

Python Web Development with Django

Course Overview

Django is the most popular and most mature Python web development framework around. It makes it easier to build better Web apps more quickly and with less code. Building web sites with Django is not just smart and efficient, but fun too!

This course is for candidate who wants to learn Django from scratch and work through a fully functional Project. This course is apt for those who have been working on Python and intend to move the expertise to the web. Through the ORM framework candidate can achieve relatively complex functionality quite rapidly.

This Python Web App Development with Django Training Course will give existing Python developers great hands-on experience building robust, commercial web applications with the Django framework.

This training helps candidates through the development of a fully functional and useful product (an e-commerce site) candidate will learn test driven development, how to write maintainable code, efficient Git branching, and much more, all while focusing on the software engineering practices that go into developing great software.

Who Should Attend

Python developers who want to benefit from the functionality of the Django web app framework in order to rapidly build, test and deploy commercial-quality web apps.

Pre-requisites

Delegates should have attended our Python Programming course or have equivalent Python coding experience.

Learning Objectives

After completing this course, students will have a fundamental understanding of how to:

- Understand Django Architecture and its take on MVC (Models, Views & Templates)
- Build and deploy robust Django web apps
- Integrate with RESTful web services
- Unit Testing and Debugging Django apps

Python web development with Django

Course Duration: 30 Hours

Course Outline

- **Introduction to Django**
 - What is Django?
 - Django and Python
 - Django's take on MVC: Model, View and Template
 - DRY programming: Don't Repeat Yourself
 - How to get and install Django
- **Getting started with Django**
 - About the 3 Core Files: models.py, urls.py, views.py
 - Setting up database connections
 - Managing Users & the Django admin tool
 - Installing and using 'out of the box' Django features
- **Django URL Patterns and Views**
 - Designing a good URL scheme
 - Generic Views
- **Django Forms**
 - Form classes
 - Validation
 - Authentication
 - Advanced Forms processing techniques
- **Django & REST APIs**
 - Django REST framework
 - Django-piston
- **Unit Testing with Django**
 - Overview / Refresher on Unit Testing and why it's good
 - Using Python's unittest2 library
 - Test
 - Test Databases
 - Doctests
 - Debugging Best Practices