



A Unit of Ascent NextGen Technologies

Python Full Stack

Master's Program

Python Full Stack

Table of Contents

Program Overview	3
Key Features	4
About Ascent Software Training Institute	4
Career Opportunities in Python Full Stack	5
Learning Path	6
Program Outcomes	7
Who Should Attend	8
Program Curriculum	9
Module 01: Python Fundamentals	9
Module 02: Strings & Data Structures	9
Module 03: Functions, Modules & Packages	10
Module 04: Object-Oriented Programming (OOP)	10
Module 05: Exception Handling & File Handling	11
Module 06: SQL & Databases with Python	11
Module 07: HTML5	12
Module 08: CSS3	12
Module 09: TypeScript / Advanced JavaScript	12
Module 10: Django: Web Development	13
Module 11: Django REST Framework (RESTful Web Services)	13
Module 12: MongoDB: NoSQL Database	14
Module 13: Kafka: Distributed Messaging System	14
Module 14: ReactJS & Redux	15
Module 15: Real-Time Project using Python, React & Django	16
Course Outcomes	16
Tools & Technologies Covered	17
Certification	18
Placement Assistance Program	19
Student Feedback	20
Talk to an Admission Counsellor	21



Program Overview

Python Full Stack Development

Build a strong career in the fast-growing field of **Python Full Stack Development** with this industry-focused training program. The program is designed to help students and professionals develop practical skills in designing, developing and deploying modern web applications using Python technologies.

The course covers essential tools and technologies including **Core Python, Advanced Python, Django, Django REST Framework, HTML, CSS, JavaScript, React, and database management (MySQL / PostgreSQL and MongoDB)**, enabling learners to build scalable and dynamic full stack applications.

Through **real-time projects, case studies and hands-on training**, students gain practical experience working with real-world scenarios. The program focuses on developing strong problem-solving skills, backend and frontend integration, and application deployment techniques used in modern software development.

By the end of the program, learners will be able to **build complete full stack applications**, develop **RESTful services**, **manage databases**, **create responsive user interfaces**, and **deploy applications** in real-world environments.

Key Features

Python Full Stack Development



65+ Hours of Instructor-Led Training

Comprehensive sessions from Python basics to complete full stack web applications, plus 30+ hours of assignments.



Hands-On Learning

Build interactive full stack web applications and work on real-world projects to gain practical experience.



Real-Time Projects

Develop full stack and REST API-based web applications and gain experience building scalable web solutions.



Certification Program

Receive the Python Full Stack Program certificate from Ascent Software Training Institute after successful completion.



Comprehensive Curriculum

Python programming, Django backend development, REST APIs, frontend with HTML5, CSS3, JavaScript and React, SQL and MongoDB.



Experienced Instructors

Learn from industry experts and senior full stack developers, with mentorship and guidance.



Flexible Learning Options

Choose online, in-person or hybrid classes. Classroom training at BTM 1st Stage, Bangalore.



Career Support & Placement Assistance

Resume building, interview preparation, job search assistance and job placement programs.

About Ascent Software Training Institute

Ascent Software Training Institute is a leading training institute that provides industry-focused IT courses designed to help students and professionals build successful careers in technology.

Our training programs focus on practical learning, real-time projects and industry-relevant skills. With experienced trainers and a structured curriculum, we help learners gain the knowledge and confidence required to succeed in today's competitive IT industry.

Career Opportunities

in Python Full Stack Development



High-Demand Career Paths

Python Full Stack Developer

- Python Developer
- Backend Developer (Python / Django)
- Frontend Developer (React / JavaScript)
- Web Application Developer
- Software Engineer



Career Growth & Advancement

Start as a junior developer and grow into roles like:

- Senior Python Developer
- Full Stack Architect
- Tech Lead
- Software Development Manager



Entrepreneurial Opportunities

- Build web-based products
- Launch SaaS platforms
- Start your own tech company



Opportunities Across Industries

- Finance
- Healthcare & biotech
- E-commerce & retail
- Marketing & advertising
- Technology & software
- Government & public sector



Freelancing & Remote Work

Many companies offer remote roles and freelancing opportunities, allowing flexibility and global exposure.

Why choose a career in Python full stack development?

- Companies rely on web applications for business operations and scalability.
- With hands-on training and real-time projects, you can build a rewarding career in this field.

Career progression depends on each person's skills, experience and the roles available.

