ Understanding the Structure of an Excel Workbook
- ~ Saving an Excel Document
- ~ Opening an Existing Excel Document
- ~ Common Excel Shortcut Keys

#### => Entering and editing text and formulas :

- ~ Entering Text to Create Spreadsheet Titles
- ~ Working with Numeric Data in Excel
- ~ Entering Date Values in Excel
- ~ Working with Cell References
- ~ Creating Basic Formulas in Excel
- ~ Relative Versus Absolute Cell References in Formulas
- ~ Understanding the Order of Operation

#### => Working with basic excel functions :

- ~ The structure of an Excel Function
- ~ Working with the SUM() Function
- ~ Working with the MIN() and MAX() Functions
- ~ Working with the AVERAGE() Function
- ~ Working with the COUNT() Function
- ~ Adjacent Cells Error in Excel Calculations
- ~ Using the AutoSum Command
- ~ Excel's AutoSum Shortcut Key
- ~ Using the AutoFill Command to Copy Formulas

#### => Modifying an excel worksheet :

- ~ Moving and Copying Data in an Excel Worksheet
- ~ Inserting and Deleting Rows and Columns
- ~ Changing the Width and Height of Cells
- ~ Hiding and Unhiding Excel Rows and Columns
- ~ Renaming an Excel Worksheet
- ~ Deleting an Excel Worksheet
- ~ Moving and Copying an Excel Worksheet

#### => Formatting data in an excel worksheet :

- ~ Working with Font Formatting Commands
- ~ Changing the Background Color of a Cell
- ~ Adding Borders to Cells
- ~ Excel Cell Borders Continued
- ~ Formatting Data as Currency Values
- ~ Formatting Percentages
- ~ Using Excel's Format Painter
- ~ Creating Styles to Format Data
- ~ Merging and Centering Cells
- ~ Using Conditional Formatting
- ~ Editing Excel Conditional Formatting

#### => Alteryx Introduction :

- ~ Introduction to Alteryx
- ~ Download and Install Alteryx
- ~ User Interface of Alteryx

#### => IN/Out Tab :

- ~ Get Data from Excel
- ~ Get Data from CSV
- ~ Append All CSV files
- ~ Browse Tool
- ~ Output Tool - Update Existing Data

- ~ *Directory Tool*
- ~ *Directory Tool - Specific Files*
- ~ *Text Input Tool*
- ~ *Date and Time Tool*

=> Preparation Tab :

- ~ *Auto Field Tool*
- ~ *Data Cleansing Tool*
- ~ *Filter Tool (Text Example)*
- ~ *Filter Tool (Number Example)*
- ~ *Filter Tool ( Date Example)*
- ~ *FORMULA TOOL ( Basic Example )*
- ~ *FORMULA TOOL - (Multiple Examples)*
- ~ *GENERATE ROWS TOOL*
- ~ *IMPUTATION TOOL*
- ~ *MULTI-FIELD BINNING TOOL*
- ~ *MULTI-FIELD FORMULA*
- ~ *MULTI ROW FORMULA*
- ~ *RANDOM % SAMPLE TOOL*
- ~ *SAMPLE TOOL*
- ~ *RECORD ID TOOL*
- ~ *SELECT TOOL*
- ~ *SORT*
- ~ *CREATE SAMPLE TOOL*
- ~ *TILE TOOL*
- ~ *UNIQUE TOOL*

=> Join Tab :

- ~ *APPEND FIELDS TOOL*
- ~ *FIND AND REPLACE TOOL*
- ~ *FUZZY MATCH TOOL*
- ~ *JOIN TOOL*
- ~ *JOIN MULTIPLE TOOL*
- ~ *UNION TOOL*
- ~ *REGEX TOOL*
- ~ *Text To Columns*

=> Transform Tab :

- ~ *CROSS TAB Tool*
- ~ *TRANSDPOSE Tool*
- ~ *RUNNING TOTAL Tool*
- ~ *SUMMARIZE TOOL*

=> Reporting Tab :

- ~ *TABLE TOOL*
- ~ *INTERACTIVE CHART Tool*
- ~ *JOIN TABLE AND CHART*
- ~ *ADD ANNOTATION*
- ~ *REPORT TEXT TOOL*
- ~ *REPORT HEADER TOOL*
- ~ *REPORT FOOTER TOOL*
- ~ *REPORT LAYOUT TOOL*

=> Documentation Tab :

- ~ *COMMENT TOOL*
- ~ *EXPLORER TOOL*
- ~ *CONTAINER TOOL*

=> Case Studies :

- ~ *Case Study 1*
- ~ *Case Study 2*
- ~ *Case Study 3*
- ~ *Case Study 4*
- ~ *Case Study 5*

# Stats for Beginners

---

Topic Name : DATA SCIENCE

Sub-topic Name : STATS

Course link : <https://ineuron.ai/course/Stats-for-Beginners>

## Course Description :-

If the goal of your career as a Data Scientist or Business Analyst then brushing up on your statistics skills is something you need to work on. But it's a difficult task to learn/re-learn all the stats seems like a daunting task. That's because we created this course. Here you will quickly get the absolutely essential stats knowledge for a Data Scientist or Analyst.

## Course Features :-

- => Lifetime Dashboard
- => Free Course
- => Certificate
- => Assignment
- => Quiz

## What you will learn :-

- => Understand what a Normal Distribution is
- => Explain the difference between continuous and discrete variables
- => Understand the Central Limit Theorem
- => Use the Z-Score and Z-Tables
- => Understand the difference between a normal distribution and a t-distribution
- => Create confidence intervals
- => Understand standard deviations
- => Understand what a sampling distribution is
- => Apply Hypothesis Testing for Proportions
- => Use the t-Score and t-Tables

## Requirements :-

- => Basics math understanding

## Instructors :-

=> krish naik :

~ Having 10+ years of experience in Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => How to Learn Statistics for Data Science As A Self Starter- Follow My Way :
  - ~ Statistics Introduction Preview
- => Population vs Sample in Statistics
- => Gaussian distribution or Normal Distribution in statistics
- => Log Normal Distribution in Statistics
- => Covariance in Statistics
- => STATISTICS- Mean, Median And Mode Explained Easily
- => STATISTICS- Population VS Sample and its Importance
- => STATISTICS- What are Random Variables and its Types and its Importance?
- => STATISTICS- Gaussian/ Normal Distribution
- => STATISTICS- What is Central Limit Theorem?
- => STATISTICS- Chebyshev's Inequality
- => Statistics- What is Pearson Correlation Coefficient? Difference between Correlation and Covariance
- => Spearman's rank correlation coefficient- Statistics
- => Statistics-Finding Outliers in Dataset using Z- score and IQR
- => Standardization Vs Normalization- Feature Scaling

# Class 10th Biology

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Class-10th-Biology>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. The Biology section focuses on concepts like food, bodily movements, surroundings of living organisms, garbage, etc.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Life processes
- => Control and coordination
- => How do organisms reproduce
- => Heredity and evolution
- => Our environment
- => Sustainable management of natural resources

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Dr Nishtha Jain :

*~ I am a doctor by profession but a teacher by passion. I have been into the teaching profession for the last 3 years. I have been and am still a mentor for various courses which include technical as well as non-technical ones. These include MS-Excel, Tableau, Computer basics, Biology, English, etc. I love to learn, explore and share my knowledge to whatever extent possible. Being an ardent educator, I have always helped all my students and will continue to do the same.*

## Curriculum details :-

=> Life processes :

- ~ Lecture 1 : Introduction and Types of life processes Preview
- ~ Lecture 2 : Nutrition, Modes of nutrition and Dental caries Preview
- ~ Lecture 3 : Respiration Preview
- ~ Lecture 4 : Excretion (in humans and plants), Haemodialysis and Organ donation
- ~ Lecture 5 : Transportation (in humans)
- ~ Lecture 6 : Blood Pressure, Transportation in Plants
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions
- ~ Lecture 5 - NCERT Solutions
- ~ Lecture 6 - NCERT Solutions
- ~ Lecture 7 - NCERT Solutions
- ~ Lecture 8 - NCERT Solutions

=> Control and coordination :

- ~ Lecture 1 : Introduction, Nervous System
- ~ Lecture 2 : Human brain, Spinal cord
- ~ Lecture 3 : Coordination in Plants
- ~ Lecture 4 : Hormones - Plants and Animals
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions
- ~ Lecture 5 - NCERT Solutions
- ~ Lecture 6 - NCERT Solutions
- ~ Lecture 7 - NCERT Solutions

=> How do organisms reproduce :

- ~ Lecture 1 : Reproduction and its types
- ~ Lecture 2 : Types of Asexual Reproduction
- ~ Lecture 3 : Sexual Reproduction in Plants
- ~ Lecture 4 : Sexual Reproduction in Humans
- ~ Lecture 5 : Menstrual Cycle and Contraceptive methods
- ~ Lecture 1 - NCERT Solutions

- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions
- ~ Lecture 5 - NCERT Solutions
- ~ Lecture 6 - NCERT Solutions

=> Heredity and evolution :

- ~ Lecture 1 : Introduction
- ~ Lecture 2 : Mendel's Laws of Inheritance
- ~ Lecture 3 : Sex determination, Evolution
- ~ Lecture 4 : Speciation, Fossils
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions
- ~ Lecture 5 - NCERT Solutions
- ~ Lecture 6 - NCERT Solutions
- ~ Lecture 7 - NCERT Solutions
- ~ Lecture 8 - NCERT Solutions
- ~ Lecture 9 - NCERT Solutions
- ~ Lecture 10 - NCERT Solutions
- ~ Lecture 11 - NCERT Solutions

=> Our environment :

- ~ Lecture 1 : Environment, Ecosystem, Food Chain, Ecosystem components, Trophic levels, Law of energy transfer
- ~ Lecture 2 : Producers, Consumers, Decomposers, Effect of human activities on the environment, Biodegradable and Non-biodegradable substances
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions
- ~ Lecture 3 - NCERT Solutions
- ~ Lecture 4 - NCERT Solutions
- ~ Lecture 5 - NCERT Solutions
- ~ Lecture 6 - NCERT Solutions



# Data Science Project

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING PROJECT

Course link : <https://ineuron.ai/course/Data-Science-Project>

## Course Description :-

Data science projects are a great way to get started in your career. Working on real-world projects provides us with a sense of an approach to real-world problems. You will learn the principles of data science through several projects and use cases in this course.

This hands-on course provides you with a diverse set of open source data science projects to help you practise, improve, and succeed in your data science career.

## Course Features :-

- => Challenges
- => Quizzes
- => Assignments
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => Data preprocessing
- => Database operations
- => Model selection
- => Project deployment
- => End-to-end real-time projects

## Requirements :-

- => Basic knowledge of Machine Learning and Deep Learning
- => A system with stable internet connection
- => Your dedication

## Instructors :-

- => Sudhanshu Kumar :
  - ~ Having 8+ years of experience in Big data, Data Science and Analytics with product architecture design and delivery. Worked in various product and service based Company. Having an experience of 5+ years in educating people and helping them to make a career transition.

## Curriculum details :-

- => Python Project :
  - ~ web crawlers for image data sentiment analysis and product review sentiment analysis Preview
  - ~ Integration with web portal
  - ~ Integration with rest api, web portal and mongo db on Azure
- => Fault detection in waffers based on sensor data :
  - ~ Introduction Preview
  - ~ The problem statement and data description
  - ~ The application flow
  - ~ Ingestion and validation part1
  - ~ Validation part2
  - ~ DB operations
  - ~ Data preprocessing
  - ~ Clustering
  - ~ Model selection and tuning
  - ~ Prediction
  - ~ Deployment
- => Cement strength prediction :
  - ~ Introduction
  - ~ The problem statement and data description
  - ~ The application flow
  - ~ Code intro and logging
  - ~ Validation and transformation
  - ~ DB operations
  - ~ Data preprocessing
  - ~ Clustering
  - ~ Model selection and tuning
  - ~ Prediction
  - ~ Deployment
- => Credit card defaulters :
  - ~ Introduction

- ~ The problem statement and data description
- ~ The application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operations
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Prediction
- ~ Deployment

=> Forest cover :

- ~ Introduction
- ~ The problem statement and data description
- ~ Application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operations
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Prediction
- ~ Deployment

=> Income prediction :

- ~ Introduction
- ~ The problem statement and data description
- ~ The application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operations
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Prediction
- ~ Deployment

=> Insurance fraud detection :

- ~ Introduction
- ~ The problem statement and data description
- ~ The application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operations
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Prediction
- ~ Deployment

=> Mushroom classification :

- ~ Introduction
- ~ The problem statement and data description
- ~ The application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operations
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Predictions
- ~ Deployment

=> Phishing classifier :

- ~ Introduction
- ~ The problem statement and data description
- ~ The application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operations
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Prediction
- ~ Deployment

=> Thyroid detection :

- ~ Introduction
- ~ The problem statement and data description
- ~ The application flow
- ~ Code intro and logging
- ~ Validation and transformation
- ~ DB operation
- ~ Data preprocessing
- ~ Clustering
- ~ Model selection and tuning
- ~ Prediction
- ~ Deployment

=> Visibility climate :

- ~ Introduction

- ~ *The problem statement and data description*
- ~ *The application flow*
- ~ *Code intro and logging*
- ~ *Validation and transformation*
- ~ *DB operations*
- ~ *Data preprocessing*
- ~ *Clustering*
- ~ *Model selection and tuning*
- ~ *Prediction*
- ~ *Deployment*

# PyTorch Basics

---

Topic Name : DATA SCIENCE

Sub-topic Name : DEEP LEARNING

Course link : <https://ineuron.ai/course/PyTorch-Basics>

## Course Description :-

This program is meant to give you the basics till an advanced understanding of one of the most popular deep learning python frameworks - PyTorch. Course curriculum includes concepts about the PyTorch framework, tutorials, and much more!

