Robotics

Topic Name : K12

Sub-topic Name : CLASS10

Course link : https://ineuron.ai/course/Robotics

Course Description :-

The Introduction to Robotics Specialization covers the fundamentals of robot flight and movement, as well as how robots detect their surroundings and alter their movements to avoid obstacles, negotiate tough terrains, and complete complicated jobs like construction and disaster response. You'll learn about real-world examples of how robots have been used in disasters, how they've improved human health care, and what their future capabilities will be. The courses lead up to a capstone in which you'll learn how to programme a robot to execute various tasks like flying and gripping objects.

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 :-

=> Introduction

=> Essential Tools Basics

=> Robot Car

=> Introduction to Bluetooth

=> Project: Bluetooth Robot Car

Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

Curriculum details :-

=> Introduction :

~ What is Robotics?
~ Components Required

=> Essential Tools Basics :

~ What is a breadboard?
~ Using a Breadboard
~ Using a Multimeter
~ Using jumper wires
~ Soldering Basics
~ Servo Motor
~ Dc motor
~ Stepper Motor

=> Robot Car :

~ Components of Robot car
~ Assembly of the car
~ Coding
~ Testing

=> Introduction to Bluetooth :

~ Components Using Bluetooth
~ Bluetooth module basics
~ Bluetooth Coding & Output

=> Project: Bluetooth Robot Car :

~ Components of Bluetooth Car
~ Assembly
~ Coding

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*

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 AIops. 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 Explaination(LIMEI)
~ The END

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

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

Ethical Hacking & Penetration Testing

Topic Name : CYBER SECURITY

Sub-topic Name : CYBERSECURITY MASTERS

Course link : https://ineuron.ai/course/Ethical-Hacking-&-Penetration-Testing

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 the world of Security Audits and Ethical Hacking

=> Setting up your own pentesting lab

=> Getting friendly with Linux and security related commands

=> Maintaining anonymity

=> Information Gathering with Kali & Web apps

=> Introduction to web pentesting and dvwa testing environment

=> Command Injection (Low-Medium-High), Mitigation and Incident Case Study

=> Cookie manipulating and insecure session management

=> Burpsuite complete training with fundamentals

=> File upload vulnerability

=> SQL Injection

=> Boolean Injection & automation with SQL Injection

Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

Curriculum details :-

=> Introduction to the world of Security Audits and Ethical Hacking :

- *~ Ethical Hacking Origin*
- *~ types of hackers*
- *~ Introduction to the world of Security Audits and Ethical Hacking*
- *~ Disclaimer and Requirements*

=> Setting up your own pentesting lab :

- *~ Grabbing tools*
- *~ Grabbing tools(CORRECTION REQUIRED)*
- *~ Installation of Virtual Box*
- *~ Installing Kali Linux*
- *~ Tour of VB & kali*
- *~ Installing Metsploitable*
- *~ Installing Windows & XP in VB*

=> Getting friendly with Linux and security related commands :

- *~ Important Network Settings & Command lines*
- *~ Basic Terminal Commands*
- *~ Managing File Permissions in Linux*
- *~ More terminal security related commands*

=> Maintaining anonymity :

- *~ Installing tor, onion routing and dark net*
- *~ Do's and Don't of Tor browser*
- *~ Proxychaining and DNSleaktest*
- *~ Changing your Mac Address*

=> Information Gathering with Kali & Web apps :

~ dnsenum (information Gathering)-
~ Zone Transfer Vulnerability-
~ Information Gathering With Dig Tool-
~ Dns Tracing And Wireshark Overview-
~ Information Gathering With Websites-
~ finding Other Domains On Same Server And Comprehensive Dns Report-
~ finding Subdomains And Facebook Subdomain Brute Force Report Studying-

=> Introduction to web pentesting and dvwa testing environment :

~ intro To Web Pentesting(n)-
~ configuring Dvwa In Kali(n)-
~ dvwa Error Solving(n)-

=> Command Injection (Low-Medium-High), Mitigation and Incident Case Study :

~ What is command injection and CI low level breach
~ Breaching medium level of security
~ Command injection - High Level
~ Command injection Mitigation and secure code logic
~ Remote Code Execution incident Report

