



Python and Django for Full Stack Web Developer

- 5 Days
- Lecture and Hands-on Labs
- Python and Django Full Stack Web Developer Certification

Course Overview

Geared for experienced web developers new to Python, Introduction to Full Stack Web Development with Python and Django is a five-day hands-on course that teaches students how to develop Web applications using the Django framework. Students will explore the basics of creating basic applications using the MVC (model-view-controller) design pattern, as well as more advanced topics such as administration, session management, authentication, and automated testing. This comprehensive, practical course provides an in-depth exploration of working with the programming language, not an academic overview of syntax and grammar. Students will immediately be able to use Python to complete tasks in the real world.

Who Should Attend

- System Administrators
- Network Engineers
- Software Developers
- Python Enthusiasts


What You'll Learn

You will learn essential Full Stack skills as they relate to Django including:

- Flow Control
- Lists and Tuples
- Working with Files
- Dictionaries and Sets
- Functions
- Errors and Exception Handling
- Using Modules
- Classes
- Django Architecture
- Configuring a Project
- Adding an Application
- Login for Nothing and Admin for free
- Basic Views (AKA Controllers and Templates)
- Querying the Models
- Working with Templates
- Forms
- Automated Testing
- AI LLM prompt engineering for jumpstarting Python Django snippets and solutions

Outline


Certification

-  Lecture + Lab: Alta3 Research Django Certification (OPTIONAL)


AI LLM Toolkit

-  Lecture + Lab: Large Language Model toolkit for AI Solution Assistance

Overview

-  Lecture: Introduction to Django





Software Control Management

-  Lecture + Lab: SCM Option #1 - GitHub
-  Lecture + Lab: SCM Option #2 - GitLab



Django Introduction

-  Lecture + Lab: Introduction to Django




Python Review

-  Lecture + Lab: Creating Classes
-  Lecture + Lab: Class Inheritance
-  Lecture + Lab: Using Classes
-  Lecture + Lab: Python Virtual Environments - venvs



HTML Review



-  Lecture + Lab: HTML Intro
-  Lecture + Lab: CSS Introduction

Django Basics





-  Lecture + Lab: Intro to Django Views
-  Lecture: Introduction to HTTP
-  Lecture + Lab: Controlling HTTP Response Codes

JSON and Django

-  Lecture: Python Data sets vs JSON
-  Lecture + Lab: Python Data to JSON file

-  Lecture + Lab: Returning JSON with Django
-  Lecture + Lab: Making requests with Django


Django Workflow

-  Lecture + Lab: Starting a Django Project
-  Lecture + Lab: Django Project Design
-  Lecture + Lab: Django Modeling and Object-Relational Mapping
-  Lecture + Lab: HTTPRequest objects









Folder Hierarchy

-  Lecture: Django File Hierarchy
-  Lecture + Lab: Django Templates



Django Apps

-  Lecture + Lab: Django App Design - To-Do app


Django Project

-  Lecture + Lab: Init A Django Project
-  Lecture + Lab: Building Models
-  Lecture + Lab: Django Admin Site
-  Lecture + Lab: Creating our Home Page
-  Lecture + Lab: Generic List and Detail Views
-  Lecture + Lab: Sessions
-  Lecture + Lab: Authentication and Permissions
-  Lecture + Lab: Building Forms

Testing

-  Lecture + Lab: Python unittest
-  Lecture + Lab: Testing a Django App

CICD

-  Lecture + Lab: Django and CI Workflows with GitLab

Prerequisites

Next Courses

- Python for Network Automation (5 days)
- Jenkins Automation Server Essentials (2 days)
- Git and GitHub (or Git and GitLab) (2 days)