



COURSE SYLLABUS

ARTIFICIAL INTELLIGENCE

Industrial Training

(3 MONTHS)



PH: 0481 2411122, 09495112288
E-Mail: info@faithinfosys.com

Marette Tower
Near No.1 Pvt. Bus Stand
Vazhoor Road
Changanacherry-01

www.faithinfosys.com

Target Audience

This course is specially designed for **B.Tech/B.E, M.Tech/M.E, and all other IT-related graduates and postgraduates** who are interested in building strong expertise in **Data Science** using a language-neutral and industry-oriented approach.

Mission

Professionalism has become a key requirement in today's job market, where companies seek well-qualified, skilled, and adaptable talent. Keeping this in view, we train students to be industry-ready, ensuring they are second to none. Our motto is Quality Education with Performance-Oriented Training, empowering learners with both theoretical knowledge and practical skills.

Course Overview

This course offers a comprehensive introduction to the core concepts, tools, and techniques of Data Science, providing students with hands-on experience in data collection, cleaning, and preprocessing, exploratory data analysis and visualization, statistical methods, and machine learning algorithms. Learners will also gain proficiency in Python for data analysis, working with large datasets and big data platforms, and applying data science to solve real-world problems.

Live Project Work

The Live Project phase enables students to apply their learning in real-world scenarios, covering the entire Data Science lifecycle— from understanding project requirements and data wrangling to exploratory analysis, feature engineering, model building, optimization, and deployment. This hands-on experience builds confidence and equips students with the practical skills needed to deliver effective, data-driven solutions in industry projects.

Add on Programs

- Personality Development
- Listening Skills
- Communication Skills
- Interview Skills
- Group Discussion

- Topics Presentation
- Awareness of IT Trends
- Aptitude Tests
- Technical Tests
- Mock Interview

This course covers concepts of software engineering. It intends to lay a foundation for software designing and professional practice by conveying fundamental knowledge about software development process, requirements analysis, design techniques, and testing methods. The course emphasizes on modeling skills with the Unified Modeling Language (UML).

Section 1: Overview of Software Development

- 1.1 Software Engineering Concepts
- 1.2 Software Engineering Development Activities
- 1.3 System Development Models and Approaches
- 1.4 Software Process and Project Management.

Section 2: Software Development Life Cycle

- 2.1 Requirement Elicitation
- 2.2 Analysis
- 2.3 System Design
- 2.4 Object Design
- 2.5 Implementation
- 2.6 Testing

Section 3: UML

- 3.1 Introduction
- 3.2 Need of UML
- 3.3 Use Case Driven Object Oriented Analysis
- 3.4 Use Case Model
- 3.5 Use Case Diagram
- 3.6 Activity Diagram
- 3.7 Sequence Diagram
- 3.8 Collaboration Diagram
- 3.9 Class Diagram

Section 4: Project Development Models

- 4.1 Waterfall
- 4.2 V model
- 4.3 Prototype model
- 4.4 Spiral model

Section 1: HTML5- The Static Web Page Creation

- 1.1 HTML5 Introduction
- 1.2 Structure
- 1.3 Elements
- 1.4 Semantics
- 1.5 Audio & Video
- 1.6 Section & Article
- 1.7 Canvas, Aside
- 1.8 Drag & Drop
- 1.9 Forms & Form Elements

Section 2: CSS 3 - The Presentation Semantics

- 2.1 CSS Properties, Selectors, Style Declaration Types
- 2.2 Colors, Backgrounds, Text and Fonts
- 2.3 Images, Links, Tables and List
- 2.4 Borders, Padding, Margin
- 2.5 Cursor, Dimension, Scrollbars, Visibility and Positioning
- 2.6 Pseudo class & Elements, @Rules(import, font-face, charset)
- 2.7 Filters, Media Types, Printing and Layouts
- 2.8 Flexbox Layout, Alignment, and Responsive Design

Section 3: Bootstrap

- 3.1 Introduction to Bootstrap
- 3.2 Bootstrap Grid System
- 3.3 Creating Layouts with Bootstrap
- 3.4 Bootstrap CSS - Understanding the CSS
- 3.5 CSS Customization / Skins
- 3.6 Responsive Web design with Bootstrap
- 3.7 Single Page Responsive site with Bootstrap
- 3.8 Bootstrap Layout Components
- 3.9 Bootstrap Plug-ins :
 - ✓ Transition
 - ✓ Modal
 - ✓ Dropdown
 - ✓ Scrollspy
 - ✓ Tab
 - ✓ Tooltip
- 3.10 Building Websites with Bootstrap

Section 4: JavaScript - The Interpreted Programming Language

- 4.1 Interpreted Programming Languages
- 4.2 Integrating JavaScript with HTML
- 4.3 Variables in JavaScript
- 4.4 Operators in JavaScript
- 4.5 Expressions in JavaScript
- 4.6 Arrays in JavaScript
- 4.7 Handling Loops & Decision structures
- 4.8 Executing Conditional statements
- 4.9 Working with Functions