=> Cookie manipulating and insecure session management :

~ Insecure Session management & cookie manipulating flaw

=> What is Cross Site Request Forgery? CSRF Attacks & Mitigation :

~ CSRF (N)
~ CSRF()N
~ CSRF Part (N)

=> Burpsuite complete training with fundamentals :

~ HTTPS Fundamental & request response method
~ Introduction anad configuring burpsuite
~ Importing HTTPS certificate in burpsuite
~ Understanding all the modules of burpsuite

=> File upload vulnerability :

~ File Upload Vulnerability (Low Security)
~ File Upload Vulnerability (Medium)

=> File inclusion Vulnerability :

~ LFI & RFI - Low Level
~ NewLFI-RFI (Medium & High level)

=> SQL Injection :

~ Sql-Lab-setup & What is database?
~ Sql Fundamentals
~ What is id and how to join the query in SQLi
~ Selecting vulnerable columns and fetching database name
~ dumping the database

=> Boolean Injection & automation with SQL Injection :

~ Boolean based queries and fundamentals
~ Boolean based live example
~ Automation with SQL Map

=> Cross site scripting :

~ XSS Reflected
~ Stored XSS & yahoo cookie stealing tale
~ DOM Based XSS & Google Lab

=> Brute force attacks, OT Bypass & Payment Gateway Bypass :

~ Brute forcing Password
~ One time password Brute force & Payment Gateway Bypass

=> Pentesting with automated tools, report & POC making :

~ Pentesting reports & POC making with Parot
~ Automation with owasp Zap
~ Httrack, WPScan & Accunetix overviews with final words

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 Iaas 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

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

=> Promethus

=> 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 aginf
~ 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)
~ Hostanme 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 sofwares
~ 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 Lusture
~ 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 Convertion Tool

=> Security Compliance :

~ Security and Complaince - 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 commnets*
~ *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
~ databse 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
~ Containarize 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 unterstanding

~ *Iaac 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*

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 Shifing*
~ *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*

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 2) 3*3 3)4*4 4) 2*3 5)3*46)2*4 7)3*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.

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 Dialoglow- Building The Solution

=> Chatbot With Google Dialoglow- Implementing The FulFillment

=> Chatbot With Google Dialoglow- Integration With Telegram

=> Chatbot With Google Dialoglow- Summary And Further Work

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 discusson 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 is 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?*

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

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

IOT

Topic Name : IOT

Sub-topic Name : IOT MASTERS

Course link : https://ineuron.ai/course/IOT

Course Description :-

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

Course Features :-

=> Course material

=> Course resources

=> On demand recorded videos

=> Practical exercises

=> Quizzes

=> Assignments

=> Course completion certificate

What you will learn :-

=> IoT Architecture & Communication Protocols

=> MQTT Introduction

=> MQTT Installation

=> MQTT Pub Sub Test

=> MQTT ACL

=> MQTT ACL username creation

=> MQTT ACL python program

=> RPi4 Sensor Data through MQTT

=> MongoDB Introduction

=> Introduction to Azure IoT Hub

=> Introduction to Azure Storage Account

Requirements :-

=> System with Internet Connection

=> Interest to learn

=> Dedication

Curriculum details :-

=> Course Introduction :

~ Course Introduction

=> IoT Architecture :

~ IoT Architecture & Communication Protocols

=> Introduction to MQTT :

~ MQTT Introduction
~ MQTT Installation
~ MQTT Pub Sub Test
~ MQTT Debugging methods
~ MQTT QoS
~ MQTT sending json file
~ MQTT listener configuration
~ MQTT Security Methods
~ MQTT username password
~ MQTT Client Tools

=> MQTT with Python :

~ MQTT Python Software Development
~ Jupyter notebook hello world
~ Python mqtt libarries
~ Paho mqtt installation
~ Paho mqtt sample test
~ Python simulation data
~ Simuation data program
~ Publish simulation data
~ Subscribe simulation data
~ Username password in python program
~ MQTT ACL
~ MQTT ACL username creation

~ MQTT ACL python program

=> RPi4 Sensor Data through MQTT :