## Course Features :-

- => Learning of different concepts of PyTorch framework
- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Basic concepts
- => Installation
- => PyTorch usecases
- => Apply PyTorch to implement DL or ML algos from scratch

## Requirements :-

- => Interest to learn
- => Deep learning ANN course
- => Deep learning concepts
- => Decent internet connection

## Instructors :-

- => Sunny Bhavleen Chandra :

~ Sr. Data Scientist and lecturer at iNeuron.ai with working experience in computer vision, natural language processing and embedded systems. Hands-on experience leveraging machine learning, deep learning, transfer learning models to solve challenging business problems. Also, he has a vast interest in Robotics.

## Curriculum details :-

- => Section: 1 PyTorch Introduction :
  - ~ Introduction to PyTorch Preview
  - ~ PyTorch installation and setup
- => Section: 2 PyTorch Tensors and Operations :
  - ~ What is tensor? & Type Conversions Preview
  - ~ Mathematical Operations
  - ~ Indexing, Slicing, Concatenation, Reshaping Ops
- => Section: 3 AutoGrad :
  - ~ Derivatives, Partial derivative, & Successive Differentiation
- => Section: 4 First Neural Network :
  - ~ Simple ANN Implementation

# Pro Aptitude - Python

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-Aptitude---Python>

## Course Description :-

This course is designed mostly for Python Coding test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> Python Aptitude Test

=> Python Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> Python Aptitude Test :

~ *Python Test 1*

~ *Python Test 2*

~ *Python Test 3*

~ *Python Test 4*

# Statistics using Python

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Statistics-using-Python>

## Course Description :-

This course will brush up on your statistics skills which is something everyone needs to work on in school. But it's a difficult task to learn all the statistical concepts which seems like a daunting task. This course is created keeping in mind the important statistical topics in school. Here you will quickly get the absolutely essential statistical knowledge for the journey to becoming a Statistician, Data Scientist, or Analyst.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with tech
- => Real-time project
- => Live class recording
- => Completion certificate

## What you will learn :-

- => Introduction to Course
- => Introduction to Probability and statistics
- => Sets
- => Permutation and Combination
- => Statistics
- => Probability
- => Regression
- => Hypothesis testing

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

=> Introduction to Course :

- ~ Course Introduction
- ~ Course pre-requisites
- ~ Who is this course for?
- ~ What you will get from this course?
- ~ Introduction to probability and statistics
- ~ How to get access to course materials?
- ~ What career path you can follow after completion of this course?

=> Assignment :

- ~ Give a real world example of probability you see everyday.
- ~ Generate a sequence of  $n$  random coin flips and returns total number of tails.
- ~ Write a function for complement of union  $A = \{1, 2, 3\}$   $B = \{3, -6, 2, 0\}$   $U = \{-10, -9, -8, -7, -6, 0, 1, 2, 3, 4\}$  `complement_of_union(A, B, U)`
- ~ Write a function that takes two natural numbers  $k$  and  $n$  as inputs and returns the set of all tuples of size  $k$  that sum to  $n$ .
- ~ Demonstrate how the sample mean approximates the distribution mean
- ~ Suppose 36% of families own a dog, 30% of families own a cat, and 22% of the families that have a dog also have a cat. A family is chosen at random and found to have a cat. What is the probability they also own a dog?
- ~ Explain a situation where PCA can be utilized in detail.
- ~ Give an example of null hypothesis.

=> Introduction to Probability and statistics :

- ~ What is Probability theory?
- ~ What is Statistics?
- ~ History of Probability and Statistics
- ~ Practical: Simulating coin flips  $k$  times using python
- ~ Stats Case Discussion: Election polls
- ~ Long-Term frequency

=> Sets :

- ~ Basics of Sets
- ~ Venn Diagrams
- ~ Relations

- ~ *Operations*
- ~ *Cartesian Products*
- ~ *Russel's Paradox*

=> **Permutation and Combination :**

- ~ *What is Permutation?*
- ~ *What is Combination?*
- ~ *Applications of Binomial Coefficient?*
- ~ *Properties of Binomial Coefficient?*
- ~ *Binomial Theorem*
- ~ *Multinomials*

=> **Statistics :**

- ~ *Statistics Introduction*
- ~ *Mean*
- ~ *Variance*
- ~ *Mean and Variance estimation*
- ~ *Standard deviation*
- ~ *Confidence interval*

=> **Probability :**

- ~ *Probability Introduction*
- ~ *Distribution types*
- ~ *Events*
- ~ *Inequalities*
- ~ *Conditional probability*
- ~ *Sequential probability*
- ~ *Total probability*
- ~ *Baye's rule*

=> **Regression :**

- ~ *Basics of Linear Algebra*
- ~ *Matrix operations*
- ~ *Solving linear equation*
- ~ *Discussion: Linear regression*
- ~ *Discussion: Polynomial regression*
- ~ *PCA intuition*

=> **Hypothesis testing :**

- ~ *Hypothesis testing example*
- ~ *Hypothesis testing p values*
- ~ *Null hypothesis*
- ~ *Z-test*
- ~ *T-test*

=> **Summary :**

- ~ *Course Outro*
- ~ *Future Scope of Statistics*

# Class 6th Physics

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Class-6th-Physics>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of science and its importance in our daily life. It will make the foundation strong for the higher classes. The Physics section focuses on various concepts related to Motion, Light, Electricity, Magnetism, etc.

## Course Features :-

- => Self Paced Videos
- => Completion Certificate

## What you will learn :-

- => Fun with Magnets
- => Electricity and Circuits
- => Light, Shadows and Reflections
- => Motion and Measurement of Distance

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => Jawala Prakash :
- ~

## Curriculum details :-

- => Fun with Magnets :
  - ~ Lecture 1 : Introduction Preview
  - ~ Lecture 2 : Magnet And its Discovery Preview
  - ~ Lecture 3 : Magnetic and Non-Magnetic Material, Poles of a Magnet
  - ~ Lecture 4 : Finding Direction with the help of Magnet, Magnetic Compass
  - ~ Lecture 5 : Making your own Magnet, How to store Magnet
- => Electricity and Circuits :
  - ~ Lecture 1 : Introduction, Electric Cell, Working of Electric Bulb Preview
  - ~ Lecture 2 : Electric Circuit, Electric Conductor and Insulator
  - ~ Lecture 3 : NCERT Question Discussion
- => Light, Shadows and Reflections :
  - ~ Lecture 1 : Light Introduction, Luminous and Non Luminous Object, Transparent, Opaque, Transparent Objects
  - ~ Lecture 2 : Understanding Shadows, Pinhole Camera
- => Motion and Measurement of Distance :
  - ~ Lecture 1 : Introduction, Measurement and Standard unit of measurement
  - ~ Lecture 2 : Measurement of length
  - ~ Lecture 3 : Types of Motion
  - ~ Lecture 4 : NCERT Question Discussion



# Programming in Python

---

Topic Name : DATA SCIENCE

Sub-topic Name : PYTHON

Course link : <https://ineuron.ai/course/Programming-in-Python>

## Course Description :-

This course will teach you the fundamentals of Python Programming. Python is an easy-to-learn programming language that allows you to get started in programming without having any prior programming experience. This course is designed for Beginners who have never coded before, as well as experienced student programmers who wish to learn Python to expand their career choices.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction to Python programming language
- => Features of python
- => Application of python
- => Integrated development environment
- => Introduction to python variables
- => Introduction to Data types
- => Introduction to python operators and Strings
- => Python Programs

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Shubham Sharma :

*~ Having 3+ years of DataScience and Web Development expertise, proficient in data modelling, data preprocessing as well as scripting languages Python and PHP. I've also worked as a mentor and a freelancer. Machine Learning and Natural Language Processing (NLP) are two of my areas of expertise.*

## Curriculum details :-

=> Introduction to Python :

- ~ What is Python?
- ~ History of Python
- ~ Features of Python
- ~ Applications of Python
- ~ Why should you learn python?

=> System Setup :

- ~ Colab

=> Python Basics :

- ~ What is IDE?
- ~ Why IDE is used?
- ~ Advantages of using an IDE
- ~ Offline editor( Python IDE)
- ~ Online editor (<https://www.onlinegdb.com/>) Linux distribution
- ~ Execute your first python program (print Hello world)
- ~ Python Indentation
- ~ Comments & Statements
- ~ Keywords & Identifiers

~ Types of errors In Python

## => Python Variables :

~ What is Variables?

~ Creating Variables

~ Rules for creating variables

~ Type function

~ Type Casting

~ Single Variable Name & Multi Variable Name

~ Unpack collections

~ Changing values of variable

~ Deleting variable

~ Practical:- Print book names using variables

## => Data Types :

~ What are Data Types?

~ Numeric data type

~ Sequence data type

~ Boolean data type

~ Set

~ Dictionary

~ Practical :- Creating a Hash table for students details

## => Python Operators :

~ Relational Operators

~ Relational Operators with Strings

~ Chaining of Relational Operators

~ Logical Operators

~ Special behaviour of == and !=, ===

~ How logical operators work with boolean types?

~ How logical operators work with Non-Boolean types?

~ Introduction to Bitwise Operators

~ Six types of Bitwise Operators

~ Assignment Operators

~ Various types of assignment operators

~ Compound Operators

~ Identity Operators

~ Membership Operators

~ Precedence and Associativity

~ Boolean Operators

## => Python Strings :

~ Defined a Single line string

~ Defined a Multiple Line string

~ Slicing

~ Modify String

~ String Concatenation

~ String Formatting

~ Escape Character

~ String Methods

## => Python Programs :

~ How to print Palindrome Number ?

~ Reverse a String

~ Count Vowels in a given string

~ Counts words in a given string

~ Fibonacci series

~ Armstrong number

~ Find the number of Prime numbers present in the list of integers.

~ Leap year

~ Wish a marriage anniversary to your mom dad using Python / Greetings messages.

# Class 6th Biology

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Class-6th-Biology>

## Course Description :-

The Science Syllabus is elegantly designed such that it introduces the basic concepts of Science and its importance in our daily life. It will make the foundation strong for the higher classes. The Biology section focuses on concepts like food, bodily movements, surroundings of living organisms, garbage, etc.

## Course Features :-

=> Self Paced Videos

=> Completion Certificate

## What you will learn :-

=> Food - Where does it come from?

=> Components of Food

=> Getting to know Plants

=> Body Movements

=> The living organisms and their surroundings

=> Water

## Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

## Instructors :-

=> Dr Nishtha Jain :

*~ I am a doctor by profession but a teacher by passion. I have been into the teaching profession for the last 3 years. I have been and am still a mentor for various courses which include technical as well as non-technical ones. These include MS-Excel, Tableau, Computer basics, Biology, English, etc. I love to learn, explore and share my knowledge to whatever extent possible. Being an ardent educator, I have always helped all my students and will continue to do the same.*

## Curriculum details :-

=> Food - Where does it come from? :

- ~ Lecture 1 : Introduction and food sources Preview
- ~ Lecture 2 : Major food components & Foods from Plants Preview
- ~ Lecture 3 : Animals' characteristics and their foods Preview
- ~ Lecture 1 - NCERT Solutions

=> Components of Food :

- ~ Lecture 1 : Introduction & Different tests
- ~ Lecture 2 : Role of various nutrients - Vitamins
- ~ Lecture 3 : Minerals, Roughage & Water
- ~ Lecture 4 : Balanced Diet & Deficiency Diseases
- ~ Lecture 1 - NCERT Solutions

=> Getting to know Plants :

- ~ Lecture 1 : Herbs, Shrubs, Trees, Creepers and Climbers
- ~ Lecture 2 : Parts of a plant - Stem
- ~ Lecture 3 : Parts of a plant - Leaf
- ~ Lecture 4 : Parts of a plant - Roots, Flowers & Pollination
- ~ Lecture 1 - NCERT Solutions
- ~ Lecture 2 - NCERT Solutions

=> Body Movements :

- ~ Lecture 1 : Human Body (Bones, Joints and Cartilage)
- ~ Lecture 2 : Types of Joints
- ~ Lecture 3 : - Xray, Skull, Shoulder bones, Ribs and Rib cage
- ~ Lecture 4 : - Bones of hands, Pelvic Bones, Muscles - 1
- ~ Lecture 5 : Bones of hands, Pelvic Bones, Muscles - 2
- ~ Lecture 6 : Gaits of Animals - Earthworm & Snail

- ~ *Lecture 7 : Gaits of Animals - Cockroaches & Birds*
- ~ *Lecture 8 : Gaits of Animals - Fish & Snakes*
- ~ *Lecture 1 - NCERT Solutions*

=> The living organisms and their surroundings :

- ~ *Lecture 1 : Introduction, Basic Characteristics*
- ~ *Lecture 2 : Types of habitats and adaptations*
- ~ *Lecture 3 : Terrestrial habitats*
- ~ *Lecture 4 : Aquatic habitats*
- ~ *Lecture 1 - NCERT Solutions*

=> Water :

