



Data Science Course



FRONTLINES EDUTECH PRIVATE LIMITED



ABOUT US

Frontlines Edutech Private Limited envisions bridging the gap between academia and industry. We understand the pain of unemployment and are trying to wipe off the words of impossible and unemployment from the brains of youth by training and turning them into diamonds. We provide tailor-made, hands-on, and need-based programs on par with industry standards. We believe that the youth of today is the future of tomorrow. And so our courses are tutored by experts from top industries who are passionate about mentoring the youth. Since its inception, Frontlines Edutech has earned the trust and gained recognition from thousands of learners across the state.



FOCUSED ON REFINING TALENT AND TURNING LEARNERS INTO HIGHLY SKILLED PROFESSIONALS SOUGHT AFTER BY TOP COMPANIES.



OUR MISSION AND VISION

At Frontlines Edutech Private Limited, our mission is to bridge the gap between academia and industry by offering practical, customized training programs that equip youth with the skills needed for easy employment. We aim to eliminate unemployment by fostering a mindset of possibility and transforming learners into skilled professionals. With expert industry mentorship, we empower the next generation with the tools to succeed in today's competitive world.

It is in our vision that each young person be prepared with the skill, confidence, and opportunity to succeed in their career. We want to lead an online skills development platform that provides focus on determining the needs of industries through education, so that it would be well-prepared in nurturing professionals who will shape the future of employment. Innovative training eliminates unemployment and helps in unlocking the potential of thousands of learners.





WHY CHOOSE US

We are Success Trainers, Mentors, and Motivators, passionate about helping you reach your full potential with a blend of warmth, wisdom, positivity, and proactivity. Our goal is always your highest good and greatest growth, offering support that is both caring and daring, fun and focused. We ensure learning is not only powerful but also enjoyable, making every step of your growth journey meaningful and engaging.





WHY LEARN THIS COURSE

- This course is for those who want to start their coding journey, whether a beginner or an expert.
- This course covers basics to advanced concepts straightforwardly and systematically.
- We take special care of Non-IT students
- Easy to learn
- Beginner Friendly
- The diversity of our learners adds richness to course discussions and interactions.





DELIVERABLES

8.PLACEMENT UPDATES

7.Q&A SESSIONS

6.COURSE COMPLETION CERTIFICATE

5.INTERVIEW GUIDANCE

4.LINKEDIN PROFILE BUILDING

2.RESUME BUILDING

3.DAILY ASSIGNMENTS

9.ON-DEMAND VIDEO COURSE CONTENT

10.DOWNLOADABLE RESOURCES

1.FROM SCRATCH TO MASTER LEVEL TRAINING





TOP COMPANIES HIRING

Infosys

tcs TATA
CONSULTANCY
SERVICES

accenture

cognizant

Capgemini

wipro

IBM

BOX8

KPMG



TOP CITIES FOR DATA SCIENCE DEMAND

- BANGALORE (BENGALURU)
- GURGAON (GURUGRAM)
- HYDERABAD
- PUNE
- MUMBAI
- CHENNAI
- NOIDA

KEY SECTORS DRIVING DEMAND FOR DATA SCIENCE

- TECHNOLOGY AND SOFTWARE DEVELOPMENT
- DATA SCIENCE AND ANALYTICS
- ARTIFICIAL INTELLIGENCE AND MACHINE LEARNING
- MEDIA AND ENTERTAINMENT
- STARTUPS AND TECH INCUBATORS



DATA SCIENCE SYLLABUS

PYTHON

COMPLETE PYTHON

- Course Introduction
 - Why Python?
 - Who Uses Python Today?
 - What Can We Do with Python?
 - How Python Developed and Supported
 - Python – Technical Strengths
 - What's Next?
- Python Interpreter
- Program Execution – programmer's view, Python's view
- Installation
 - Python
 - Related Software: PyCharm, Anaconda
 - Setup and Configure Python on Laptop
 - IDLE: UI, Usage, Features
- All Numeric types in Python, coding/Hands-on
- Python Variables, objects, References, Shared References, coding/Hands-on
- Garbage Collection of objects
- All built-in types in Python: Strings, Lists, Dictionaries, Tuples, Sets, Files
- Python Statements - coding/Hands-on
- Assignments, Expressions and Prints
 - if-else, if-elif-else, if-else ternary expression
 - while and for loops
 - Comprehensions vs regular