~ Raspberry Pi 4 Introuction
~ Rpi python program for mqtt publish
~ Rpi sensor interfacing
~ Replacement of simulation with actual sensor data

=> MongoDB Introduction :

~ NoSQL databases
~ NoSQL MongoDB Properties
~ MongoDB Installation
~ MongoDB CLI
~ MongoDB Python Library pymongo

=> IoT with Cloud Computing :

~ Cloud Computing and IoT Services

=> Introduction to Azure IoT Hub :

~ Azure IoT Hub Introduction
~ Azure IoT Hub In Depth
~ Your First Azure IoT Device
~ Your First Azure IoT Device Continued
~ D2C Azure Developer Options
~ D2C Python Code
~ Tools for monitoring messages
~ C2D Python code
~ Azure IoT Hub Python SDK

=> Introduction to Azure Storage Account :

~ Azure Storage Account Service Introduction
~ Create blob storage account
~ Saving d2c data in container blob

=> Introduction to X509 Certificates :

~ Introduction to X509 certificates
~ Openssl for working with X509
~ Generating self signed certificate using openssl
~ D2C using Self Signed certificates

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

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

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 devlopment?
~ 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*
~ *Challenege*
~ *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 :

~ Keccack256
~ 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 overwiew :

~ 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 Marketpalce 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*
~ *Extrinsics*

~ *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*

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*

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

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 Desing, 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 deisgn

=> Deployment Architecture

=> Mutistage 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 withpractical expertise using ML-Ops, deep learning& machine learning. I alsoLove 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 understandsequence 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 deisgn
~ Deployment Architecture
~ Mutistage 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 deisgn
~ Deployment Architecture
~ Mutistage 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 deisgn
~ Deployment Architecture
~ Mutistage 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 Indentification Pipeline
~ Face Registration Pipeline
~ Capturing Images of a Person
~ Generating Embedding of Facial Image
~ Save Embeddding in Database
~ Face Identification Pipeline
~ Detecting face of a Person at login portal
~ Generate embedding of captured face
~ Search Generated Embedding in DB using simillarity 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 deisgn
~ Deployment Architecture
~ Mutistage 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

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

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 wtih 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 Methodlogies 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*
~ *BI Project Excecution*
~ *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 Colums*
~ *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

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 comparision 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

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

AIOPS Course

Topic Name : DATA SCIENCE

Sub-topic Name : MLOPS

Course link : https://ineuron.ai/course/AIOPS-Course

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, etc.

Course Features :-

=> Course material

=> Course resources

=> On demand recorded videos

=> Practical exercises

=> Quizzes

=> Assignments

=> Course completion certificate

What you will learn :-

=> AI/MLOps

=> Mlflow

=> Linux

=> GIT

=> DVC

=> Docker

=> Kubernetes

=> TFX

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.

=> 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 :-

=> AIOPS Course :

~ 17th July induction session
~ 18th July Live Class Introduction to AI/MLOps
~ 20th July Live doubt clearing Session
~ 24th July Live Class MLOps level1 workflow
~ 25th July Live Class MLOps level2 workflow
~ 31st July Live Class Linux Setup in Virtual Box
~ 1st August Live Class SHH Setup to Virtual Box
~ 3rd August doubt clearing session
~ 4th August live class General Discussion
~ 7th August live class Linux Introduction Evolution of Linux Differences b/w Windows Linux Monolithic Kernel
~ 8th August Live Class Kernel Boot Process Linux Runlevels Filetypes in Linux Package Mangement
~ 10th August Doubt Clearing session
~ 14th August Live Class Componemts in Linux OS File system Linux Commands