- ~ *Lecture 1 : Introduction, Sources & Uses of Water*
- ~ *Lecture 2 : - Evaporation, Transpiration*
- ~ *Lecture 3 : - Water Cycle, Clouds, Rainfall*
- ~ *Lecture 4 : Groundwater, Water table*
- ~ *Lecture 5 : Floods, Droughts and their consequences*
- ~ *Lecture 6 : - Water shortage, Ways of conserving water and increasing its availability*
- ~ *Lecture 1 - NCERT Solutions*
- ~ *Lecture 2 - NCERT Solutions*

=> Garbage In, Garbage Out :

- ~ *Lecture 1 : Introduction*
- ~ *Lecture 2 : Biodegradable and Non-biodegradable wastes*
- ~ *Lecture 3 : Vermicomposting*
- ~ *Lecture 4 : Plastics, Paper recycling, minimise garbage & plastics*
- ~ *Lecture 1 - NCERT Solutions*
- ~ *Lecture 2 - NCERT Solutions*

# JavaScript Marathon

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : JAVASCRIPT

Course link : <https://ineuron.ai/course/JavaScript-Marathon>

## Course Description :-

This community course will help you to grab the fundamentals of JavaScript.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Introduction
- => Running Javascript in Browser
- => Strings & Numbers
- => var, let & const
- => Data Types
- => Type Conversions
- => Swap Numbers
- => String Handling
- => String Searching

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Welcome to the Course :
  - ~ *Welcome to the Course Preview*
- => Getting Started with JavaScript :
  - ~ *Introduction to JavaScript*
  - ~ *Introduction and lexical structure*
  - ~ *Mystery of numbers*
  - ~ *Strings and Regex*
  - ~ *Symbols and Global Objects*
  - ~ *Conversion Confusion*
  - ~ *How conversion works in javascript*

# Machine Learning and Data Science Bootcamp

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-and-Data-Science-Bootcamp>

## Course Description :-

This is a data science detailed course where you will learn all the stack required to work in data science, data analytics and big data industry.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Getting started with Machine Learning
- => Installation for Windows and MAC
- => Python Quick Refresher
- => Mastering NUMPY Library
- => Mastering PANDAS Library
- => Mastering MATPLOTLIB Library
- => Mastering SEABORN Library
- => Multi index Matrix
- => Portfolio Project - Classic 911 analysis
- => Data preprocessing for Machine Learning
- => Supervised, Unsupervised and Reinforcement Learning
- => Linear regression algorithm
- => Portfolio Project - Housing dataset analysis
- => Decision Tree Regression Algorithm
- => K-Nearest Neighbors Algorithm
- => Support Vector Machine Classifier
- => Nave Bayes Algorithm
- => Neural Network and Deep Learning

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Hitesh Choudhary :

~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!

## Curriculum details :-

- => Getting started with Machine Learning :
  - ~ Why Machine Learning and How it works
  - ~ Where we are using Machine Learning
  - ~ What is machine learning
- => Installation for Windows and MAC :
  - ~ what you need - Windows
  - ~ Installing python Anaconda and setup - Windows
  - ~ Let 27s collect our tools first- MAC
  - ~ Installing python and anaconda - MAC

=> Python Quick Refresher :

- ~ *Python datatypes*
- ~ *Making decisions in python*
- ~ *Loops in python*
- ~ *Practice Python - 1 - Average list*
- ~ *Practice Python - 2 - Palindrome*
- ~ *Practice Python - 3 - Identity matrix*
- ~ *Practice Python - 4 - Multiplication table*
- ~ *Practice Python - 5 - Second largest*
- ~ *Practice Python - 6 - merging lists*

=> Mastering NUMPY Library :

- ~ *Anaconda and python notebooks*
- ~ *What is numpy*
- ~ *Basics of numpy - generating matrix*
- ~ *Numpy - matrix operations*
- ~ *Numpy file paths and copy issues*
- ~ *Numpy 2D selection*
- ~ *Numpy conditional returns*
- ~ *Numpy Mean Deviation 2C dot and cross products*

=> Mastering PANDAS Library :

- ~ *Introduction to PANDAS library*
- ~ *Handle series with Pandas*
- ~ *DataFrames in Pandas*
- ~ *Subselection using pandas*
- ~ *Conditional selection in PANDAS*
- ~ *Multiple conditions in PANDAS*
- ~ *basics of datacleanup*
- ~ *Merging the data and operations*
- ~ *Reading and writing files*

=> Mastering MATPLOTLIB Library :

- ~ *Introduction to MATPLOTLIB*
- ~ *Our first linear graph using MATPLOTLIB*
- ~ *plotting histograms in matplotlib*
- ~ *plotting ads data with stackplot*
- ~ *Pie chart for ads*

=> Mastering SEABORN Library :

- ~ *Introduction to SEABORN*
- ~ *Plotting graphs with SEABORN*
- ~ *Factor plot and Fat consumption data*
- ~ *Swarmplot with IRIS dataset*

=> Multi index Matrix :

- ~ *Multilevel indexing*

=> Portfolio Project - Classic 911 analysis :

- ~ *Setup of resource files and python notebook*
- ~ *Loading dataset and verifying it*
- ~ *Answering top 3 questions in dataset*
- ~ *Python knowledge with Pandas*
- ~ *working with data time of python*
- ~ *Group the data by Days and months*

=> Data preprocessing for Machine Learning :

- ~ *Data preprocessing basics for Machine Learning*
- ~ *importing dataset and libraries*
- ~ *Separating dependent and independent matrixes*
- ~ *Imputation of missing values*
- ~ *Dummy matrix and one hot encoder*
- ~ *Preparing test and training dataset*
- ~ *Feature scaling - Might be needed*

=> Supervised, Unsupervised and Reinforcement Learning :

- ~ *Supervised, Unsupervised and Reinforcement Learning*

=> Linear regression algorithm :

- ~ *Linear Regression theory*
- ~ *Importing libraries and dataset*
- ~ *creating test and training data sets*
- ~ *Training the machine for prediction*
- ~ *plotting graphs on training and predictions*

=> Portfolio Project - Housing dataset analysis :

- ~ *Housing dataset analysis using Linear Regression*

=> Decision Tree Regression Algorithm :

- ~ *How decision tree Algorithm works*
- ~ *Loading our dataset for Decision tree*
- ~ *Predicting values using Decision Tree algorithm*

=> K-Nearest Neighbors Algorithm :

- ~ *K-Nearest Neighbors theory*
- ~ *loading data and libraries*
- ~ *splitting data into training and test sets*
- ~ *Applying KNN confusion matrix and plotting*

=> Support Vector Machine Classifier :

- ~ *Theory of Support Vector Machine SVM*
- ~ *Loading libraries and dataset*

- ~ *Test and training data with feature scaling*
- ~ *Confusion matrix and stackoverflow debugging*

=> Nave Bayes Algorithm :

- ~ *What is Bayes theorem*
- ~ *Naive bayes and scikit docs for it*
- ~ *importing dataset for NB*
- ~ *data preprocessing for NB*
- ~ *prediction and confusion matrix for NB*

=> Neural Network and Deep Learning :

- ~ *Neural Network and Deep learning*
- ~ *Installing tensorflow*



# Basics of Computer Education

---

Topic Name : K12

Sub-topic Name : CLASS6

Course link : <https://ineuron.ai/course/Basics-of-Computer-Education>

## Course Description :-

In this course, you will learn the basics of computers. Having decent computer knowledge will help you build your foundation for emerging technology. Upon completing this course, you will be able to navigate through the ecosystem of the modern computer knowing the underlying technology behind it.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Introduction of Computers
- => Components of Computer
- => Features of Computer
- => Overview about Input and output devices
- => Overview of operating system
- => Memory management
- => How to Operate Computer
- => Introduction of Microsoft office
- => Paint
- => Email Operation
- => YouTube Operation

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Computer :
  - ~ *What is a Computer?*
  - ~ *Functions of Computer*
  - ~ *History of Computer*
  - ~ *Classification of Computer (Based on Size, Work and Purpose)*
  - ~ *Feature of Computer*
  - ~ *Applications of Computer*
- => Components of Computer :
  - ~ *I/O (Input/Output)*
  - ~ *CPU*
  - ~ *GPU*
  - ~ *Hard Disk*
  - ~ *SSD*
  - ~ *Memory Unit*
  - ~ *Primary Memory*
  - ~ *RAM*
  - ~ *ROM*
  - ~ *EPROM*
  - ~ *EEPROM*
  - ~ *PROM*
  - ~ *Secondary Memory*

=> Input / Output Devices :

- ~ Keyboard
- ~ Mouse
- ~ Light Pen
- ~ Touch Screen
- ~ Printer
- ~ Scanner

=> Operating System (OS)

:

- ~ What is Operating System
- ~ Types of Operating system
- ~ Single User Operating System
- ~ Multi User Operating System
- ~ Multitasking Operating System
- ~ MultiProgramming Operating System
- ~ Mobile Operating system

=> How to Operate PC :

- ~ Open
- ~ Restart
- ~ Shutdown
- ~ Sleep
- ~ Other Functions

=> Folder Operation :

- ~ Create
- ~ Delete
- ~ Open
- ~ Rename
- ~ Cut
- ~ Paste
- ~ Copy
- ~ Shortcut Keys

=> Functions of Operating System :

- ~ File Management
- ~ Process Management
- ~ I/O Management
- ~ Storage Management

=> Memory Management :

- ~ Bit
- ~ Bytes
- ~ Kilobytes
- ~ Megabytes
- ~ Gigabytes
- ~ Terabytes

=> Microsoft Office :

- ~ What is Microsoft Office?
- ~ Microsoft Word
- ~ Microsoft Excel
- ~ Microsoft Powerpoint
- ~ Notepad

=> Paint :

- ~ Paint Exploration

=> Email Operation :

- ~ Email Operation

=> YouTube Operation :

- ~ Keyword Search
- ~ Channel Creation
- ~ Channel Search
- ~ Upload Video in Channel

# Complete ReactJs Developer Bootcamp

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : REACT

Course link : <https://ineuron.ai/course/Complete-ReactJs-Developer-Bootcamp>

## Course Description :-

This course will teach you React.js in a hands-on manner, utilising all of the most up-to-date patterns and best practises. To become a React.js developer, you will master all of the foundations as well as advanced ideas and associated subjects. This course will provide you with a wealth of essential material and expertise, whether you are new to React or have some basic React experience.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => React templates
- => React states
- => React hooks
- => React context API
- => Firebase integration
- => Redux Apps

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction :
  - ~ Into to REACTJS course
- => After launch updates :
  - ~ React router v6
- => Getting started with ReactJS :
  - ~ How to use exercise files
  - ~ What is react and myths
  - ~ Tools that we need
- => Going All classic :
  - ~ Section 2 introduction
  - ~ Finishing the hello world task
  - ~ Delete and recreate everything
  - ~ Adding CSS to our Hello World
  - ~ Everything in its own file
  - ~ Reusable components
- => Create a react template :
  - ~ Section 3 introduction
  - ~ Understand the existing template
  - ~ Move navbar and understand the errors
  - ~ Convert the HTML template into React App
  - ~ Reusable Card and Assignment
- => Getting friendly with states :
  - ~ Section 4 introduction
  - ~ What are props and states

- ~ *Preparing the state based applications*
- ~ *Complete counter application*
- ~ *Assignment for counter app*

=> Building a Tic Tac Toe :

- ~ *section 5 Introduction*
- ~ *Your need to study first*
- ~ *Preparing the Tic Tac Toe*
- ~ *Sending icons from components*
- ~ *Setup layout for tictactoe*
- ~ *Game is almost working*
- ~ *Finishing tictactoe and assignment*

=> Learn React Context API with projects :

- ~ *Section 6 Introduction*
- ~ *The problem that contextAPI solves*
- ~ *Detail on Context and Provider*
- ~ *Detail on Consumer in contextAPI*
- ~ *Understand the working of dark and light mode*
- ~ *Creating a theme Toggler with Context API*
- ~ *Finishing the theme switcher app*

=> App with Context API with reducers and actions :

- ~ *Section 7 introduction*
- ~ *What are we building here*
- ~ *Create brain of the application*
- ~ *useReducer for our app*
- ~ *Add an input form*
- ~ *Sending a dispatch*
- ~ *Display the context data and dispatch*

=> Local storage and useEffect hooks :

- ~ *Section 8 introduction*
- ~ *Introducing the Effect hook*
- ~ *A form to submit the data*
- ~ *Looping through all the values*
- ~ *Hooks and local storage in action*

=> Learn to handle API :

- ~ *Section 9 introduction*
- ~ *Learn to read docs for API*
- ~ *lets read Axios docs*
- ~ *Drill down the API*
- ~ *Extracting information from API*

=> Designing a shopping cart API :

- ~ *Section 10 introduction*
- ~ *A walk through Pexels and JSON*
- ~ *Add item to the cart*
- ~ *Buy item and remove item*
- ~ *Fetching photos from API*
- ~ *Store everything in state*
- ~ *Card for every product*
- ~ *Create cart section*
- ~ *Bring the shop together*
- ~ *Removing the duplicate*

=> Firebase with Github App :

- ~ *Section 11 introduction*
- ~ *What we are about to build*
- ~ *React Router crash course*
- ~ *Your tour to configure firebase*
- ~ *Read firebase docs with me*
- ~ *Creating components for firebase app*
- ~ *Bring in the react router*
- ~ *Headers and Footers*
- ~ *Conditional rendering in Navbar*
- ~ *Adding firebase configuration*
- ~ *User Signup in firebase*
- ~ *Logout and signin user*
- ~ *User card component*
- ~ *Repo component*
- ~ *Home page and finish the app*

=> Firebase real time database :