- Parallel Traversals: map and zip functions
- . Other important functions: range, len, enumerate
- Iterations and Comprehensions- coding/Hands-on
- Python online Documentation
- Python Functions – def, nested functions
- Variable Scopes – basics, LEGB rules, global, nonlocal
- Function Arguments - coding/Hands-on
 - Arguments and shared references
 - Arguments passing basics
 - Arguments matching basics
 - Arguments matching syntax
 - Multiple Results
- Advanced Function Concepts – Recursive functions, Attributes, Annotations, lambda
- Generators and comprehensions – Generator functions, yield, Generator expression
- Python Modules
 - Definition, why modules?
 - Typical Python program architecture
 - Import statement - coding/Hands-on
 - How Import works: Find it, Compile it, Run it
 - Standard library modules
 - __pycache__ folder for byte code files
 - Module search path



- Module Coding Basics
 - Module creation
 - import statement, from statement, from * statement
 - Module Namespaces, Namespace dictionaries: __dict__
 - Reloading modules
- Module Packages
 - Package import basics
 - Why Package imports?
 - Relative import basics
 - Why relative imports?
 - Package Namespaces
- Advanced Module Topics
 - Data hiding in modules - coding/Hands-on
 - Mixed usage modes: name and main - coding/Hands-on
 - The as extension for import and from - coding/Hands-on
- Introduction to Python Classes- coding/Hands-On
 - Why Classes?
 - Classes, constructors and Instances
 - Method calls
 - Attribute inheritance search
 - OOP is about code reuse
 - Subclassing by Inheritance
 - Polymorphism in Action
 - Class vs instance attributes
 - Storing objects in DB – Pickles & Shelves
- Coding with Classes - coding/Hands-on
 - Abstract super classes
 - Nested classes
 - Classes vs Modules
 - Namespace dictionaries
 - LEGB scopes rule revisited
- Operator Overloading- coding/Hands-on
 - Constructors: __init__
 - Indexing, Slicing: __getitem__ and __setitem__
 - Attribute Access: __getattr__ and __setattr__
 - String Representation: __repr__ and __str__
 - Right side and In-Place Uses: __radd__ and __iadd__
 - Call Expressions: __call__
 - Comparisons: __lt__, __gt__ and others
 - Boolean Tests: __bool__ and __len__
 - Destructors: __del__
- Special Features of Classes
 - Inheritance: “IS-a” relationship
 - Composition: “HAS-a” relationship
 - Pseudo private class attributes
 - Bound and unbound method objects
 - Class objects
 - “Mix-In” classes
- Advanced Class Topics
 - “New style” class model
 - Diamond inheritance change
 - MRO: Method Resolution Order
 - Slots: Attribute Declarations
 - Properties: Attribute Accessors
 - Static and Class methods
 - The “super” built-in function
- Exception Basics- coding/Hands-on
 - Why Exceptions?
 - Default Exception handler
 - Catching Exceptions
 - Raising Exceptions



- Coding Exceptions - coding/Hands-on
 - The try/except/else statement
 - try/finally statement
 - raise statement
 - assert statement
 - with/as context managers
 - Nesting Exception Handlers
- Exception Objects
 - Class based exceptions
 - Why Exception hierarchies?
 - Built-in Exception Classes
 - Custom Exceptions
- Regular Expressions with Python
 - What are regular expressions?
 - regex module in python
 - The match Function
 - The search Function
 - Matching vs searching
 - Search and Replace Meta characters, advanced patterns

REGULAR EXPRESSIONS WITH PYTHON

- What are regular expressions?
- Regex module in python
- The match Function
- The search Function
- Matching vs searching
- Search and Replace
- Meta characters, advanced patterns

DATA LIBRARIES

- Introduction to numpy
- Creating arrays
- Indexing Arrays
- Array Transposition
- Universal Array Function

- Array Processing
- Array Input and Output
- Introduction to Pandas, Series, Dataframes
- Data reading with Pandas
- Data cleaning with Pandas
- Data wrangling with Pandas
- Data selection with Pandas
- Data extraction with Pandas
- Introduction to Matplotlib
- Data Visualization with matplotlib