Learning Path

Python Full Stack Development

1

Python Programming Fundamentals

Python basics, data types & variables, control structures, functions & modules

2

Web Development Basics

HTML5 & CSS3, JavaScript fundamentals, responsive web design

3

Python Backend Development

Object-oriented programming, file handling, exception handling

4

Django / Flask Framework

Introduction to Django, models, views & templates, REST API development

5

Database Management

SQL fundamentals, MySQL / PostgreSQL, database integration with Python

6

Frontend Frameworks

React JS basics / Bootstrap, API integration, dynamic web applications

7

Real-Time Projects

Full stack web application, REST API-based project

8

Career Support & Placement

Resume building, interview preparation, job assistance

Program Outcomes

At the end of this Python Full Stack Developer program, you will be able to:

- ✓ **Master core Python** programming concepts including syntax, OOP, modules, packages and exception handling.
- ✓ **Develop backend applications** using Django or Flask with clean, scalable architecture.
- ✓ **Build RESTful APIs** and integrate third-party services for modern web solutions.
- ✓ **Design and manage databases** using MySQL / PostgreSQL with efficient data modelling and queries.
- ✓ **Create responsive frontend applications** using HTML, CSS, JavaScript and React.
- ✓ **Integrate frontend and backend** to develop complete full stack web applications.
- ✓ **Implement authentication, authorization and security** in web applications.
- ✓ **Deploy applications** on cloud platforms for real-world use.
- ✓ **Work on real-time projects** to gain hands-on industry experience.
- ✓ **Build a professional portfolio** with end-to-end full stack applications.
- ✓ **Prepare for technical interviews**, coding tests and career opportunities in Python full stack development.



Who Should Attend

This training program is designed for individuals who want to build or advance their skills in **software and technology**. The following roles and backgrounds are ideal for this program:

- ✓ **Fresh graduates** looking to start a career in **software and technology**.
- ✓ **Professionals** from non-technical backgrounds who want to **transition** into software and tech-related roles.
- ✓ **Junior to mid-level professionals** aiming to **upskill** in the latest **software tools** and **programming** languages.
- ✓ **IT professionals** looking to specialise in Python full stack development and modern development practices.
- ✓ **Working professionals** who want to enhance their productivity and technical skills.
- ✓ Individuals passionate about **technology**, problem-solving and innovation.

No prior programming experience is required: the course starts with Python fundamentals.

Program Curriculum

Python with Django, React & Redux, Django REST Framework, MongoDB and Kafka. The course teaches Python in a practical, hands-on way using industry-standard methods, tools and technologies, and builds algorithmic thinking and real-time problem-solving skills so you can write code that works.

MODULE 01 Python Fundamentals

Python is a high-level, general-purpose, cross-platform programming language created by Guido van Rossum. Its clear syntax makes it easy to learn and highly productive.

- What is Python, why Python and where it is used
- Features, limitations and versions of Python
- Installing Python and the PyCharm IDE
- Writing and running Python code
- Variables, identifiers, indentation and reserved words
- Data types: int, float, complex, bool, str
- Type casting functions
- Operators: arithmetic, relational, equality, logical, assignment, ternary
- Operator precedence
- Input and output: reading input, eval(), command line arguments, formatted printing
- Flow control: if, if-else, if-elif-else
- Loops: for and while
- break, continue, pass and del statements

MODULE 02 Strings & Data Structures

Work with text and Python's built-in collections.

- Defining and accessing strings
- String operators, indexing and slicing
- String methods and formatting
- Lists: creation, indexing, slicing, traversal and methods
- Nested lists and list comprehension
- Tuples: properties, packing and unpacking, methods
- List vs tuple
- Sets: properties, creation and methods
- Dictionaries: access, add, update, delete and methods

PROGRAM CURRICULUM

MODULE 03 **Functions, Modules & Packages**

Write reusable code and organise larger programs.

- Defining and calling functions; return values
- Returning multiple values
- Positional, keyword, default and variable-length arguments
- Global and local variables
- Recursion
- Lambda (anonymous) functions
- filter(), map() and reduce()
- Modules: types, import statements and dir()
- Creating user-defined modules; module search path
- Packages and sub-packages; popular Python packages

MODULE 04 **Object-Oriented Programming (OOP)**

Model real-world problems with classes and objects.

- Classes, objects and the self variable
- Constructors; method vs constructor
- Instance, static and local variables
- Instance, class and static methods
- Setter and getter methods
- Inner classes and nested methods
- Garbage collection and destructors
- Inheritance: single, multiple and multilevel
- Polymorphism
- Abstract methods, abstract classes and interfaces
- Public, private and protected members
- Data hiding, abstraction and encapsulation

PROGRAM CURRICULUM

MODULE 05 **Exception Handling & File Handling**

Handle errors safely and work with files.