- ~ *Section 12 introduction*
- ~ *A challenge application*
- ~ *Firebase real time database*
- ~ *Setting context and actions*
- ~ *Creating reducers for contact*
- ~ *Header and Footer tasks*
- ~ *How to upload image in firebase storage*
- ~ *Add and update contact in firebase*
- ~ *Add or update finder*
- ~ *Update star and delete contact*
- ~ *Use dispatch and FIXME*
- ~ *Get all data from firebase*
- ~ *Loop through firebase object*
- ~ *Firebase finale and assignment*

=> Bonus-Redux App :

- ~ 3 Principles of redux
- ~ Bring in the central state
- ~ Actions make redux simpler
- ~ Reducer - brain part of app
- ~ Component dispatching the info
- ~ 2 most important method for Redux
- ~ Provider to give access of store
- ~ Finally creating that store

=> More bonus stuff -Extra production tips :

- ~ Axios optimise API calls

=> Bonus updates :

- ~ React 18 updates

# Kotlin

---

Topic Name : MOBILE DEVELOPEMENT

Sub-topic Name : KOTLIN

Course link : <https://ineuron.ai/course/Kotlin>

## Course Description :-

This course will teach you Kotlin programming. Since we start from the beginning, this course is ideal for total beginners. Several code tasks will allow you to put everything you have learned into practise. So, at the end, you'll be able to write your own Kotlin apps. If you're an Android developer, this course will help you learn the basics of the language. You'll be able to develop more powerful Android applications with Kotlin because it allows you to maintain a cleaner and more expressive code base, leverage notions that go beyond even Java 8, and maintain a cleaner and more expressive code base.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Kotlin basics
- => Conditionals in Kotlin
- => Advanced Data types
- => Data structures in Kotlin
- => Object oriented Kotlin
- => Making android apps with Kotlin

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting started with Kotlin :

- ~ Course promo
- ~ Introduction to Course
- ~ Kotlin - Face to Face
- ~ Getting the tools to start with kotlin
- ~ Setting up IDE for Kotlin

=> Basics of Kotlin :

- ~ Getting the tools to start with kotlin
- ~ Numbers in Kotlin
- ~ Strings in Kotlin
- ~ Handling nulls in kotlin
- ~ Our very first file in Kotlin

=> Conditionals in Kotlin :

- ~ If else conditionals in Kotlin
- ~ Finding Odd and Even
- ~ Designing a grading system
- ~ Finding biggest of all
- ~ WHEN in kotlin
- ~ Syntactic sugar for if else and when

=> Advance datatypes in Kotlin :

- ~ Basics of Array in Kotlin
- ~ Array list in kotlin

- ~ *For loop basics*
- ~ *While loop in Kotlin*
- ~ *Selecting valid users only with continue*
- ~ *Functions in Kotlin*
- ~ *Integrating Java with Kotlin*
- ~ *Generating Email list with Kotlin*

=> Object oriented Kotlin :

- ~ *Classes and objects*
- ~ *Creating methods in class*
- ~ *Constructor in Kotlin*
- ~ *Default parameters*
- ~ *Inheritance, open and override in Kotlin*
- ~ *Abstract classes in Kotlin*
- ~ *Interfaces in Kotlin*
- ~ *Data class in Kotlin*
- ~ *Reserve airline seats using enums*

=> Making Android Apps with Kotlin :

- ~ *Getters and setter in bank application.mp4*
- ~ *Limiting the access in classes*
- ~ *Our first hello world Toast Application in Android*
- ~ *Setting things for Calculator App in Android*
- ~ *Creating a calculator App- Finish*

# GCP Projects

---

Topic Name : CLOUD

Sub-topic Name : GCP PROJECT

Course link : <https://ineuron.ai/course/GCP-Projects>

## Course Description :-

GCP is one of the most used and fastest-growing cloud platforms in the cloud industry currently. This course will get hands-on experience in building and implementing various real-time ML, DL & NLP-based projects with GCP cloud services.

## Course Features :-

- => Real-time project implementation
- => Quizzes
- => Assignment problems
- => Downloadable resources
- => Completion certificate

## What you will learn :-

- => GCP IAM and Security
- => GCP Storage & Database Services
- => GCP Management Tools
- => GCP AI Stack
- => Real-time Project implementation on ML, DL & NLP

## Requirements :-

- => Prior Understanding of GCP
- => GCP Account
- => Prior understanding of ML, DL and NLP
- => A System with a decent internet connection
- => Your dedication

## Instructors :-

=> Khushali Shah :

~ A data scientist having rich experience working with MNCs and start-ups in the field of data science and machine learning. She has expertise in Chatbot development for various domains & been developing professionally for 6+ years with diverse job history. She also had positions in software module development, web app development, functional designs, requirement gathering, client interaction, and server setup/admin & can help everywhere in the stack; she loves wearing multiple hats to an extent. She also believes in enhancing her skills by training and learning new things day by day.

## Curriculum details :-

=> Projects :

- ~ Guest Book Project Preview
- ~ Overview Preview
- ~ Vertex AI image classification
- ~ Custom image classification
- ~ Install SDK
- ~ authenticating user
- ~ training
- ~ user\_message



# Data Structure and Algorithm Job Preparation

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA INTERVIEW

Course link : <https://ineuron.ai/course/Data-Structure-and-Algorithm-Job-Preparation>

## Course Description :-

Algorithmic programming techniques are a must-have skill. Learn Algorithms through programming and puzzle solving to advance your Software Engineering or Data Science career. Then, implement each algorithmic problem in this program to ace coding interviews.

## Course Features :-

- => Downloadable resources
- => Roadmap
- => Assignments
- => Quizzes
- => Interview questions
- => Completion certificate

## What you will learn :-

- => Problem solving
- => Analytical skill
- => Design Solution
- => Architecture design
- => Answer Confidently in interview.
- => Upscale your skill as a Developer.

## Requirements :-

- => Prior Knowledge of Data Structure & Algorithms concepts.
- => A System with internet connection.
- => Your dedication

## Instructors :-

- => Priya Bhatia :
  - ~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

- => Maximum Subarray Sum :
  - ~ Given an integer array, find the sum of the largest contiguous subarray within the array. Note that if all the elements are negative, you should return 0. [Preview](#)
- => K Closest Points :
  - ~ Given a list of coordinates, write a function to find k closest points(measured by Euclidean Distance) to the origin. [Preview](#)
- => Finding of Kth Smallest Element in 2D Sorted Matrix :
  - ~ Given an n-by-n matrix of elements that are sorted in ascending order both in the columns and rows of the matrix. Return the kth smallest element of the matrix.
- => Mirror Image in Binary Tree :
  - ~ Given a binary tree, write a function to determine whether the tree is a mirror image of itself. Two trees are mirror images of each other if their root values are the same, and the left subtree is a mirror image of the right subtree.
- => Maximum Product :
  - ~ Given an integer array, return the maximum product of any three numbers in the array. [Preview](#)
- => Intersection Of Elements in an array :
  - ~ Given two arrays, write a function to get the intersection of elements between the two arrays.
- => Diameter Of Tree :
  - ~ Given a binary tree, write a function to determine the diameter of the tree, which is the longest path between any two nodes.
- => Maximum Length Of Common Subarray :
  - ~ Given two arrays, return the maximum length of the common subarray within both arrays. For example, if two arrays are [2, 5, 7, 9] and [1, 2, 5, 8], then the function return two as the length of the maximum common subarray is [2,5].
- => Finding of Peak Elements :
  - ~ Given an integer array, find the peak element and return its index. Here, Peak Element is an element that is strictly greater than its neighbours. If an array contains multiple peaks, return the index to any of the peaks.
- => Top K Frequent Elements :
  - ~ Given an integer array and integer k, return the k most frequent elements. For example, if an array is [2, 2, 3, 3, 3, 5] and k = 2 then function return [2, 3].

=> Permutation Of list :

~ Given a list of one or more distinct integers, write a function to generate all the permutations of those integers.

=> Combinations of k numbers :

~ Given an integer  $n$  and an integer  $k$ , output a list of all the combinations of  $k$  numbers chosen from 1 to  $n$ . For example, if  $n = 3$  and  $k = 2$  function should return  $[1,2],[1,3],[2,3]$ .

=> Removal Of kth Node from end :

~ Given a linked list, return the head of the same linked list but with the  $k$ th node from the end of the linked list removed. For example given the linked list 2->3->7->1->4 and  $k = 3$  then function remove the 7 node and return the linked list 2->3->1->4.

=> Length Of Longest Path :

~ Given the  $m$ -by- $n$  matrix with positive integers, find the length of the longest path of increasing integers with the matrix. For example if the matrix is  $\begin{bmatrix} 1, & 2, & 3 \\ 4, & 5, & 6 \\ 7, & 8, & 9 \end{bmatrix}$  then function should return 5 since the longest path would be 1-2-5-6-9

=> Reverse Linked List from the given start and end position :

~ Given the head of the singly linked list and two integer start and end where  $start \leq end$ , reverse the nodes of the list from position start to position end and return the reversed linked list.

=> Finding of Cycle in Linked List :

~ Given the head of a linked list, determine if the linked list has a cycle in it or not.

=> Rotate Array :

~ Given an array, rotate the array to the right by  $k$  steps where  $k$  is non-negative.

=> Longest Substring Without Repeating Characters :

~ Given a string  $s$ , find the length of the longest substring without repeating characters. For example, if  $s = "abcdabc"$  the result should be "abcd" with the length of 4.

=> Number of Friend Group :

~ Say that there are  $n$  people. If person  $X$  is a friend with person  $Y$ , and person  $Y$  is a friend with person  $Z$ , then person  $X$  is considered to be an indirect friend of person  $Z$ . Define a friend group to be any group that is either direct or indirect friends. Given an  $n$ -by- $n$

~ Adjacency matrix  $N$  where  $N[i][j]$  is one of the persons  $i$  and person  $j$  are friends and is zero otherwise, write a function to get how many friend groups exist.

=> Greedy Algorithms vs Dynamic Programming :

~ As you are aware of the fact that both Greedy Algorithms and Dynamic Programming are used to solve the optimization problem, then what's the difference between these two approaches and how do we decide when we should go for what approach?

=> QuickSort :

~ In QuickSort, sorting of  $n$  numbers,  $n/10$ th element is selected as Pivot using  $O(\log N)$ . Then, what will be the worst-case time complexity of QuickSort?

=> Correlation :

~ Given two lists  $X$  and  $Y$ , return their correlation.

=> QuickSort vs MergeSort :

~ Out of MergeSort and QuickSort, which one do you think is more suitable for practical use cases and why?

~ Why is QuickSort preferred for Array and MergeSort for LinkedList?

=> Median Calculation :

~ Given a continuous stream of integers, write a class with functions to add new integers to the stream and a function to calculate the median at any time.

=> Length Of longest well-formed substring :

~ Given a string with left and right parentheses, write a function to determine the length of the longest well-formed substring. For example, if the input string is " $()()()$ " then the function should return four since the longest well-formed substring is " $()()()$ ".

=> Sum to the target Number :

~ Given a target number, generate a random sample of integers that sum to that target that are also within a standard deviation of the mean.

=> Sort an array of 0s, 1s and 2s :

~ Given an array of size  $N$  containing only 0s, 1s and 2s, sort the array in ascending order using optimized time complexity.

=> Detection Of Cycle in Undirected Graph :

~ Given an undirected graph with  $V$  vertices and  $E$  edges, check whether it contains any cycle or not.

=> Detect Cycle in Directed Graph :

~ Given a directed graph, check whether the graph contains at least one cycle, else return False

=> Binary Tree Level Order Traversal :

~ Given the root of a binary tree, return the level order traversal of its nodes' values.

=> Binary Tree Zigzag Level Order Traversal :

~ Given the root of a binary tree, return the zigzag level order traversal of its node values.

=> Intersection Of Two Linked Lists :

~ Given the head of two singly linked lists, return the node at which the two lists intersect. If the two linked lists have no intersection at all, return null

=> Rotate Image :

~ Given an  $n$ -by- $n$  matrix representing an image, rotate the image by 90 degrees(clockwise).

=> Smallest Number Of Perfect Squares :

~ Given positive integers  $n$ , find the smallest number of perfect squares that sum up to  $n$ . For example if  $n = 13$ , you should return 2, since  $13 = 9 + 4$

=> Application of Graph :

~ Demonstrate any real-world application of graph?

=> Anagrams :

~ Given an array of strings, return all groups of strings that are anagrams.

=> Leaf at the same level :

~ Given a Binary Tree, check if all the leaves are at the same level or not.

=> Reverse Words in a String :

~ Given an input string  $s$ , reverse the order of the words.

=> Spiral Matrix :

~ Given an  $m$ -by- $n$  matrix, return all the elements of the matrix in spiral order.

=> Sorted Input array :

~ Given an array of integers that is already sorted in non-decreasing order, find two numbers such that they add up to a specific target number. Let these two numbers be  $numbers[index\_1]$  and  $numbers[index\_2]$  where  $1 \leq first < second \leq numbers.length$

=> Array vs Linked List :

~ How is an Array different from Linked List?

=> Lowest Common Ancestor :

~ Given a Binary Tree, find the lowest common ancestor in a binary tree

=> Next Greater Element :

~ Given an array, find the next greater element for every element. For example  $[3, 4, 1, 10]$ , the next greater element for each element  $[4, 10, 10, -1]$ .

=> Edit Distance :

~ Given two strings  $str1$  and  $str2$ , find the minimum number of edits required to convert  $str1$  into  $str2$ . For example  $str1 = "prya"$  and  $str2 = "Priya"$ , we can convert  $str1$  to  $str2$  by inserting 'i'.

