

## Full Stack Web Developer Series

# Introduction to Full Stack Web Development with Python and Django (TTPS4860)

Modern Skills for Robust Web Development | Python Essentials, Classes, Django, Views, Querying, Templates, Testing & More

## Course Snapshot

- **Course: Introduction to Web Applications with Python and Django (TTPS4860)**
- **Duration:** 5 days
- **Audience & Skill-Level:** This is an **introductory-level** course geared for anyone **new to Python** who wants to use Python and Django in full stack web development projects. Prior scripting or development experience is required.
- **Format / Hands-on:** This course combines engaging instructor-led presentations and practical demonstrations with hands-on programming exercises, challenge labs, use case exploration and engaging group activities. Student machines are required.
- **Flexible Delivery Options:** This course can be delivered for your team or organization **online-live (virtual), onsite in-person, self-paced** or across our immersive **blended learning experience platform (LXP)**.
- **Public Schedule:** This course is currently available on our Public Open Enrollment Schedule.
- **Customizable:** We're flexible! This course agenda, topics, labs, hours and delivery modalities can be adjusted to target your specific training skills objectives, tools and learning goals. Please ask for details.

## Description

Geared for 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. As an integral part of our **Full Stack Web Developer Boot Camp** series, this course teaches you 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.

**Course Topics:** This is the high-level list of topics covered in the course. See below for detailed sessions, lessons and labs:

- The Python Environment
- Getting Started
- 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
- Using the Admin interface
- Basic Views (AKA Controllers and Templates)
- Querying the Models
- Working with Templates
- Forms
- Automated Testing

## Learning Objectives

This course is approximately **40% hands-on**, combining expert lecture, real-world demonstrations and group discussions with machine-based practical labs and exercises. Working in a hands-on learning environment, guided by our expert team, attendees will learn to:

- Develop full-stack web sites based on content stored in an RDMS
- Use python data types appropriately
- Define data models
- Understand the architecture of a Django-based web site
- Create Django templates for easy-to-modify views

- Map views to URLs
- Take advantage of the built-in Admin interface
- Provide HTML form processing

If your team requires different topics, additional skills or a custom approach, our team will collaborate with you to adjust the course to focus on your specific learning objectives and goals.

### Audience

This **introductory-level** Python course is geared for experienced web developers new to Python who want to use Python and Django for full stack web development projects.

### Pre-Requisites

To ensure a smooth learning experience and maximize the benefits of attending this course, you should have the following prerequisite skills:

**Take Before:** Students should have incoming practical skills aligned with those in the course(s) below, or should have attended the following course(s) as a pre-requisite:

- TT4605: Introduction to HTML5, CSS3 & JavaScript (3 days)

**Next Steps / Follow-on Courses:** We offer a wide variety of follow-on courses for next-level web development, Python, React, React Native, Full Stack and more. Please see our **Modern Web Developer / Full Stack Courses & Learning Journey Paths** for options based on your specific role and goals.

**Enhanced Learning Services:** Please also ask about our robust Learning Experience Platform (LXP), Skills Assessment & Skills Prep Services, Skills Immersion Programs & Camps, Coaching and Mentoring Services and Extended Learning Support programs.

---

### Course Topics / Agenda

*Please note that this list of topics is based on our standard course offering, evolved from typical industry uses and trends. We'll work with you to tune this course and level of coverage to target the skills you need most. Topics, agenda and labs are subject to change, and may adjust during live delivery based on audience skill level, interests and participation.*