MATHEMATICS

- Linear Algebra
- Statistics
- Probability
- Differential Calculus

MACHINE LEARNING

INTRODUCTION TO MACHINE LEARNING

- What is Machine Learning
- Why use Machine Learning
- Examples of ML applications
- Types of ML Systems
- Supervised Learning
- Unsupervised Learning
- Batch vs Online Learning
- Instance-based vs Model-based Learning
- Challenges of Machine Learning
- Overfitting vs underfitting training data
- All phases of End to End ML Project



CLASSIFICATION MODELS

- Binary Classifier
- Performance Measures
 - Accuracy
 - Confusion Matrix
 - Precision and Recall
 - ROC Curve
- Multi Class Classification
- Error Analysis
- Multi Label Classification
- Multi Output Classification

REGRESSION MODELS

- Linear Regression
- Gradient Descent
 - Batch Gradient Descent
 - Stochastic Gradient Descent
 - Mini-batch Gradient Descent
- Polynomial Regression
- Regularized Linear Models
 - Ridge Regression
 - Lasso Regression
 - Early Stopping
- Logistic Regression

SUPPORT VECTOR MACHINES

- Linear SVM Classification
- Soft Margin vs Hard Margin Classification
- Nonlinear SVM Classification
- SVM Regression Models

DECISION TREES

- Introduction to Decision Tree
- Training Decision Tree
- Visualizing Decision Tree
- Estimating Class Probabilities
- The CART Training Algorithm
- Computational Complexity
- Gini Impurity vs Entropy
- Regularization of Hyperparameters
- Regression, Instability

RANDOM FORESTS

- Ensemble Learning
- Voting Classifiers
- Bagging and Pasting
- Bagging & Pasting in sci-kit Learn
- Out of bag evaluation
- Random Patches and Random Subspaces
- Random Forests
- Extra Tree and Feature importance
- Introduction to Boosting
- AdaBoost and Gradient Boost
- Stacking

UNSUPERVISED LEARNING TECHNIQUES

- Clustering
- K-Means Algorithm and Limits of K-Means
- Preprocessing using Clustering
- Semi-supervised Learning using Clustering
- DBSCAN



DEEP LEARNING

INTRODUCTION TO ARTIFICIAL NEURAL NETWORKS WITH KERAS

- Biological Neuron
- The Perceptron
- Multilayer Perceptron and Back propagation
- Regression MLPs
- Classification MLPs
- Implementing MLPs with Keras
- Building an Image classifier using Sequential API
- Building Regression MLP using Sequential API
- Building complex Models using Functional API
- Saving and Restoring Model
- Using Callbacks
- Fine tuning neural network hyperparameters
- Number of Hidden layers
- Number of neurons per hidden layer
- Learning rate, Batch size and other hyperparameters

TRAINING DEEP NEURAL NETWORKS

- Vanishing/Exploding Gradients problems
 - Glorot and He initialization
 - Nonsaturating Activation Functions
 - Batch Normalization
 - Gradient Clipping

- Reusing Pretrained Layers
 - Transfer Learning
 - Unsupervised Pretraining
- Faster optimizers
- Avoiding overfitting through Regularization
 - L1 & L2 regularization
 - Dropout
 - MC (Monte Carlo) Dropout
 - Max-Norm Regularization

LOADING AND PREPROCESSING DATA WITH TENSORFLOW

- Quick tour of TensorFlow
- Tensors and Operations
- Tensors vs NumPy
- The Data API
 - Chaining Transformations
 - Shuffling Data
 - Preprocessing Data
 - Prefetching Data
- Preprocessing Input Features
 - Encoding categorical Features using one-hot vectors
 - Encoding categorical features using Embeddings
 - Keras Preprocessing Layers
- TF Transform
- TFDS Project



GENERATIVE AI OVERVIEW

- Introduction to Generative AI
- Architecture of Generative AI
- Introduction to LLM (Large Language Models)
- Foundational or Public LLMs
- Various components in LLMs
- Character splitters
- Embeddings and Vector stores
- Significance of RAGs in LLMs
- Prompts in LLMs
- Prompt Engineering





FRONTLINES EDUTECH

Building Trust & Careers

Follow Us



FRONTLINES MEDIA



FRONTLINES EDUTECH



FRONTLINES MEDIA-IGNITED-MINDS



For Enquiry Contact Us:



📞 +(91)-83330 77727

✉️ media.frontlines@gmail.com

🌐 www.frontlinesedutech.com