=> Binary Tree is BST or not :

~ Given a Binary Tree, check if a Binary Tree is BST or not.

=> Maximum number of nodes :

~ What is the maximum number of nodes in a binary tree of height  $k$ ?

=> Single Number :

~ Given an array of integers, every element appears twice except for one. Find that single element in an array. For example  $arr = [1, 3, 3, 2, 2]$ , output = 1

=> Linked List :

~ What is the primary advantage of Linked List?

=> Height Of Binary Tree :

~ Given a Binary Tree, calculate the height of the Binary Tree

=> Remove Duplicates from Sorted Array :

~ Given an array of integers sorted in non-decreasing order, remove the duplicates in place such that unique elements appear only once, and the relative order should be kept the same.

=> Build Heap :

~ What is the time complexity of building a heap, and also, Give a mathematical intuition behind the time complexity of Build Heap?

=> Merge Sorted Array :

~ Given two integer arrays,  $arr1$  and  $arr2$ , sorted in non-decreasing order, you have to merge these two arrays into one sorted array in non-decreasing order.

=> Heap :

~ What is the advantage of the heap over a stack?

=> Sorting :

~ What is the meaning of a stable and unstable sorting algorithm? Demonstrate it with the help of some sorting algorithm.

=> Matrix Multiplication :

~ Can we do the task of matrix multiplication in less than  $O(n^3)$  time complexity? If yes, then how and what's the optimized time complexity we can get from that?

=> Inplace vs Outplace Sorting Algorithm :

~ Can you explain the difference between in place and outplace sorting algorithms, and is there any algorithm you know so far which is outplace sorting algorithm?

=> Non-Comparison Sorting Algorithm :

~ Can you explain what a non-comparison-based sorting algorithm is with an example?

=> Tree vs Graph :

~ What is the difference between Tree and Graph-based Data Structure?

=> Topological Sort :

~ Can you explain the topological sort in a graph and where it is used practically?

=> Longest Common Prefix :

~ Write a function to find the longest common prefix string amongst an array of strings. For example,  $str = [Priya, Priyanka, Priyanshu]$ , then the output should be "Priya".

=> Palindrome in Linked List :

~ Given the head of a singly linked list, return true if it is a palindrome.

=> Minimum Value in Stack :

~ Design a Stack Data Structure that supports Push, Pop, Top and give us the minimum value in a stack in constant time.

=> Longest Palindromic Substring :

~ Given a string  $s$ , return the longest palindromic substring in  $s$ . For example :  $str = "babad"$ , so output is "bab".

=> Kth Smallest Value in Binary Search Tree :

~ Given the root of BST and an integer  $k$ , return the  $k$ th smallest value

=> Divide two integers :

~ Given two integer dividends and divisor, divide two integers without using multiplication, division and mod operator.

=> Positive Missing Integer :

~ Given an unsorted integer array, return the smallest positive integer. For example :  $arr = [1, 2, 0]$  output is 3.

=> Stable vs Unstable Sorting Algorithm :

~ If the user's requirement is a minimum number of swaps, then which sorting algorithm should we prefer?

=> Binary Search :

~ Can we implement Binary Search in Linked List? If yes, then how and If not, then is there any alternative of Binary Search in Linked List which gives

*almost the same time complexity like Binary Search provides us.*

=> Complete vs Almost Complete Binary Tree :

~ *What is the difference between Complete vs Almost Complete Binary Tree?*

=> Heap Data Structure :

~ *When we should go for Minheap or Maxheap based Data Structure.*

=> Huffman Coding :

~ *What is the application of Huffman Coding?*

=> Simple vs Multigraph :

~ *What is the difference between Simple Graph and Multi Graph? Which type of graph is used practically.*

=> Generate Parentheses :

~ *Given  $n$  pair of parenthesis, write a function to generate all well-formed combinations of parentheses.*

=> Factorial Trailing Zeros :

~ *Given an integer  $n$ , return the number of trailing zeros in  $n!$*

=> Spanning Tree in Complete Graph :

~ *Given  $n$ , which indicates the number of vertices in a graph, how we can get the number of Spanning Tree in Complete Graph.*

=> Graph Connected or Not :

~ *You are given a graph. How can you determine whether the graph is connected or not?*

=> DFT vs BFT :

~ *What Data Structure is internally used for the implementation of DFT and BFT?*

=> MergeSort :

~ *In MergeSort, if the input is "A" sorted subarrays of each size "B", Then what is the time complexity of a single sorted array.*

=> Cycle in Graph :

~ *Given a graph, check whether it contains a cycle or not?*

=> Sum Greater than 1000 :

~ *Given a sorted array of  $n$  elements, find any two elements such that the sum of an element "A" and "B" is greater than 1000.*

=> Maximum Number of Vowels in a Substring :

~ *Given a string  $s$  and an integer  $k$ , return the maximum number of vowel letters in any substring  $s$  with length  $k$ .*

=> Complete Binary Tree :

~ *Which data structure is preferable to store the complete binary tree and why?*

=> Minimum nodes in Binary Tree :

~ *What is the minimum number of nodes that a binary tree can have?*

=> Doubly Linked List :

~ *Illustrate any real-life application of Doubly Linked List end to end.*

=> Analysis in Algorithm :

~ *Why do we need to do an algorithm analysis?*

=> AVL Tree vs BST :

~ *How can AVL Tree be useful in various operations as compared to BST?*

=> B-Tree :

~ *Where most of the time B-Tree based data structure is used frequently and how?*

=> Interpolation Search :

~ *Usually, we studied Linear and Binary Search. Do you have an idea about interpolation search and how it's working?*

=> Divide and Conquer vs Dynamic Programming :

~ *Can you explain the major differences that you have observed between Divide and Conquer vs Dynamic Programming, and how do we decide when we should go for which approach?*

=> Recursion :

~ *Which Data Structure is used internally to perform recursion operations?*

=> Hashing :

~ *What is the worst-case time complexity of searching an element in Hash Table?*

=> Postfix Form :

~ *What is the postfix form of  $(A + B) * (C - D)$*

=> Tree Data Structure :

~ *What is the real-life applications of Tree-Based Data Structure?*

=> Overlapping Of Two Rectangles :

~ *How to find if two given rectangles overlap or not?*

=> First Non-Repeating Character :

~ *Given a string, find its first non-repeating character. For example :  $str = ["mmadlsals"]$ , output should be "d".*

# Stored Procedures in SQL

---

Topic Name : DATA ANALYTICS

Sub-topic Name : SQL

Course link : <https://ineuron.ai/course/Stored-Procedures-in-SQL>

## Course Description :-

In this course you will learn about A stored procedure is a prepared SQL code that you can save, so the code can be reused over and over again

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => Creating a stored procedure with parameters
- => Creating and executing stored procedures with output parameters
- => Salesforce connector
- => Stored procedure

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Stored Procedures in SQL :
  - ~ Sql server installation Preview
  - ~ Creating a stored procedure with parameters part 1 Preview
  - ~ Creating a stored procedure with parameters part 2
  - ~ Creating a stored procedure with parameters part 3
  - ~ Creating and executing stored procedures with output parameters part 1
  - ~ Creating and executing stored procedures with output parameters part 2
  - ~ Advantages of stored procedure

# Finance Complaint

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING PROJECT

Course link : <https://ineuron.ai/course/Finance-Complaint>

## Course Description :-

By taking into account the complaints already made against financial products, we may develop a machine learning (ML) model that can help us determine if freshly made complaints are troublesome or not, allowing the business to move swiftly to address the problem and meet the needs of the consumer.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self-Paced Learning
- => Dashboard Access
- => Course Materials
- => Assignments

## What you will learn :-

- => Real Time Projects
- => Finance Complaint
- => Understand MLOPS best practices
- => EDA and Feature engineering on Financial Complaint registered by customers.
- => Industry standard Machine Learning development
- => Implementation of Continuous Training
- => Deploying Machine Learning Model as an endpoint API
- => Continuous Monitoring and Model Management
- => ML training using Spark on more than 3 Millions records.

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learning

## Instructors :-

=> Avnish Yadav :

~ 3+ years of experience in various domains such as data scientist, data analyst, database developer, and .net developer. Implemented various sophisticated business requirements, performed an analysis of various data to capture insights and hidden patterns. Fine and tuned various regression and classification-based algorithms for prediction. Implemented various ETL pipelines to fulfil the business requirement. Automated various machine learning pipelines such as data loading, data cleaning, data validation, model selection, model tuning, and model monitoring pipeline. Implemented machine learning pipeline in azure machine learning studio. I have a keen interest to solve complicated machine learning problems to fulfil business requirements.

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Project :- Finance Complaint :

- ~ Introduction of Instructor
- ~ Project Overview
- ~ End Notes
- ~ Problem Description
- ~ Understand the application scope
- ~ Tour to existing solution
- ~ End Notes
- ~ Solution Description
- ~ Notebook Walkthrough
- ~ Tour to Architecture diagram
- ~ cost involved
- ~ End Notes
- ~ Structure overview
- ~ Data Ingestion

- ~ Data Validation
- ~ Data Transformation
- ~ Model Training and Tunning
- ~ Model Evaluation
- ~ Model Pusher
- ~ Training Pipeline
- ~ Schedule Training pipeline to Update/Create Model
- ~ Deploy prediction endpoint API
- ~ Instance Prediction
- ~ Batch Prediction
- ~ Performance of Model
- ~ Model Management
- ~ Tracibility and Verifiability and Auditability
- ~ Conclude the project
- ~ Assignments & External Resources

# Text Summarization

---

Topic Name : DATA SCIENCE

Sub-topic Name : NLP PROJECT

Course link : <https://ineuron.ai/course/Text-Summarization>

## Course Description :-

This is an advanced NLP project where we take news data from several sources and use Transformer models from HuggingFace and Fine tune it to generate news summary from complete news.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access
- => Course Materials
- => Assignments

## What you will learn :-

- => Real Time Projects
- => Text Summarization
- => Understand best coding practice in modular way
- => How to use Pre-Trained model from hugging face
- => How to Fine tune a pretrained model
- => Create a custom dataset and dataloader for faster loading of the data
- => Use Training pipeline and Prediction pipeline along with CICD to GCP

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Bharath J P V :

~ Enthusiast Data Scientist with a strong background in Mathematics and Statistics. Completed My Master in Statistics. Have experience teaching Mathematics and Statistics for more than a year. I thought for more than 1000 students and helped them make their careers in their respective fields. I believe in "we rise by lifting others". Following this principle, I hope to make your life easier.

## Curriculum details :-

- => Welcome to the Course :
  - ~ Course Overview
  - ~ Dashboard Introduction
- => Project :- Text Summarization :
  - ~ Introduction of Instructor
  - ~ Project Overview
  - ~ End Notes
  - ~ Problem Description
  - ~ Understand the application scope
  - ~ Tour to existing solution
  - ~ End Notes
  - ~ Solution Description
  - ~ BERT Methodologies
  - ~ Notebook Walkthrough
  - ~ Tour to Architecture diagram
  - ~ Cost involved
  - ~ End Notes
  - ~ Structure overview
  - ~ Data Ingestion
  - ~ Data Validation
  - ~ Data Transformation
  - ~ Model Training and Tuning
  - ~ Model Evaluation
  - ~ Model Pusher
  - ~ Training Pipeline
  - ~ Frontend app design



- ~ *Tour to the cloud and Service Overview*
- ~ *IAM setup*
- ~ *GCP setup*
- ~ *Workflow*
- ~ *Adding Self hosted runner*
- ~ *Conclude the project*
- ~ *Points to improve from current project*
- ~ *Assignments & External Resources*

# Azure Synapse

---

Topic Name : CLOUD

Sub-topic Name : AZURE

Course link : <https://ineuron.ai/course/Azure-Synapse>

## Course Description :-

Microsoft Azure SQL Data Warehouse - You will be able to deploy Azure Synapse Analytics (formerly known as Azure SQL Data warehouse) in Azure Cloud environment. You will have good internal MPP architecture understanding, and so you will be able to analyze your on-premises data warehouse and migrate data to Azure Data Warehouse.

## Course Features :-

- => Self-paced Recording
- => Assignment in all modules
- => Quiz in every module
- => Completion Certificate

## What you will learn :-

- => you will learn how Azure Synapse Analytics enables you to perform different types of analytics through its components that can be used to build Modern Data Warehouses through to advanced analytical solutions.
- => You will learn how Azure Synapse Analytics solves the issue of having a single service to fulfill the broad range of analytics requirements that organizations face today and take a tour of the core application used to interact with the various components of Azure Synapse Analytics.
- => You will learn the various components of Azure Synapse Analytics that enable you to build your analytical solutions in one place.

## Requirements :-

- => A system with internet connection
- => Your dedication
- => Interest to learn

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Azure Synapse :
  - ~ Module introduction Preview
  - ~ why warehouse in cloud?
  - ~ Traditional vs modern warehouse architecture
  - ~ what is synapse analytics service Preview
  - ~ demo create dedicated sql pool
  - ~ demo connect sql pool with ssms
  - ~ demo create azure synapse analytics workspace
  - ~ Demo explore synapse studio v2
  - ~ demo create dedicated sql pool and spark pool from inside synapse studio
  - ~ demo analyse data using dedicated sql pool
  - ~ analyse data using apache spark notebook
  - ~ demo analyse data using serverless sql
  - ~ demo data factory copy tool from synapse integrate tab
  - ~ demo monitor synapse analytics studio
  - ~ azure synapse a game-changer
  - ~ azure synapse benefits
  - ~ summary

# Aptitude Test

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Aptitude-Test>

## Course Description :-

Quantitative aptitude is a test that assesses a person's numerical and problem-solving abilities. This is a common section seen on most competitive examinations. This Aptitude course has been designed to help students get started and succeed in tests and interviews.

