



PYTHON FULL STACK

Build a strong understanding of Python programming and web development technologies to become a Full-Stack Python Developer with React. Get hands-on experience with various Django frameworks for backend development and React for frontend development. Create a portfolio of projects by developing end-to-end web applications using Django and React.

HTML & CSS

In this course, you will learn how to design and implement layouts that adapt seamlessly across different screen sizes using techniques like CSS Flexbox, media queries, and Tailwind CSS. The course will guide you through structuring HTML pages and applying styles to make websites responsive and user-friendly.

Software Development Fundamentals

- Importance of Frontend
- Fundamentals
- Syntax
- Debugging
- Tweaking Code

HTML Basics

- Getting Started
- HTML Elements
- HTML Headings
- HTML Paragraphs
- HTML Button
- HTML Attributes
- HTML Images
- HTML Hyperlinks
- HTML Containers
- HTML Lists

CSS Basics

- CSS Rulesets
- CSS Properties
- CSS Colors
- CSS Units
- CSS Box Model
- CSS Margins
- CSS Paddings
- CSS Borders

Intermediate CSS

- Block vs inline elements
- CSS Selectors
- CSS Inheritance
- CSS Cascading
- Class Selector
- Id Selector
- Type Selector
- CSS Specificity

CSS Layouting

CSS Flexbox

- Display
- Flex-direction
- Justify-content
- Align-items
- Flex-wrap
- Flex-row
- Align-content
- Align-self
- Flex-grow
- Flex-shrink
- Order

Sizing Elements

- Overflow
- Box Sizing
- Content Box
- Border Box

CSS Media Queries

- Media Query
- Media Types
- Media Features
- Width, min-width, max-width
- Height, min-height, max-height
- Orientation
- Combining Multiple Media Queries
 - Logical Operators

CSS Grid

- Implicit and Explicit Grid
- Grid rows and columns
- Grid Positioning and Alignment

CSS Positioning

- Position Property
 - Static, fixed, relative, absolute
- Changing Element's Position
 - Relative Positioning
 - Absolute Positioning
 - Fixed vs Absolute
- Positioned Ancestors

Tailwind CSS

- Introduction to Tailwind CSS
 - Why Tailwind CSS?
 - Advantages
 - Setting up Tailwind CSS
 - Common Utility Classes
- Tailwind Display & Responsive Utilities
 - Flex properties
 - Grid Utilities
 - Responsive Utilities



Course Project

E-commerce Website:

Build a responsive Swiggy website using HTML and CSS

JavaScript

In this course, you will learn JavaScript concepts and programming techniques to build dynamic web applications. You'll master the core features of JavaScript, including DOM manipulation, event handling, arrays, functions, and working with real-time data through HTTP requests. By the end of this course, you'll be able to create interactive and responsive websites that respond to user actions and integrate with real-world data.

Introduction to JavaScript

- JavaScript Introduction
- ECMAScript
- Browser Console

Variables & Data Types in JavaScript

- Declaring Variables
- Understanding Primitive Types and the typeof Operator
- Variable Assignment and Naming Rules

Type Conversion

- Implicit and Explicit Conversion
- Converting Strings to Numbers and Booleans

Sequence of Instructions & Operators

- Understanding Sequence of Instructions and Expressions
- Working with Modulus, Exponentiation, Increment, and Decrement Operators
- BODMAS Rule in JavaScript

Operators

- Relational Operators
- Loose vs Strict Equality and Type Coercion
- Logical Operators
- Conditional Statements
- Nested Conditions and the Switch Statement
- Using the Ternary Operator

String Basics & Methods

- String Concatenation, Template Literals, and Indexing
- String Methods

Introduction to Loops

- while and do-while Loop Syntax
- The for Loop and Nested Loops
- Loop Control: break, continue

Functions

- Function Declaration, Parameters, and Arguments
- Understanding Return Statements and Default Parameters
- Named Function Expressions vs Function Declarations
- Callback Functions