- |   |   |   |
|---|---|---|
| <p><b>1. The Python Environment</b></p> <ul style="list-style-type: none"> <li>• Starting Python</li> <li>• Using the interpreter</li> <li>• Running a Python script</li> <li>• Getting help</li> <li>• Editors and IDEs</li> </ul> | <ul style="list-style-type: none"> <li>• Reading from the keyboard</li> </ul>   | <ul style="list-style-type: none"> <li>• Basic sorting</li> <li>• List comprehensions</li> </ul>  |
| <p><b>2. Variables and values</b></p> <ul style="list-style-type: none"> <li>• Using variables</li> <li>• Built in functions</li> <li>• String data</li> <li>• Numeric data</li> <li>• Converting data types</li> <li>•</li> </ul>  | <p><b>4. Flow Control</b></p> <ul style="list-style-type: none"> <li>• About flow control</li> <li>• Conditional expressions (<b>if</b>)</li> <li>• Relational and Boolean operators</li> <li>• <b>while</b> loops</li> <li>• Exiting from loops</li> </ul>                                     | <p><b>6. Working with Files</b></p> <ul style="list-style-type: none"> <li>• File I/O overview</li> <li>• Opening files</li> <li>• Reading/writing files</li> </ul>   |
| <p><b>3. Basic input and output</b></p> <ul style="list-style-type: none"> <li>• Writing to the screen</li> <li>• String formatting</li> <li>• Command line parameters</li> </ul>   | <p><b>5. Array-like Types</b></p> <ul style="list-style-type: none"> <li>• About array-like types</li> <li>• Lists and list methods</li> <li>• Tuples</li> <li>• Indexing and slicing</li> <li>• Iterating through a sequence</li> <li>• Sequence functions, keywords, and operators</li> </ul> | <p><b>7. Dictionaries and Sets</b></p> <ul style="list-style-type: none"> <li>• About dictionaries</li> <li>• Creating and using dictionaries</li> <li>• Getting values</li> <li>• About sets</li> <li>• Creating and using sets</li> </ul> |
|   |   | <p><b>8. Functions</b></p> <ul style="list-style-type: none"> <li>• Defining functions</li> <li>• Returning values</li> </ul>   |

- Arguments and arameters
  - Variable Scope
  - Sorting with functions
- 9. Errors and Logging**
- Exception overview
  - Using try/catch/else/finally
  - Handling multiple exceptions
  - Logging setup
  - Basic logging
- 10. Modules and Packages**
- Creating Modules
  - The **import** statement
  - Module search path
  - Creating packages
- 11. Introduction to Classes**
- About OO programming
  - Defining classes
  - Constructors
  - Properties
  - Instance methods and data
  - Inheritance
- 12. Django Architecture**
- Django overview
  - Minimal Django layout
  - Built in flexibility
- 13. Getting Started**
- Sites and apps
- Shared configuration
  - Executing manage.py
  - Starting the project
  - Generating app files
  - App configuration
  - Database setup
  - The development server
- 14. Using cookiecutter**
- What is cookiecutter
  - Creating projects with cookiecutter
- 15. Creating models**
- Defining models
  - Adding data
  - Related objects
  - SQL Migration
  - Simple model access
- 16. The Admin interface**
- Setting up the admin user
  - Using the admin interface
  - Editing data
- 17. Querying Models**
- QuerySets
  - Field lookups
  - Chaining filters
  - Slicing QuerySets
  - Related fields
  - Q objects
- 18. Views**
- What is a view?
  - HttpResponse
  - URL route configuration
  - Shortcut: `get_object_or_404()`
- 19. Templates**
- About templates
  - Variable lookups
  - The **url** tag
  - Shortcut: **render()**
- 20. Forms**
- Forms overview
  - GET and POST
  - The Form class
  - Processing the form
  - Widgets
  - Validation
  - Forms in templates
- 21. Unit Testing**
- Why create tests?
  - When to create tests
  - Using Django's test framework
  - Using the test client
  - Running tests
  - Checking code coverage

### Setup Made Simple! Learning Experience Platform (LXP)

All applicable course software, digital courseware files or course notes, labs, data sets and solutions, live coaching support channels and rich extended learning and post training resources are provided for you in our “easy access, no install required” online **Learning Experience Platform (LXP)**, remote lab and content environment. Access periods vary by course. We’ll collaborate with you to ensure your team is set up and ready to go well in advance of the class. Please inquire about set up details and options for your specific course of interest.

### For More Information

For more information about our training services (instructor-led, self-paced or blended), collaborative coaching services, robust Learning Experience Platform (LXP), Career Experiences, public course schedule, partner programs, courseware licensing options or to see our complete list of course offerings, solutions and special offers, please visit us at [www.triveratech.com](http://www.triveratech.com), email [Info@triveratech.com](mailto:Info@triveratech.com) or call us toll free at **844-475-4559**. Our pricing and services are always satisfaction guaranteed.

**TRIVERA TECHNOLOGIES • Collaborative IT Training, Coaching & Skills Development Solutions**  
[www.triveratech.com](http://www.triveratech.com) • toll free +1-844-475-4559 • [Info@triveratech.com](mailto:Info@triveratech.com) • Twitter TriveraTech