## Course Features :-

- => Downloadable resources
- => Completion certificate
- => Quizzes
- => Various tricks to solve aptitude questions the easy way
- => Practice with many different types of questions

## What you will learn :-

- => Numbers & Algebra
- => Percentage
- => Average
- => Time & Work
- => Distance
- => Time & Speed
- => Ratio, Proportion & Mixture
- => English Grammar
- => Verbal Analogy
- => Synonyms
- => Antonyms

## Requirements :-

- => No Prior knowledge
- => A system with internet connection.
- => Your dedication

## Instructors :-

=> Jwala Prakash :

~ I have 4+ years of experience in teaching mathematics and physics for grade 9 and 10.  
I am also an experienced teacher for mathematics aptitude. I have qualified mains exam twice of the most reputed central government exam, staff selection commission(SSC)

=> Jayant Topnani :

~ Having 2+ years teaching experience and have mentored students online & offline across all boards.

## Curriculum details :-

=> Number Systems :

- ~ Numbers & their types
- ~ Prime numbers
- ~ Divisibility
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> HCF & LCM :

- ~ Definition
- ~ Factors & Multiples
- ~ Methods to find HCF, LCM
- ~ HCF & LCM of fractions
- ~ Tricks & Tips
- ~ Formulae
- ~ Previous year questions

=> Simplification :

- ~ BODMAS rule
- ~ Modulus of real number

- ~ Vinculum
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Logical reasoning :

- ~ Clocks
- ~ Calendars
- ~ Seating arrangement
- ~ Coding decoding
- ~ Blood Relations
- ~ Syllogism
- ~ Data sufficiency
- ~ Cubes & Dice
- ~ Series
- ~ Puzzles
- ~ Directions
- ~ Non-verbal reasoning
- ~ English, verbal ability
- ~ Synonyms, Antonyms
- ~ Reading Comprehension
- ~ Error Detection and Correction
- ~ Fill in the blanks
- ~ Cloze test
- ~ Grammar
- ~ Sentence Rearrangement
- ~ Coding
- ~ C++ Language
- ~ SQL
- ~ Competitive programming
- ~ Python
- ~ Data structures & algorithm
- ~ Decimal Fractions
- ~ Definition
- ~ Types
- ~ Operations
- ~ Comparison
- ~ Conversion Decimal to fraction
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Surds and Indices :

- ~ Definition
- ~ Types
- ~ Laws
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Problems on Ages :

- ~ Introduction
- ~ Questions
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Time Speed Distance :

- ~ Introduction
- ~ Average speed
- ~ Relative Speed
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Time and Work :

- ~ Basics & Concepts
- ~ Questions
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Boats and Streams :

- ~ Basics & Concepts
- ~ Questions
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Pipes and Cisterns :

- ~ Introduction
- ~ Questions
- ~ Formulae
- ~ Tricks & Tips
- ~ Previous year questions

=> Progressions :

- ~ AP, GP, HP Basics
- ~ Sequence & Series Difference
- ~ Questions

- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Averages :

- ~ *Introduction*
- ~ *Definition & types*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Alligations and Mixtures :

- ~ *Introduction*
- ~ *Types*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Percentages :

- ~ *Basics & Concepts Preview*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Profit Loss :

- ~ *Introduction*
- ~ *Basics & Concepts*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> SI & CI :

- ~ *Introduction*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Ratio and Proportions :

- ~ *Introduction*
- ~ *Concepts & Definitions*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & Tips*
- ~ *Previous year questions*

=> Probability :

- ~ *Introduction & examples*
- ~ *Experiment*
- ~ *Sample space*
- ~ *Event & its probability*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Permutation & Combination :

- ~ *Introduction*
- ~ *Permutations*
- ~ *Combinations*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Heights and Distances :

- ~ *Basics of trigonometry*
- ~ *Trigonometric identities*
- ~ *T-ratios*
- ~ *Angel of elevation & depression*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Problems on Trains :

- ~ *Introduction*
- ~ *Various types of problems on trains*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Perimeter, Volume & Area :

- ~ *Introduction*
- ~ *Results on some polygons*
- ~ *Questions*
- ~ *Formulae*
- ~ *Tricks & tips*
- ~ *Previous year questions*

=> Partnership :

- ~ Introduction
- ~ Working & sleeping partners
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Quadratic Equations :

- ~ Introduction
- ~ Methods of finding roots
- ~ Nature of roots
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Coordinate Geometry :

- ~ Introduction
- ~ Cartesian system
- ~ Quadrants
- ~ Location of points
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Logarithms :

- ~ Definition
- ~ Types
- ~ Properties
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Set Theory :

- ~ Definition
- ~ Types
- ~ Operations
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Geometry :

- ~ Introduction
- ~ Plane geometry
- ~ Solid geometry
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Work & Wages :

- ~ Introduction
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Square & Cube Root :

- ~ Square root
- ~ Cube root
- ~ Questions
- ~ Race
- ~ Introduction
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Stocks & Shares :

- ~ Introduction
- ~ Some facts
- ~ Questions
- ~ Formulae
- ~ Tricks & tips
- ~ Previous year questions

=> Chain Rule :

- ~ Direct proportion
- ~ Indirect Proportion

=> Algebra :

- ~ Introduction & theory Preview
- ~ Questions
- ~ Formulae & summary
- ~ Tricks & tips
- ~ Previous year questions

# MLFlow Production Server Setup

---

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS PROJECT

Course link : <https://ineuron.ai/course/MLFlow-Production-Server-Setup>

## Course Description :-

MLflow is an open-source platform for machine learning lifecycle management. In this course, we will learn how to setup MLflow server in production with Mysql as the backend store and an s3 bucket as artifact registry.

## Course Features :-

- => Do Everything In Industry Grade Lab
- => Learn As Per Your Timeline
- => Hands-On Industry Real-Time Projects.
- => Self Paced Learning
- => Dashboard Access
- => Course Materials
- => Assignments

## What you will learn :-

- => Real Time Projects
- => Mlflow production server setup
- => Understand MLOPS best practices
- => Understand Aws cloud ec2 and mysql
- => Setup Mlflow server in production
- => Understand authentication with nginx
- => Get hands on with server setup

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Ketan Gangal :

*~ I have worked in data science for more than two years, and I have a track record of successfully implementing data science pipelines in production with practical expertise using ML-Ops, deep learning & machine learning. I also love sequence processing because it is deeply inspired by humans as our feeling, thoughts, emotions, sensations, language are sequential in nature if we can enable machine to understand sequence of information and act accordingly we can make significant progress towards true artificial intelligence.*

## Curriculum details :-

=> Welcome to the Course :

- ~ Course Overview
- ~ Dashboard Introduction

=> Project :- MLflow Production Server Setup :

- ~ Introduction to mlops
- ~ What is experiment tracking
- ~ Introduction to Mlflow
- ~ End Notes
- ~ Problem Description
- ~ Understand the application scope
- ~ Tour to existing solution
- ~ End Notes
- ~ Cost involved
- ~ End Notes
- ~ Amazon Cloud Overview
- ~ Quick Overview to elastic cloud compute
- ~ Quick Overview to amazon RDS
- ~ Quick Overview to amazon S3
- ~ Ec2 Server Setup
- ~ Amazon RDS Setup
- ~ Artifact Store Setup
- ~ Nginx Authentication
- ~ End Notes
- ~ Integration into training script
- ~ Create your prediction end point

- ~ End Notes
- ~ Overview to Amazone scale setup ECS
- ~ Conclude the project
- ~ Assignments & External Resources



# Mastering DSA with C++

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH C++

Course link : <https://ineuron.ai/course/Mastering-DSA-with-C++>

## Course Description :-

For performance-critical applications that need speed and effective memory management, C++ is an important programming language. It's employed in a variety of fields such as software and game development, virtual reality, robotics, and scientific computing. In this course you will learn the fundamentals of C++ with various data structures and algorithms.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => C++ programming basics
- => C++ data types
- => C++ data structures
- => Input/Output in C++
- => Control Flow in C++
- => Loops
- => Functions
- => OOP in C++
- => Memory management
- => Macros

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

- => Introduction to DSA :
  - ~ Why we need Data structures and algorithms
  - ~ Time based approach
  - ~ Concept of Big O and graphs
  - ~ Data Structures and Algorithms HB
- => Problem Solving :
  - ~ Start with a challenge - reverse string
  - ~ Reverse a string - solution
  - ~ Interview approach to solve a problem
  - ~ Classic interview steps for DSA problems
- => Data Structure Introduction :
  - ~ Memory process - Stack and Heap
  - ~ Physical and logical data structures
  - ~ Abstract Data Types - ADT
- => Recursion in depth :
  - ~ Introduction to recursion
  - ~ Tracing the recursion tree
  - ~ Trace tree assignment
  - ~ Trace tree solution

- ~ Types of Recursion
- ~ Complex recursion tree
- ~ What is Factorial
- ~ Factorial program in CPP
- ~ Fibonacci series THEORY
- ~ Fibonacci series and its version cpp Code
- ~ What is Power Program
- ~ Power Program cpp code
- ~ What is a Combination Program
- ~ Combination Program cpp code
- ~ Classic Tower of Hanoi problem
- ~ Classic Tower of Hanoi cpp code

=> Linked List in depth :

- ~ Introduction to Linked List
- ~ Add value in linked list - cases
- ~ Push Append and insertat in LinkedList - cpp code
- ~ Deletion of linked list THEORY.
- ~ Deletion in linked list CPP code
- ~ Delete complete linked list cpp code
- ~ Count all nodes in linkedlist cpp code
- ~ Reversing a linked list THEORY
- ~ Reversing a linked list cpp code

=> Circular Linked List in Depth :

- ~ Circular linked list THEORY
- ~ Circular Linked List push cpp code
- ~ Traverse a circular linked list cpp code
- ~ Deletion in circular linked list cpp code
- ~ count nodes in circular linked list cpp code
- ~ convert linked list to circular linked list cpp code

=> Doubly Linked List in Depth :

- ~ Theory for doubly linked list
- ~ Doubly linked list push cpp code
- ~ Insert After in doubly linked list cpp code
- ~ add to last in doubly linked list cpp code
- ~ Traverse a doubly linked list cpp code
- ~ Deleting a node in doubly linked list cpp code

=> Stack and Queue :

- ~ Stack - Push and Pop operation THEORY
- ~ Stack operations with cpp code
- ~ Queue concept THEORY
- ~ Queue implementation in cpp code
- ~ Circular queue THEORY
- ~ Circular queue cpp code

=> Binary Search Tree :

- ~ What is Binary Search tree and creation THEORY update
- ~ Insertion and Deletion in BST THEORY
- ~ InOrder Traversal of BST THEORY
- ~ Pre Order traversal in BST THEORY
- ~ Post order traversal in BST THEORY
- ~ Creating a Binary Search tree cpp code
- ~ search a key in BST cpp code
- ~ Insertion in BST cpp code
- ~ deletion of key in BST cpp code
- ~ inorder preorder and postorder traversal in BST cpp code

=> Hashing :

- ~ What is Hashing THEORY
- ~ Hash chaining with linked list
- ~ Linear Hash Shifting
- ~ Square hash shifting

=> AVL Tree :

- ~ What is AVL tree and height
- ~ Finding balance factor
- ~ Left Left and Right Right Rotation in AVL Tree
- ~ LR and RL rotation with 1 trick
- ~ Creating a AVL tree - Important
- ~ Deletion in AVL Tree.

=> HEAP :

- ~ Heap - Max and min Heap
- ~ Insertion and deletion in HEAP

=> Sorting algorithms :

- ~ Categories of sorts
- ~ Selection sort - Theory
- ~ Selection sort - cpp Code
- ~ Bubble Sort - Theory
- ~ Bubble Sort - cpp Code
- ~ Insertion sort - Theory
- ~ Insertion sort - cpp Code
- ~ Quick Sort - Theory
- ~ Quick Sort - Theory part 2
- ~ QuickSort-cpp code
- ~ Counting Sort - Theory
- ~ Merge Sort Theory

- ~ Merge sort cpp code
- ~ Counting Sort - cpp Code

# Pro C Programming Language

---

Topic Name : APTITUDE

Sub-topic Name : APTITUDE

Course link : <https://ineuron.ai/course/Pro-C-Programming-Language>

## Course Description :-

This course is designed mostly for C test takers.

## Course Features :-

=> Quizzes

=> Course completion certificate

## What you will learn :-

=> C Theoretical Test

=> C Practical Test

## Requirements :-

=> System with minimum i3 processor or better

=> At least 4 GB of RAM

=> Working internet connection

=> Dedication to solve

## Curriculum details :-

=> C Coding Test :

~ C Test 1

~ C Test 2

~ C Test 3

~ C Test 4

~ C Test 5

~ C Test 6

# Machine Learning in R

---

Topic Name : DATA SCIENCE

Sub-topic Name : MACHINE LEARNING

Course link : <https://ineuron.ai/course/Machine-Learning-in-R>

## Course Description :-

The Machine Learning with R course has been specifically developed to aid in the development of a solid understanding of the fundamentals of machine learning. You'll learn how to prepare data for modeling, train your models, visualize and evaluate their performance, and fine-tune their parameters for improved results. Learn the abilities you'll need to work as a machine learning scientist.