Section 5: TypeScript - The Typed Superset of JavaScript

- 5.1 Introduction to TypeScript
- 5.2 Types and Variables
- 5.3 Functions
- 5.4 Objects, Interfaces & Classes
- 5.5 Enums & Generics
- 5.6 Advanced Type Concepts

Section 6: AngularJS

- 6.1 AngularJS Introduction
- 6.2 Single Page Application (SPA)
- 6.3 Directive, Filters and Data Binding
 - ✓ What are Directives?
 - ✓ Using Directives and Data Binding Syntax
 - ✓ Data-Binding Example using AngularJS Directives
 - ✓ Iterating with the ng-repeat Directive
 - ✓ ng-repeat Example
 - ✓ The AngularJS API Reference for Directives
 - ✓ Using Filters
 - ✓ Using Filters Demo
- 6.4 Views, Controllers and Scope
- 6.5 Modules, Routes And Factories
 - ✓ Creating a Module
 - ✓ Creating a Controller in a Module
 - ✓ The Role of Routes
 - ✓ Defining Routes
 - ✓ Defining Routes Demo
 - ✓ Using Factories and Services
 - ✓ The Role of the Factory

PROGRAMMING LANGUAGE (Python-MySQL)

Course Description:

The contents of this course are a comprehensive solution that moulds you to a Python specialist by providing a combination of on hand labs and the training provided in the class. It helps the trainee to learn and develop various Python technology applications that definitely meets the current industry needs.

Section 1: Python – Understanding the Preliminaries Evolution of .NET

- 1.1 Installation and Environment Variables setting
- 1.2 Statements and Syntax
- 1.3 Variables Assignment
- 1.4 Identifiers
- 1.5 Basic Inputs
- 1.6 Numbers
- 1.7 Operators

Section 2: Conditionals and Loops

- 2.1 If else statement
- 2.2 If elif else statement
- 2.3 Switch
- 2.4 While loop, For loop
- 2.5 Break, Continue, Pass

Section 3: Sequences: String, List and Tuple , Mapping and Set types

- 3.1 Sequences
- 3.2 Strings and operators
- 3.3 String built-in Functions
- 3.4 List and methods
- 3.5 Tuples
- 3.6 Mapping Type: Dictionaries
- 3.7 Mapping type built-in functions
- 3.8 Set Type –operators, built-in methods

Section 4: Object-Oriented Programming

- 4.1 Object-Oriented Programming
- 4.2 Classes
- 4.3 Class Attributes
- 4.4 Instances
- 4.5 Instances and Attributes
- 4.6 Binding and Method Invocation
- 4.7 Static Methods and Class Methods
- 4.8 Inheritance

Section 5: File handling (read/write CSV, JSON) ADO.NET

- 5.1 Basic File Handling Concepts
- 5.2 File Built-in Functions
- 5.3 Handling CSV Files
- 5.4 Handling JSON Files

Section 6: Errors And Exceptions

- 6.1 Exceptions in Python
- 6.2 Detecting and Handling Exceptions
- 6.3 Context Management
- 6.4 Exceptions as Strings
- 6.5 Raising Exceptions
- 6.6 Assertions
- 6.7 Standard Exceptions
- 6.8 Creating Exceptions

Section 7: NumPy

- 7.1 Introduction to NumPy
- 7.2 NumPy arrays and operations
- 7.3 Random Number Functions in NumPy
- 7.4 Introduction to pandas
- 7.5 DataFrames and Series in Pandas
- 7.6 Reading and Writing data in Pandas
- 7.7 Data manipulation: filtering, grouping, merging

Section 8: Data Visualization

- 8.1 Introduction to Matplotlib and Seaborn
- 8.2 Line plots, bar charts, histograms, scatter plots
- 8.3 Customizing plots

Section 9: Python Framework - Django

- 9.1 Introduction to Django
- 9.2 Install Django
- 9.3 MVC: Model, View and Template
- 9.4 About the 3 Core Files: models.py, urls.py , views.py
- 9.5 Setting up database connections
- 9.6 Managing Users & the Django admin tool
- 9.7 Django URL Patterns and Views
- 9.8 Django Forms

Section 1: Relational Database Basics

- 1.1 Brief History of MySQL
- 1.2 Relational Databases and Popular Databases
- 1.3 SQL Statements Section

Section 2: Data Manipulation Language (DML)

- 2.1 INSERT
- 2.2 UPDATE
- 2.3 DELETE
- 2.4 SELECT

Section 3: Data Definition Language (DDL)Sequences

- 3.1 CREATE
- 3.2 ALTER
- 3.3 DROP

Section 4: Sub-Queries, Joins and Unions with MySQL

- 4.1 Order By, Like , And & Or, Where , Between
- 4.2 Joins & Unions
- 4.3 Aggregate Functions and Grouping