Arrow Functions, Scope & Hoisting

- Arrow Function Syntax and Implicit Return
- Understanding Scope: Global, Function, and Block Scope
- Hoisting with var, let, const, and Functions

Arrays

- Creating Arrays and Accessing Array Elements
- Array Methods

Objects

- Working with Objects: Creating, Accessing, and Modifying Properties
- Iterating Over Objects
- Creating and Iterating Over Arrays of Objects
- Filtering, Mapping, and Sorting Arrays of Objects

Error Handling

- Using try-catch for Error Handling
- The finally Block

JavaScript Functions

- Creating Objects Using Factory Functions and Constructor Functions
- Understanding Object Property Value Shorthand Notations

JavaScript Classes

- Introduction to JS Classes and Constructor Methods
- Inheritance using extends and super()

JavaScript Promises

- Understanding Asynchronous vs Synchronous Code
- Promise States: Pending, Fulfilled, Rejected
- Consuming Promises with .then(), .catch(), and Promise Chaining
- Creating and Consuming Promises
- Using async and await for Simplified Asynchronous Code

Dynamic Javascript

In this course, you'll learn essential JavaScript concepts to build dynamic web applications. Focus areas include DOM manipulation, event handling, schedulers, and HTTP requests. By the end, you'll be able to create responsive websites that interact with users and real-time data.

DOM and Events

- DOM Manipulation
- Event Handling

Callbacks & Schedulers

- Callbacks: Functions as arguments
- Schedulers: `setInterval()`, `setTimeout()`

Event Listeners

- Event Types: `keydown`, `keyup`
- Event Object: `event.type`, `event.target`

HTTP Basics

- Protocols: HTTP, HTTPS
- Request/Response: Start Line, Headers, Body

HTTP Requests in JS

- Methods: GET, POST, PUT, DELETE
- JSON Stringify and Parsing

Forms

- Form Elements
- Form Events

JS Modules

- CommonJS Module Exports
- ES6 Module Exports

Node Packages

- Core Modules: Path, Package
- NPM: Creating Projects, Third-Party Packages

React JS

In this course, you will learn how to build stateful web applications with the ReactJS library. Upon completion, you will be comfortable creating an application in ReactJS from scratch.

Introduction to React JS

- Why React?
- Introducing JSX
- Rendering Elements, Components, and Props

State and Effect Hooks

- `useState()`, `useEffect()`
- Handling Events
- Component Life Cycle

Routing

- `useParams()`, `useNavigate()`
- Protected Route with JWT Authentication

React Context

- use() Provider Component



Course Project

E-commerce Application

Build an e-commerce application
using React JS

Core Python

This course introduces students to Python programming, covering fundamental concepts, data types, operators, control flow, loops, functions, and debugging techniques. Students will also develop problem-solving skills through hands-on practice and exercises. By the end of the course, learners will be able to write Python programs confidently and solve coding problems efficiently.

Introduction to Python

- Programming with Python
- Practice Walkthrough
- Leveraging Gen AI for Accelerated Learning

Variables & Data Types

- Variables and Data Types

Sequence of Instructions

- Sequence of Instructions
- Input and Output Basics

Debugging Basics

- Practice Walkthrough
- How to Debug Your Code?

Type Conversions & Operators

- Type Conversions
- Relational Operators

Logical & Conditional Statements

- Logical Operators
- Conditional Statements

Conditional & Looping Constructs

- Nested Conditional Statements
- Loops
- Nested Loops
- Loop Control Statements

For Loops

- For Loop

Strings

- String Methods
- Comparing Strings
- Naming Variables

Lists

- Lists
- Working with Lists
- Lists and Strings

Functions

- Functions
- Function Arguments

Built-in Functions & Recursion

- Built-in Functions
- Function Call Stack
- Recursion

List Methods & Tuples

- List Methods
- Tuples
- Sequences

Sets

- Sets
- Set Operations

Nested Lists & Formatting

- Nested Lists
- String Formatting

Advanced Python

This course builds on Core Python concepts and introduces advanced topics including matrices, Python expressions, dictionaries, object-oriented programming (OOP), encapsulation, inheritance, abstraction, polymorphism, modules, scope, and error handling. Students will gain the skills needed for advanced Python programming, structured code development, and problem-solving in real-world scenarios.