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Rstudio
- => Histograms
- => Scatterplots
- => overlaying plots
- => Simple Linear Regression
- => Multiple Linear Regression
- => Logistic Regression
- => Support Vector Machine
- => Decision Tree Classification

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

- => MD Imran :
  - ~ Working as Data Scientist with experience in solving real world business problems across different domains.

## Curriculum details :-

- => Basics of Statistical Computing in R :

- ~ Installing R
- ~ Rstudio
- ~ Packages
- ~ Plot()
- ~ Bar Charts
- ~ Histograms
- ~ Scatterplots
- ~ overlaying plots
- ~ summary()
- ~ describe()
- ~ selecting cases
- ~ data formats part 1
- ~ data formats part 2
- ~ factors
- ~ entering data
- ~ importing data

- => Data Preprocessing in R :

- ~ Getting Started
- ~ Dataset Description
- ~ Importing the Dataset
- ~ Difference between mean and ave
- ~ Taking care of Missing Data part 1

- ~ *Taking care of Missing Data part 2*
- ~ *Encoding Categorical Data*
- ~ *Splitting the dataset into the training set and test set*
- ~ *Feature scaling*

=> Simple Linear Regression :

- ~ *simple Linear Regression Intuition*
- ~ *Simple Linear Regression in R Step 1*
- ~ *Simple Linear Regression in R Step 2*
- ~ *Simple Linear Regression in R Step 3*
- ~ *Simple Linear Regression in R Step 4*

=> Multiple Linear Regression :

- ~ *Multiple Linear Regression in R Step 1*
- ~ *Multiple Linear Regression in R Step 2*
- ~ *Multiple Linear Regression in R Step 3*

=> Logistic Regression :

- ~ *Logistic Regression in R - Step 1*
- ~ *Logistic Regression in R - Step 2*
- ~ *Logistic Regression in R - Step 3*
- ~ *Logistic Regression in R - Step 4*
- ~ *Logistic Regression in R - Step 5*
- ~ *Logistic Regression in R - Step 6*

=> Support Vector Machine :

- ~ *Support Vector Machine Demo*

=> Decision Tree Classification :

- ~ *Decision Tree Classification Demo*

# Vuejs

---

Topic Name : WEB DEVELOPEMENT

Sub-topic Name : VUE JS

Course link : <https://ineuron.ai/course/Vuejs>

## Course Description :-

VueJS is the shooting star in the world of JavaScript frameworks, regardless of whatever measure you choose (Google Trends, Tweets, etc.). This course covers the most recent version of Vue in great depth and from the ground up. In this course, we will go over all of the fundamental of VueJs. Vue JS and other frontend frameworks are incredibly popular because they provide the same dynamic, fantastic user experience that we have come to expect from mobile applications - but now in the browsers as well. And it is no surprise that positions requiring frontend framework expertise such as VueJS are among the highest-paying in the business!

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => VueJs project structure
- => VueJs data types and methods
- => Passing data to props
- => Adding editable forms in todo
- => Passing methods in vueJs

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Hitesh Choudhary :

*~ I like to make videos related to code and tech in my free time. I also lead a few tech teams in startups, help in hiring talent for companies. I am also on a part time traveller, with 31 countries checked off so far!*

## Curriculum details :-

=> Getting Started With VueJS :

- ~ Introduction to Vue JS
- ~ Important note on Vue docs
- ~ Vue web page via CDN
- ~ Injecting Vue on web page
- ~ Another method to add app

=> Basics of VueJS :

- ~ A nice card in Vue
- ~ Directives in VueJS
- ~ Handling Arrays in VueJS
- ~ loops and assignment in VueJS
- ~ Handling Booleans and conditionals in VueJS
- ~ Login and logout in VueJS
- ~ Why people avoid v-show

=> 2 way binding in VueJS :

- ~ Getting the values from html in VUEJS

=> 3 way binding in VueJS :

- ~ Model the data in VueJS

=> 4 way binding in VueJS :

- ~ Computed and methods in VueJS

=> 5 way binding in VueJS :

- ~ Handling computed in VueJS

=> 6 way binding in VueJS :

- ~ Assignment time in VueJS

=> 7 way binding in VueJS :

- ~ Life Cycle hooks in VueJS

=> Moving to Vue cli :

- ~ Vue cli and GUI

- ~ Redo the project in VueJS

- ~ Setup you HTML for counter app

- ~ Counter app and assignment

=> Conditionals in VueJS :

- ~ bulding logics for Rating app

- ~ Finishing up rating app in VueJS

- ~ Word generator project in VueJS

- ~ Word generator methods

- ~ A nasty bug to find in VueJS

=> Components and third part libraries :

- ~ Adding third party libraries

- ~ Your first component

- ~ Watcher in VueJS

- ~ craft a winning login in tictacToe VueJS

- ~ Making our game functional in VueJS

- ~ Reload the game in Vue JS

=> Handling local storage in VueJS :

- ~ Building a local storage app in VueJS

- ~ Bring in Moment and UUID

- ~ A reuseable header in Vue JS

- ~ Input form component in VueJS

- ~ Movie card component in VueJS

- ~ Handling local storage in VueJS

- ~ Bring all components together and bug assignment VueJS

- ~ LifeCycle events in action VueJS

=> Handling API in VueJS :

- ~ Introducing the API in VueJS

- ~ Setting up API project in VueJS

- ~ Axios to fire request on web VueJS

- ~ Handling response with check Vuejs

- ~ Testing the response VueJS

- ~ Summing up user card Vue JS

=> Routing and state management :

- ~ A new router app in vuejs

- ~ Basics of routing middleware

- ~ router link in vue js

- ~ All about routing in Vuejs

- ~ Getting started with Github app in vuejs

- ~ Firebase config settings in vue

- ~ Creating lots of files for vue git project

- ~ Store in vuejs

- ~ Signup gitapp in vuejs

- ~ map getters in vuex

- ~ map actions in vuex

- ~ handling user card in vuex

- ~ preparing repo table in vuex

- ~ handling home component with store in vuex

- ~ Auth Guard in vue router

- ~ debugging session



# Complete DSA in Python

---

Topic Name : DATA STRUCTURE

Sub-topic Name : DSA WITH PYTHON

Course link : <https://ineuron.ai/course/Complete-DSA-in-Python>

## Course Description :-

A comprehensive chase to excel any interview for the Data Structures and Algorithms. This course has been specifically designed to provide resources that would assist you in cracking problem-solving interviews. The presented problems in the course would suffice to look on to positive outcomes in the interviews.

## Course Features :-

- => Course Materials
- => Self Paced Learning
- => Lifetime Dashboard Access
- => Completion Certificate

## What you will learn :-

- => Introduction to Algorithms
- => Analysis in Algorithms
- => Array Data Structure
- => Heap Data Structure
- => Recursion
- => Divide and Conquer
- => Linked List Data Structure
- => Stack and Queue
- => Hashing Data Structure
- => Tree Data Structure
- => Binary Search Tree
- => Graph Traversal Algorithms
- => Application of greedy algorithm
- => Dynamic Programming
- => Research Area- P, NP, NP-Hard and NP-Complete Problems

## Requirements :-

- => System with minimum i3 processor or better
- => At least 4 GB of RAM
- => Working internet connection
- => Dedication to learn

## Instructors :-

=> Priya Bhatia :

~ Expertise in data structure competitive programming and solving analytical problems and implementing data structure algorithm in multiple programming language. I have done my M.Tech in Artificial Intelligence at IIT Hyderabad and have an experience of implementation in multiple projects.

## Curriculum details :-

- => Introduction to Algorithms :
  - ~ Complete DSA Roadmap
  - ~ Why DSA required
  - ~ Algorithms Introduction
  - ~ Steps to construct an algo
- => Analysis in Algorithms :
  - ~ Types of Analysis
  - ~ Asymptotic Notation - Big O Time Complexity
  - ~ Asymptotic Notation - Omega Time Complexity
  - ~ Asymptotic Notation - Theta Time Complexity
  - ~ Apriori Analysis - Time Complexity Analysis Part1
  - ~ Apriori Analysis - Time Complexity Analysis Part2
  - ~ Apriori Analysis - Time Complexity Analysis Part3
  - ~ Practice Set - Asymptotic Notations
  - ~ Complexity Classes
  - ~ Recurrence Relation Introduction
  - ~ Substitution Method - Problem 1

- ~ Substitution Method - Problem 2
- ~ Substitution Method - Problem 3
- ~ Recursive Tree Approach - Problem 1
- ~ Recursive Tree Approach - Problem 2
- ~ Recursive Tree Approach - Problem 3
- ~ Practice Set - Substitution and Recursive Tree Approach
- ~ Masters Theorem Case 1
- ~ Masters Theorem Case 2
- ~ Masters Theorem Case 3
- ~ Practice Set - Masters Theorem

=> Array Data Structure :

- ~ Introduction to Array Data Structure
- ~ Array Data Structure Implementation
- ~ Address of an element in 1D array
- ~ Address of an element in 2D array
- ~ Searching of an element - Linear Search
- ~ Searching of an element - Binary Search
- ~ Recurrence Relation of Binary Search
- ~ Implementation of Binary Search
- ~ Binary Search Interview Problem
- ~ Search a 2D Matrix
- ~ Searching of an element - Ternary Search
- ~ Recurrence Relation of Ternary Search
- ~ Implementation of Ternary Search
- ~ Sorting in an array - Comparison and Non-Comparison
- ~ Stable and Unstable sorting algorithms
- ~ Inplace and Outplace Sorting algorithms
- ~ Comparison Sort - Bubble Sort
- ~ Comparison Sort - Bubble Sort Implementation
- ~ Comparison Sort - Selection Sort
- ~ Comparison Sort - Selection Sort Implementation
- ~ Comparison Sort - Insertion Sort
- ~ Comparison Sort - Insertion Sort Implementation
- ~ FAANG Interview Question on Arrays - Best Time to Buy and Sell Stock
- ~ FAANG Interview Question on Arrays - Collinear Points
- ~ FAANG Interview Question on Arrays - Majority Element
- ~ FAANG Interview Question on Arrays - Sort Colors

=> Heap Data Structure :

- ~ Basics of Heap Sort - Full Binary Tree vs Complete Binary Tree vs Almost Complete Binary Tree
- ~ Concept of Minheap and Maxheap Tree
- ~ Insertion in Minheap or Maxheap Tree
- ~ Deletion in Minheap or Maxheap Tree
- ~ Creation of Minheap or Maxheap Tree
- ~ Time Complexity Derivation to build minheap or maxheap
- ~ Comparison Sort - Heap Sort
- ~ FAANG Interview Question on Heap - Top K frequent elements
- ~ FAANG Interview Question on Heap - K Closest Points to Origin

=> Recursion :

- ~ Introduction to Recursion
- ~ Factorial Finding using Recursion with its Implementation
- ~ Fibonacci Series using Recursion with its Implementation
- ~ Count Of number of ways to reach upstairs

=> Divide and Conquer :

- ~ Introduction to Divide and Conquer
- ~ Applications of Divide and Conquer - Finding of maxima and minima
- ~ Applications of Divide and Conquer - Implementation of finding of maxima and minima
- ~ Applications of Divide and Conquer - Finding of power of an element with its Implementation
- ~ Applications of Divide and Conquer - Binary Search
- ~ Applications of Divide and Conquer - Recurrence relation of Binary Search
- ~ Applications of Divide and Conquer - Implementation of Binary Search
- ~ FAANG Interview Question- Two Pointers Problem
- ~ Applications of Divide and Conquer - Merge Sort
- ~ Applications of Divide and Conquer - Implementation of Merge Sort
- ~ FAANG Interview Question on MergeSort - Finding of single sorted array complexity
- ~ Applications of Divide and Conquer - Quick Sort
- ~ Applications of Divide and Conquer - Implementation of Quick Sort
- ~ FAANG Interview Scenario Based Question on QuickSort complexity
- ~ Applications of Divide and Conquer - Randomized QuickSort
- ~ Applications of Divide and Conquer - Selection Procedure
- ~ Applications of Divide and Conquer - Implementation of Selection Procedure
- ~ Applications of Divide and Conquer - Count Of number of an inversions
- ~ Applications of Divide and Conquer - Strassen's Matrix Multiplication

=> Linked List Data Structure :

- ~ Introduction to Linked List
- ~ Insertion of a node in Linked List - Front
- ~ Insertion of a node in Linked List - After a given node
- ~ Insertion of a node in Linked List - End
- ~ Deletion of a node in Linked List
- ~ Searching of a node in Linked List
- ~ FAANG Interview Question - Reversal of a node in Linked List
- ~ FAANG Interview Question - Count of all nodes in Linked List
- ~ FAANG Interview Question - Floyd's Cycle Detection Algorithm
- ~ FAANG Interview Question - Merge Of two Sorted Linked List

=> Skip List Data Structure :

~ Skip List- Motivation, Build-in, Search, Insertion and Deletion skip list

=> Stack and Queue :

~ Introduction to Stack Data Structure and Push Operation in depth  
~ Stack- Pop operation  
~ Implementation of Stack using array and linked list  
~ Queue- Insertion and Deletion operation  
~ Implementation of Queue using array and linked list  
~ FAANG Interview Question - Valid Parenthesis

=> Hashing Data Structure :

~ Introduction to Hashing Data Structure  
~ Hash Function and its types  
~ Implementation of Hash Functions  
~ Open addressing - Linear Probing and Primary Clustering  
~ Open addressing - Quadratic Probing and Secondary Clustering  
~ Open addressing - Double Hashing  
~ Chaining  
~ Load Factor and Rehashing