- Syntax errors vs runtime errors
- Common runtime errors and abnormal termination
- try ... except, try ... except ... else
- try ... finally
- Argument of an exception; raising exceptions
- Custom exceptions
- Assertions and using them effectively
- Opening, reading, writing and closing files
- Directory and file management
- Storing data with pickle

MODULE 06 **SQL & Databases with Python**

Store and query data in relational databases from Python.

- Introduction to SQL and relational databases
- Connecting to a database from Python (SQLite)
- Creating tables
- Insert, select, update and delete operations
- WHERE, AND & OR, LIKE and LIMIT clauses
- ORDER BY, GROUP BY, HAVING and DISTINCT
- MySQL / PostgreSQL with Django

PROGRAM CURRICULUM

MODULE 07 **HTML5**

The standard markup language for documents displayed in a web browser.

- Introduction to HTML
- HTML structure
- URL reference techniques
- Semantic elements of HTML5
- HTML tags, sections, divisions, buttons & attributes
- Lists, tables and forms
- Form control elements

MODULE 08 **CSS3**

Describe the presentation of pages written in HTML.

- Introduction to CSS
- CSS box model
- Levels of stylesheets
- Selectors in CSS
- Chrome Developer Tools
- Colours, padding, margin etc.
- CSS stylings
- Positioning elements
- Visibility and z-index
- Flex & Grid properties
- Media queries
- Pseudo selectors
- Transitions

MODULE 09 **TypeScript / Advanced JavaScript**

JavaScript is one of the core technologies of the web, alongside HTML and CSS.

- Introduction to JavaScript
- Variables & data types
- Type conversion and type coercion
- Conditional statements: if/else & switch
- For, while and do-while loops
- Functions
- Document Object Model
- Element selectors and traversing the DOM
- Events and event objects
- Local and session storage
- CRUD operations
- Math objects
- Prototype inheritance
- Callback and arrow functions
- Error handling
- Regular expressions
- Form validations
- Generators and sets
- for...of and for...in loops & maps

PROGRAM CURRICULUM

MODULE 10 Django: Web Development

Django is a free, open-source Python web framework that follows the model-template-views (MVT) pattern, so you can build better web apps faster.

Introduction & setup

What a web framework is, MVC & MVT design patterns, Django's history, installing Django, starting a project, the development server, project structure, creating applications, Django commands

Views & URLs

Dynamic web pages, mapping URLs to views, how Django processes a request, URL configuration and loose coupling, dynamic URLs, request and response objects, HTML and non-HTML responses, function-based and class-based views

Templates

Template system basics, rendering templates, context objects, tags and filters, template loading, the include tag, template inheritance

Forms

Django forms and types of forms, ModelForm, base templates for forms, implementing add, edit and delete

Models & ORM

Configuring the database, defining models, ORM operations, inserting, updating, selecting, filtering, ordering and deleting data, changing the database schema

Admin site

Activating and using the Django admin, users, groups and permissions, customising the admin interface

MODULE 11 Django REST Framework (RESTful Web Services)

Django REST Framework is a powerful, flexible toolkit for building web APIs that other applications can use.

- REST API introduction; understanding REST APIs
- Creating REST APIs in Django
- Postman: REST API testing
- Understanding mixins
- Implementing CRUD operations using Django
- Serialization introduction
- Setting up Django REST Framework (DRF)
- Creating REST APIs in DRF
- Serializers and validations
- ModelSerializers
- DRF views and APIViews: CRUD
- ModelViewSet: CRUD
- REST API security: authentication & authorization
- Securing REST services
- Token-based authentication
- JWT authentication

PROGRAM CURRICULUM

MODULE 12 MongoDB: NoSQL Database

MongoDB is a popular open-source NoSQL database that is flexible, scalable and fast.

- Introduction to NoSQL databases
- Understanding the need for NoSQL
- RDBMS vs NoSQL
- MongoDB introduction and key features
- Installing and configuring MongoDB
- Creating databases and collections
- MongoDB operations
- Inserting single and multiple documents
- Updating documents
- Reading documents (querying)
- Deleting databases, collections and documents
- MongoDB and Django integration
- Django, MongoDB CRUD operations implementation

MODULE 13 Kafka: Distributed Messaging System

Apache Kafka is a distributed publish-subscribe messaging and streaming platform for real-time data.

- Introduction to messaging systems
- Types of messaging models (queue, topic)
- Why we need a messaging system
- What is Kafka; Kafka architecture
- Kafka components: producer, consumer, broker, cluster
- Topics, partitions, offsets, consumer groups
- Message retention and the Kafka commit log
- Kafka installation; starting ZooKeeper and the Kafka server
- Starting a producer and a consumer
- Topic operations: create, list, delete, describe
- Publishing and consuming with the console producer/consumer
- Publishing and consuming messages using Python
- Django, Kafka and MongoDB integration

PROGRAM CURRICULUM

MODULE 14 **ReactJS & Redux**

React is an open-source JavaScript library for building user interfaces from components. Redux is a predictable state container for JavaScript applications.