Section 5: Constraints and Normalization

- 5.1 Understanding Primary and Foreign Keys
- 5.2 Understanding Database Normalization
- 5.3 Views and Triggers

Section 1: Introduction to Data Science

- 1.1 What is Data Science?
- 1.2 Data Science lifecycle
- 1.3 Data Analyst vs Data Scientist
- 1.4 Real-world applications

Section 2: Data Wrangling and Preprocessing

- 2.1 Handling Missing Values
- 2.2 Removing Duplicates
- 2.3 Encoding Categorical Variables
- 2.4 Feature Scaling
- 2.5 Outlier Detection & Treatment

Section 3: Exploratory Data Analysis (EDA)

- 3.1 Univariate Analysis
- 3.2 Bivariate Analysis
- 3.3 Multivariate Visualization
- 3.4 Correlation and Covariance

Section 4: Statistical Foundations

- 4.1 Descriptive Statistics
- 4.2 Probability Basics
- 4.3 Inferential Statistics

Section 5: Feature Engineering & Selection

- 5.1 Feature Engineering
 - Creating new features from existing ones
 - Extracting date, time, and text-based features
 - Binning and transformations (log, square root)
- 5.2 Feature Selection
 - Filter methods (correlation, variance threshold)
 - Wrapper methods (recursive feature elimination)
 - Embedded methods (Lasso regularization)
- 5.3 Dimensionality Reduction (Intro Only)

Machine Learning

Section 1: Introduction to Machine Learning

- 1.1 What is Machine Learning?
- 1.2 Types of Machine Learning
 - Supervised vs Unsupervised Learning
 - Brief intro to Reinforcement Learning
- 1.3 Real-world ML use cases

Section 2: Scikit-learn & ML Workflow Basics

- 2.1 view of Scikit-learn
- 2.2 Steps in a typical ML workflow:
 - Importing data
 - Preprocessing
 - Model training and evaluation
- 2.3 Hands-on: Basic classification/regression pipeline using Scikit-learn

Section 3: Supervised Learning Algorithms

- 3.1 Linear Regression
- 3.2 Logistic Regression
- 3.3 Decision Trees
- 3.4 Random Forest
- 3.5 K-Nearest Neighbors (KNN)
- 3.6 Support Vector Machines (SVM)

Section 4: Unsupervised Learning

- 4.1 K-Means Clustering
- 4.2 Hierarchical Clustering
- 4.3 PCA (Dimensionality Reduction)

Section 5: Model Evaluation & Validation

- 5.1 Train/test split
- 5.2 Cross-validation
- 5.3 Classification Metrics
- 5.4 Accuracy, Precision, Recall, F1-score, ROC-AUC

Artificial Intelligence

Section 1: Introduction to AI

- 1.1 What is AI?
- 1.2 History and evolution of AI
- 1.3 AI vs ML vs DL
- 1.4 AI subfields (NLP, CV, Robotics, etc.)

Section 2: Search Algorithms & Problem Solving

- 2.1 State space search
- 2.2 BFS, DFS
- 2.3 Heuristics and A* Algorithm
- 2.4 Constraint satisfaction problems

Section 3: Knowledge Representation & Reasoning

- 3.1 Logic-based representation
- 3.2 Propositional and First-order logic
- 3.3 Inference rules, truth tables

Section 4: Natural Language Processing

- 4.1 Text preprocessing: tokenization, stemming, lemmatization
- 4.2 Bag of Words, TF-IDF
- 4.3 Sentiment analysis
- 4.4 Named Entity Recognition (NER)

Section 5: Neural Networks & Deep Learning Basics

- 5.1 Perceptron and activation functions
- 5.2 Feedforward networks and backpropagation
- 5.3 Keras/TensorFlow intro
- 5.4 Build and evaluate basic neural networks

Section 6: AI Applications and Trends

- 6.1 AI in healthcare, finance, robotics, etc.
- 6.2 Ethical considerations in AI



Live Project (The Mapping of what you learn...)

A software development process provides a basis for the organized production of software, using a collection of predefined techniques and notations. The process starts with the formulation of the problem, and then continues through analysis, design and implementation.

Development Stages :

1. System Conception

Conceive an application and formulate tentative requirements. It deals with genesis of an application.

2. Analysis

Analysis focuses on creation of models. It specifies what must be done, not how it should be done. Developers must fully understand the problem before addressing the additional complexities of design. During analysis developers consider the available sources of information and resolve ambiguities.

3. System Design

During system design, the developer makes strategic decisions with broad consequences. The system designer must understand how a new system interacts with other system, and the system must support future modifications

4. Class Design

Developer expands and optimizes analysis models, there is a shift in emphasis from application concepts toward computer concepts. Developers choose algorithms to implements major system functions.

5. Implementation

Implementation is the stage for writing the actual code. Developers map design elements to programming language and database code.

6. Testing

During testing, Testers once again revisit the original business requirements and verify that the system delivers the proper functionality. It also uncovers the accidental errors that have been introduced.