=> Tree Data Structure :

~ Basics of Tree - Full Binary Tree vs Complete Binary Tree vs Almost Complete Binary Tree

=> Tree Traversal Algorithms :

~ Tree Traversal Algorithms- Inorder, Preorder and PostOrder  
~ FAANG Interview Questions on Tree Traversal Algorithm

=> Binary Search Tree :

~ Introduction to Binary Search Tree  
~ Insertion and Inorder Traversal in BST  
~ FAANG Interview Question- Minimum value in BST  
~ FAANG Interview Question- Find unique possible BST's  
~ Searching in Binary Search Tree  
~ Deletion in Binary Search Tree

=> Graph Traversal Algorithms :

~ Basics Of Graph- Simple vs Multigraph, Null vs Complete Graph, Relationship between edges and vertices in Simple Graph  
~ Introduction to Graph Traversal Algorithms  
~ Introduction to Depth First Search  
~ DFS Psuedocode and illustration using an example  
~ DFS Coding Implementation  
~ BFS Intro, Psuedocode and illustration using an example  
~ BFS Coding Implementation

=> Greedy Algorithm :

~ Introduction to greedy algorithm

=> Application of greedy algorithm :

~ Fractional Knapsack Problem  
~ Implementation of Fractional Knapsack Problem  
~ Basics Of Graph- Simple vs Multigraph, Null vs Complete Graph, Relationship between edges and vertices in Simple Graph  
~ Introduction to Spanning Tree and Minimum Spanning Tree  
~ Minimum Spanning Tree- Kruskal 's Algorithm  
~ Minimum Spanning Tree- Prim's Algorithm  
~ Single Source Shortest Path- Dijkstra's algorithm  
~ Single Source Shortest Path- Dijkstra's algorithm Implementation  
~ Huffman Coding  
~ Optimal Merge Pattern  
~ Job Sequencing with Deadline

=> Dynamic Programming :

~ Introduction to Dynamic Programming

=> Application of Dynamic Programming :

~ Fibonacci Series using Dynamic Programming  
~ 0-1 Knapsack Problem

=> Research Area- P, NP, NP-Hard and NP-Complete Problems :

~ Research Area- P, NP, NP-Hard and NP-Complete Problems

=> Some ending tips for all students :

~ Some ending tips for all students

=> Detailed Interview Process to crack FAANG Companies(SDE Roles) :

~ Detailed Interview Process to crack FAANG Companies

# Manual Testing Foundations

---

Topic Name : TESTING

Sub-topic Name : MANUAL TESTING

Course link : <https://ineuron.ai/course/Manual-Testing-Foundations>

## Course Description :-

This course will help you get started with Software Testing. We will discuss different terms and terminologies to develop a QA mindset. We will also discuss about roles and responsibilities of a software tester and what are the day-to-day activities that you have to perform as a tester. You will also learn STLC (Software Testing Life Cycle) and its different phases and at the end, we will use a very useful in-demand tool called "JIRA".

## Course Features :-

- => Course material
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Objectives of Testing
- => Scope of Testing
- => Prerequisites of becoming a Tester
- => Potential growth in the Software Testing Career
- => Roadmap to a testing career
- => Phases of Testing
- => Unit Testing
- => Integration Testing
- => System Testing
- => UAT Testing -Alpha & Beta Testing
- => Deployment Process

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Kiran Sahu :

~ QA Manager with 12+ years of professional experience, worked in Brands like Infosys, Delhivery, Mydala, Aurea, Jive, Crossover, Agama Solutions & OSTC, have experience of working in global platforms and with multinational professionals. Strong domain knowledge on Retail, Logistics, Banking, Trading, Ecommerce Applications. Experience in Training and Mentoring Candidates all across the globe on Software Testing, MySQL and Agile.

## Curriculum details :-

=> Getting Started with Software Testing and STLC :

- ~ Basics of Software Testing
- ~ Need of Software Testing
- ~ Objectives of Testing
- ~ Scope of Testing
- ~ Prerequisites of becoming a Tester
- ~ Potential growth in the Software Testing Career
- ~ Roadmap to a testing career
- ~ SDLC
- ~ STLC
- ~ SDLC Vs STLC

=> Phases of Testing and Types of Testing :

- ~ Phases of Testing
- ~ Unit Testing
- ~ Integration Testing
- ~ System Testing
- ~ UAT Testing -Alpha & Beta Testing

- ~ *Deployment Process*
- ~ *Ecommerce Project Example*
- ~ *Types of Testing*

=> JIRA- A Complete Overview :

- ~ *Introduction to JIRA*
- ~ *Introduction to Agile*
- ~ *What is Scrum*
- ~ *What is Sprint Cycles*
- ~ *Importance of Jira in Agile*
- ~ *Project Management using Jira in Agile*
- ~ *Bug Tracking using Jira in Agile*
- ~ *Sprint Report- Burndown Chart in Jira*

# Video editing with Adobe Premiere Pro

---

Topic Name : K12

Sub-topic Name : CLASS10

Course link : <https://ineuron.ai/course/Video-editing-with-Adobe-Premiere-Pro>

## Course Description :-

In this course, you will learn the basics of video editing fundamentals like adding background music, cut & trim, adding text, and many others. You will also learn to color correct and grade your videos to create professional-level content. Students after successful completion will gain hands-on practical experience in making top-notch videos. You can start applying for freelance jobs to earn a fortune out of it.

## Course Features :-

- => Online Instructor-led learning
- => Practical Implementation
- => Integrate academic knowledge with the tech
- => Real-time Project
- => Live Class Recording
- => Doubt Clearing
- => Assignment in all the Module
- => Quiz in every Module
- => Career Counselling
- => Completion Certificate

## What you will learn :-

- => Installation of Adobe Premiere Pro
- => Exploring the Premiere pro workspace and customizing it
- => Exploring the Premiere pro workspace and customizing it
- => Introduction to The Editing Tools
- => Video Properties
- => Adding Style in Your Videos
- => Adding Video and Audio Transitions
- => Audio Editing in Premiere Pro

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Curriculum details :-

- => Introduction to the course :
  - ~ Course introduction
  - ~ Who is this course for ?
  - ~ Course overview & Course outcome
  - ~ Course pre-requisite
  - ~ What is Video editing?
  - ~ What are the different tools used for Video editing?
  - ~ Why Adobe premiere pro ?
- => Assignment :
  - ~ What makes you think you need to edit your videos?
- => Installation :
  - ~ Installation of Adobe premiere pro
  - ~ Basic overview of the platform
- => Assignment :
  - ~ What are the alternatives to Adobe Premiere pro?
- => Premiere Pro Basics :
  - ~ How to start a Premiere pro project?
  - ~ Exploring the Premiere pro workspace and customize it.
  - ~ How to import and organize media in Premiere pro?
- => Assignment 1 :
  - ~ Create your own workspace in premier pro

=> Video Editing Basics :

- ~ How to start a new sequence?
- ~ Let us understand sequences
- ~ What is Timeline?
- ~ How to add clips to the timeline?
- ~ The Editing Tools
- ~ What are Razor cuts and how to use it?
- ~ What is Ripple edits and how to use it?
- ~ What are Slips and how to use it?
- ~ How to synchronize audio and video?
- ~ Video Properties
- ~ What is Scale?
- ~ What s Position?
- ~ What is Opacity?
- ~ Types of Cuts
- ~ What is Straight?
- ~ What is J-cut?
- ~ What is L-cut?
- ~ Basic colour grading

=> Assignment 2 :

- ~ Create HD and 4K sequences

=> Assignment 3 :

- ~ Create a timeline using atleast 10 different videos syncing with audio

=> Assignment 4 :

- ~ What is the difference between Scale, Rotation, Opacity?

=> Assignment 5 :

- ~ Make a sequence using L-cut, J-cut and Jump cut

=> Adding Style to Your Videos :

- ~ What is Video style section?
- ~ Create a zoom in with Keyframes
- ~ Create zoom out with Keyframes
- ~ Using Nests to create a cool zoom sequence
- ~ How to use Blend modes to combine videos?
- ~ How to create a split Create Effect with Borders?
- ~ How to apply The Ken Burns effect- zooming in and out of photos ?

=> Assignment 6 :

- ~ Create a cinematic zoom effect using keyframes

=> Adding Video and Audio Transitions :

- ~ How do we add video and audio transitions in Premiere pro?
- ~ How do we customize our video transition properties ?
- ~ How do we add audio transitions and create custom audio fades?

=> Assignment 7 :

- ~ Use different types of audio and video transitions from the effects panel

=> Audio Editing in Premiere Pro :

- ~ How do we make our Audio louder or quieter ?
- ~ How to remove background noise from audio in Premiere pro?

=> Assignment 8 :

- ~ Use the Effects panel to experiment with various audio effects.

=> Course Summary :

- ~ Course Outro
- ~ Future learning path

# System Design with Design Patterns

---

Topic Name : SYSTEM DESIGN

Sub-topic Name : SYSTEM DESIGN MASTERS

Course link : <https://ineuron.ai/course/System-Design-with-Design-Patterns>

## Course Description :-

The software engineering interview process includes system design questions as a routine element of the process. The way you perform in these interviews reflects on your ability to work with complicated systems, which is reflected in the position and salary offered by the interviewing organisation. The purpose of this course is to help you master software engineering interviews.

## Course Features :-

- => 6 Months Programme
- => Course resources
- => On demand recorded videos
- => Practical exercises
- => Quizzes
- => Assignments
- => Course completion certificate

## What you will learn :-

- => Design principles
- => Introduction and types
- => Creational Patterns
- => Structural Patterns
- => Behavioural Patterns
- => Important System Design Concepts
- => System Design Problems
- => Designing Facebook Messenger
- => Designing Twitter
- => Designing Youtube
- => Designing Netflix

## Requirements :-

- => System with Internet Connection
- => Interest to learn
- => Dedication

## Instructors :-

=> Anjali Sheel :

~ Currently working for Microsoft as SDE2 in the windows team with more than 5 years of experience in software development. Mentoring students in the Microsoft's engage program. Have more than one year of teaching experience for competitive programming. Have given various webinars for learner community for interview preparation. Have done M.Tech from Delhi Technological university(formerly known as DCE). Have 4 years of experience working at Siemens healthineers, a german product based company.

## Curriculum details :-

- => Design principles :
  - ~ DRY principles
  - ~ KISS principles
  - ~ SOLID principles
  - ~ CUPID principles
- => Introduction and types :
  - ~ OOPS overview
  - ~ The Singleton Pattern part 1
  - ~ The Singleton Pattern part 2
  - ~ The Singleton Pattern part 3
- => Creational Patterns :
  - ~ The Factory Pattern
  - ~ The Factory Method Pattern
  - ~ The Abstract Factory Pattern
  - ~ The Singleton Pattern
  - ~ The Builder Pattern
  - ~ The Prototype Pattern



~ *Summary of Creational Patterns*

=> **Structural Patterns :**

- ~ *The Adapter Pattern*
- ~ *The Bridge Pattern*
- ~ *The Composite Pattern*
- ~ *The Decorator Pattern*
- ~ *The Faade Pattern*
- ~ *The Flyweight Pattern*
- ~ *The Proxy Pattern*
- ~ *Summary of Structural Patterns*

=> **Behavioural Patterns :**

- ~ *Chain of Responsibility Pattern*
- ~ *The Command Pattern*
- ~ *The Interpreter Pattern*
- ~ *The Iterator Pattern*
- ~ *The Mediator Pattern*
- ~ *The Memento Pattern*
- ~ *The Observer Pattern*
- ~ *The State Pattern*
- ~ *The Strategy Pattern*
- ~ *The Template Pattern*
- ~ *The Visitor Pattern*
- ~ *Null Object pattern*

=> **Important System Design Concepts :**

- ~ *System Design Basics*
- ~ *Key Characteristics of Distributed Systems*
- ~ *Load Balancing*
- ~ *ClientServer Model*
- ~ *Network Protocols*
- ~ *Storage*
- ~ *Latency And Throughput*
- ~ *Availability*
- ~ *Caching*
- ~ *Data Partitioning*
- ~ *Indexes*
- ~ *Replication*
- ~ *Sharding*
- ~ *Proxies*
- ~ *Redundancy*
- ~ *SQL vs. NoSQL*
- ~ *CAP Theorem and*
- ~ *PACELC Theorem*
- ~ *Consistent Hashing*
- ~ *Long Polling vs WebSockets vs Server Sent Events*
- ~ *Bloom Filters*
- ~ *Quorum, Leader and Follower, Heartbeat, Checksum*
- ~ *Rate Limiting*
- ~ *Logging And Monitoring*
- ~ *Security And HTTPS*
- ~ *API Design*

=> **System Design Problems :**

- ~ *System Design Interviews: A step by step guide*
- ~ *Designing a URL Shortening service like TinyURL*
- ~ *Designing Pastebin*
- ~ *Designing Instagram*
- ~ *Designing Dropbox*
- ~ *Designing Facebook Messenger*
- ~ *Designing Twitter*
- ~ *Designing Youtube*
- ~ *Designing Netflix*
- ~ *Designing Typeahead Suggestion*
- ~ *Designing an API Rate Limiter*
- ~ *Designing Twitter Search*
- ~ *Designing a Web Crawler*
- ~ *Designing Facebooks Newsfeed*
- ~ *Designing Yelp or Nearby Friends*
- ~ *Designing Uber backend*
- ~ *Designing Ticketmaster*