React tools	npm (dependencies & tasks), Babel (ES6 to ES5), Webpack (bundling), build process for DEV & PROD
Advanced JavaScript & ES6	Functional programming, execution context & this, bind/call/apply, template strings, default arguments, rest & spread, arrow functions, destructuring, modules, classes, promises, fetch, let & const, for...of
First React application	Creating a React app, project structure, debugging, component-based design, one-way data flow, the virtual DOM, JSX overview
Components, state & props	Class and functional components, parent and child components, state, setState, props
Events & CSS in React	Event handling in class and functional components, inline and local CSS, manipulating CSS using events
Lifecycle & Hooks	Class component lifecycle methods, useState, useEffect, manipulating CSS using hooks
Forms, routing & storage	React forms and submission, React Router, maps, conditional rendering, local storage with arrays and objects
Redux	Managing application state with Redux: actions, reducers, store and react-redux

PROGRAM CURRICULUM

MODULE 15 Real-Time Project using Python, React & Django

Develop a real-time project using Python, ReactJS, Redux, Django, Django REST Framework, MongoDB and Kafka, working the way a development team does.

React + Redux → Django REST APIs → Django → MongoDB + Kafka

- Git & GitHub: code repository
- pip & venv: packages and virtual environments
- PyCharm: IDE
- Postman: API testing
- JSON: common data format

OUTCOMES Course Outcomes

By the end of this program, you will:

- ✓ Learn Python programming in depth
- ✓ Master Python object-oriented programming
- ✓ Handle exceptions and errors
- ✓ Develop single-page applications using React & Redux
- ✓ Understand MVC vs MVT
- ✓ Secure Django REST APIs with token-based authentication and JWT
- ✓ Work with Kafka (distributed messaging)
- ✓ Work with Python collections: lists, tuples, sets and dictionaries
- ✓ Become proficient in functional programming in Python
- ✓ Understand web development with Django and ReactJS
- ✓ Master Django models, ORM, views, templates and configuration
- ✓ Build REST APIs with Django REST Framework
- ✓ Gain good knowledge of MongoDB (NoSQL)
- ✓ Develop a project using React & Redux, Django, Django REST API, MongoDB & Kafka

Tools & Technologies Covered

Python Backend

Python

Django

Django REST Framework

Flask

Frontend

HTML5

CSS3

JavaScript

TypeScript

React

Redux

Bootstrap

Databases & Messaging

MySQL

PostgreSQL

SQLite

MongoDB

Kafka

APIs & Security

REST APIs

JSON

Token authentication

JWT

Postman

Development Tools

PyCharm

Git

GitHub

pip

venv

Tool names are used to describe the training content only.

Certification



On successful completion of the program, you receive the **Python Full Stack Program** certificate from Ascent Software Training Institute.

Placement Assistance Program



Resume Building

Crafting standout resumes tailored to Python and full stack developer roles.



Career Counselling

Personalised career guidance and a roadmap for job success.



Job Alerts

Regular job alerts and listings for the latest openings.



Job Referrals

Connecting with hiring partners for referrals to companies.



Mock Interviews

Simulating real interview scenarios with expert feedback.



Interview Preparation

Key tips, common questions and best answers for technical interviews.



Soft Skills Training

Developing communication, teamwork and presentation abilities.



Placement Support

Guidance through the hiring process until you land a job.

Placement assistance supports your job search. Final hiring decisions are made by employers, based on each candidate's skills and interview performance.

Student Feedback



Prayag Verma

Python Developer

I am delighted to announce that I've been selected as a full-stack Python developer at Deloitte. My thanks to Ascent for their invaluable support and guidance.



Anitha

Python Developer

I'm delighted to announce my selection as a Python developer at Infosys. I extend my gratitude to the exceptional support provided by the Ascent faculty.

Feedback from Python students, as published by Ascent Software Training Institute. Outcomes are as stated by the students.

Talk to an Admission Counsellor

We have a dedicated team to counsel and guide you to choose the right course for your career goals. Book a free demo class, in the classroom or online.



CALL
+91 90350 37886
080-4219 1321



WEBSITE
www.ascentcourses.com
www.asti.co.in



EMAIL
hr@ascentcourses.com



VISIT US
#96, 2nd Floor, 100 Ft Ring Road, near Mico Layout, above Apollo Pharmacy,
BTM 1st Stage, Bengaluru – 560029



HOURS
Mon–Sat 8:00 AM – 7:30 PM · Sun 8:00 AM – 2:00 PM