Matrices & Python Expressions

- Introduction to Matrices
- Introduction to Shorthand Expressions
- Introduction to List Comprehensions

Dictionaries

- Dictionaries
- Working with Dictionaries

Object-Oriented Programming (OOP)

- Introduction to Object-Oriented Programming
- Object-Oriented Programming
- Classes & Objects
- Attributes and Methods
- Encapsulation
- Inheritance
- Abstraction
- Polymorphism

Python Modules & Scope

- Scope & Namespaces

Error Handling

- Errors & Exceptions

Introduction to Databases

Learn about the incredibly prevalent databases today. Through this course, you'll develop strong fundamentals and be proficient in concepts related to Databases and DBMS. Most importantly, you'll also be able to perform powerful queries on databases using SQL.

Relational Databases

- Introduction to SQL
- Database Design
- ACID Properties

SQL Operations

- CREATE
- INSERT
- SELECT
- UPDATE
- DELETE

Advanced SQL Modelling

- ER Model
- Creating a Relational Database

Advanced SQL

- Joins
- Subqueries
- Indexes
- Views

Django Fundamentals

This course provides a comprehensive introduction to Django, covering the framework's core concepts, project setup, database integration, and web application development. Students will learn to create and manage models, perform CRUD operations, work with class-based views, templates, static files, middleware, authentication, and the Django admin interface.

Django Basics

Django Basics

- Introduction to Django
- Creating a Project & Running the Server
- MVT Architecture
- Creating Apps

Database & Models

- Database Integration
- Creating Models

CRUD Operations

- Performing CRUD Operations

Class-Based Views (CBVs)

- Introduction to Class-Based Views (CBVs)

Templates & Static Files

- Introduction to Templates
- Template and Static File Usage

Model Relationships

- One-to-One Relationships
- Many-to-One Relationships
- Many-to-Many Relationships

User Model Customization

- User Model Modifications

Authentication

- Authentication
- JWT Authentication

Middleware

- Introduction to Middleware
- Middleware Implementation

Admin Customization

- Admin Interface Setup
- Advanced Admin Customization

Debugging

- Common Errors and Debugging

Django REST Framework (DRF)

This course focuses on building RESTful APIs using Django REST Framework (DRF). Students will learn about serializers, views, routers, and authentication mechanisms including JWT and OAuth2. The course also covers CRUD API endpoints, production best practices, and security measures to prepare students for real-world API development and deployment.

Django REST Framework (DRF)

- Introduction to DRF
- Serialization in Context
- Serializers
- Views (APIView, Generic Views, ViewSets)
- Routers
- CRUD API Endpoints
- JWT Authentication
- OAuth2 Authentication

Production & Security

- Production Settings & Security



Course Project

E-commerce Application

Build an Ecommerce backend application using Django

Integration and Deployment

This course teaches how to integrate Django backend with React frontend to build full-stack web applications. Students will learn to connect the frontend and backend, implement CRUD operations, handle authentication, manage state in React, and deploy the application to production.

Django + React Integration

- Introduction to Django and React
- Setting Up Django Backend
- Setting Up React Frontend
- Fetching Data from Django API in React
- CRUD Operations with Django and React
- Authentication with Django and React
- Basic State Management in React
- Final Integration and Building the Full App

Docker Deployment & CI/CD with Jenkins on AWS

- Introduction to Docker
- Dockerfile Creation
- Docker Compose for Multi-Container Applications
- Running a Django App with Docker
- Automating Deployment with Makefile
- Creating and Pushing Docker Images to Docker Hub
- Introduction to Jenkins and CI/CD
- Setting Up Jenkins for Docker Builds
- Automating Docker Builds and Deployments with Jenkins
- Deploying Docker on AWS EC2
- AWS EC2 Security and Networking

Tools

- Git
- Docker
- Makefile
- Jenkins
- AWS