~ 17th August Doubt Clearing Session
~ 21st August Live Class Archiving Compression VI Editor Runlevel changes Accounts User Mangement
~ 22nd August live class Searching in LinuxGrep Vi Editor File Permissions
~ 24th August Doubt clearing session
~ 28th August Live Class Scp Passwordless SSH Cronjobs Service Mangement
~ 30th August Live class Linux in AWS Part1
~ 31st August Doubt Clearing session
~ 4th September Live Class Linux in AWS Part2
~ 5th September live Class GIT Introduction
~ 7th September doubt clearing session
~ 11th September Live class GIT-Working with Repos History Check
~ 12th September Live Class Merging Merge Conflicts
~ 14th September Doubt Clearing session
~ 18th September live class Rebasing Interactive Rebasing Amend
~ 19th September Live class Revert Reset Cherry Pick Reflog Checkout
~ 21st September Doubt Clearing session
~ 25th September Live Class DVC introduction
~ 26th September Live Class DVC basic ML use case
~ 28th September Doubt clearing session
~ 3rd October Live Class DVC basic ML use case
~ 5th October Doubt Clearing session
~ 9th October Live class DVC DL use case with tensorflow
~ 10th October Live Class DVC basics DL-Tf use case continued
~ 12th October Doubt clearing session
~ 23rd October Live Class DVC basics DL-Tf usecase till stage 04
~ 25th October Live Class DVC basics DL Tf - final stage COMPLETED
~ 26th October Doubt Clearing session
~ 30th October Live Class DVC -NLP use case
~ 31st October Live Class DVC -NLP usecase completed till plots
~ 2nd November Live Class EXTRA SESSION: CML
~ 7th November Live Class Docker Installation
~ 13th November Live Class Docker Session -1
~ 14th November Live Class Docker Session -2
~ 16th November Doubt Clearing session
~ 20th November Live Class Docker Project
~ 21st November Live Class Docker Project
~ 23rd November Doubt clearing session
~ 27th November Live Class Bank-Note Authentication Docker project
~ 28th November Live Class Docker Task
~ 4th December Live Class Docker with Gpu Nlp Application Deployment
~ 5th December Live Class Docker Compose
~ 18th Dec Live Class Recording Microservice vs Monolithic
~ 19th Dec Live Class Recording Microservice vs Monolithic and Docker Networking
~ 08th Jan Live Class Recordings Kubernetes session 1 Introduction to Kubernetes
~ 9th Jan Live Class Kubernetes session 2: Installation and basics
~ 15th Jan Live Class Kubernetes session 3: Basic Commands
~ 22 Jan Live Class
~ 23rd Jan Live Class
~ Jan 29th Live Class
~ Jan 31th Jan
~ 6th Feb Live Class
~ Feb 12th Live Class
~ 8th Feb live Class
~ 13th Feb Live Class MLFLOW session 01
~ 19th Feb Live Class
~ 20th Feb Live Class Part1 MLFLOW session 02
~ 20th Feb Live Class Part2 MLFLOW session 02
~ 26th Feb live class MLFLOW session 03
~ 27th Feb Live Class MLFLOW session 04
~ March 5th Live Class MLFLOW using AWS Infrastructure
~ March 6th Live Class MLFLOW using AWS Infrastructure - 2
~ 12th March Live Class MLflow using Azure Infrastructure -1
~ 13th march Live Class MLflow using Azure Infrastructure -2
~ 20th March Live Class MLOps using GCP Infrastructure with Terraform automation
~ 26th March Live Class Kubeflow - 1
~ 27th March Live class Kubeflow - 2
~ 2nd April Live Class Kubeflow - 3
~ 3rd April Live Class Part 1 Introduction of TFX based ML Pipeline
~ 3rd April Live Class Part 2 Introduction of TFX based ML Pipeline
~ 9th April Live Class TFX project initial setup
~ 10th April Live Class TFX data ingestion custom component
~ 16th April Live Class Data Validation and Feature Engineering using TFX
~ 17th April Live Class Model training and Model Evaluation using TFX
~ 23rd April Live Class Deployment of source code at s3 bucket and Apache Airflow
~ 24th April Live Class Publishing python projects as python package at PYPI
~ 30th April Live Class Digital Ocean CI CD (Text Classification)
~ 1st May Live Class LSTM TEXT CLASSIFICATION
~ 7th May Live Class End to End Aiops project with infrastructure and endpoint monitoring
~ 8th May Live Class End to End Aiops project with infrastructure and endpoint monitoring

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. Wepresentan 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 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 :

~ 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

<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

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 Tunning
~ 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

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 (Resister,Led,Transistor,Capacitor,Diode)
~ Basic Concepts electicity (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 intergration 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

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