Test Automation with Python / PyTest Essentials - TTPS4832

Implement Testing Strategies with Python using PyTest

Duration: 3 Days
Skill Level: Introductory
Available Format: Instructor-Led Online; Instructor-Led, Onsite In Person ; Blended; On Public Schedule

Test Automation with Python / PyTest Essentials is a three day, hands-on course geared for QA professionals and testers with basic Python expertise that explores automated testing with PyTest. You'll gain proficiency in writing clear and effective automated tests, organizing and managing test cases, and utilizing PyTest's fixtures for optimal configuration of test environments.

What You'll Learn

Overview

Test Automation with Python / PyTest Essentials is a three day, hands-on course geared for QA professionals and testers with basic Python expertise that explores automated testing with PyTest. You'll gain proficiency in writing clear and effective automated tests, organizing and managing test cases, and utilizing PyTest's fixtures for optimal configuration of test environments. You'll also explore interpreting test reports for actionable insights and employing best practices for maintainable test suites. The program emphasizes practical, hands-on experience, guided by an expert, with 50% of the course dedicated to interactive labs, ensuring that you'll develop real-world skills that are directly applicable to their daily work in modern business enterprises.

Throughout the course you'll learn and practice with core functionalities of PyTest, such as parameterization to maximize test coverage and the use of markers for test selection. By exploring PyTest's powerful built-in features, you'll learn how to streamline your testing process and learn how to work with complex testing scenarios. The focus on organizing tests and adopting PyTest conventions will provide you with an in-depth, practical understanding of efficient test structuring, critical for swift troubleshooting and agile adaptation in fast-paced development environments. Lastly, you'll learn how to apply these skills in a practical way, making you job ready right after class. You'll exit this course equipped with the skills required to construct robust test suites and advance your testing capabilities.

Objectives

This course combines engaging instructor-led presentations and useful demonstrations with valuable hands-on labs and engaging group activities. Throughout the course you'll explore:

- Become proficient with pytest from day one by solving real-world testing problems
- Use pytest to write tests more efficiently
- Scale from simple to complex and functional testing
- Write and run simple and complex tests
- Organize tests in files and directories
- Find out how to be more productive on the command line
- Markers and how to skip, xfail and parametrize tests
- Explore fixtures and techniques to use them effectively, such as tmpdir, pytestconfig, and monkeypatch
- Convert unittest suites to pytest using little-known techniques

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 in an **introductory-level** course geared for QA, Test team members and others who want to use the Python testing framework PyTest to implement code testing strategies. Attendees should have prior basic Python scripting experience. Students should have some familiarity with tools to be used in this course: PyCharm, Jupyter Notebook and basic GIT.

Pre-Requisites

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:

- **TTPS4800** Python Programming Basics (3 days)

TTPS4800	Introduction to Python Programming Basics

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.

Optional Pre-Course Prep:

Getting Started (Self-paced review)

- Quick review of Python programming basics and core concepts.
- Introduction to the idea of automated testing and its benefits.
- Installation guide for Python, PyTest, pip, and a simple text editor like Sublime.
- Tools: Python, PyTest, pip, Sublime Text (or similar basic editor).

Day 1: Building Foundations in Automated Testing with PyTest

Introduction to Automated Testing with PyTest

- Grasp the fundamentals of automated testing and where PyTest fits in.
- Understanding the need for automated testing in software development.
- The anatomy of a simple PyTest test case.

- Exploring the user-friendly features of PyTest for beginners.

Writing Basic PyTest Test Cases

- Learn how to write and structure simple test cases in PyTest.
- Introduction to assertions and why they're key to testing.
- The basics of test discovery and naming conventions.
- Understanding PyTest's command-line options to execute tests.
- Lab

Organizing Tests

- Discover how to efficiently organize your tests for clarity and ease of access.
- Grouping tests and understanding the purpose of markers.
- Introduction to fixtures for reusable test setup and teardown.
- Configuring PyTest with a simple pytest.ini file.

Parameterizing Tests

- Explore how to write tests that can run with different inputs.
- Introduction to the concept of parameterization in tests.
- How to use parameterization to cover more test scenarios.
- Simple strategies for efficient test writing.
- Lab

Day 2: Expanding Test Skills with PyTest

Diving Deeper into Fixtures

- Deepen your understanding of fixtures for more complex test scenarios.
- Sharing fixtures across multiple test files with conftest.py.
- Using scope to control setup and teardown behavior.

- Utilizing autouse fixtures for automatic application.

Exploring Built-in PyTest Features

- Learn about the built-in features of PyTest that make writing tests easier.
- Built-in fixtures for common test scenarios.
- Using built-in marks to skip or expect failures in tests.
- Introduction to the -k expression for targeted test runs.
- Lab

Enhancing Test Reports

- Discover how to generate and interpret test reports for better insights.
- Using command-line options for verbose test reports.
- Introduction to pytest-html for generating HTML reports.
- Best practices for reading and analyzing test outcomes.
- Lab

Adopting PyTest Best Practices

- Adopt best practices for maintainable and efficient testing with PyTest.
- Structuring your test project for long-term maintainability.
- Naming conventions and organization tips for easy navigation.
- Techniques for writing clear, concise, and effective test cases.
- Lab

Day 3: Mastering Essential PyTest Techniques

Testing Data-Driven Applications

- Equip yourself with techniques to test applications driven by various data sets.
- Deeper dive into parameterization for data-driven tests.
- Utilizing CSV and JSON files for test data.

- Strategies for managing test data files and directories.
- Lab

Handling Test Failures and Errors

- Understand the nuances of handling test failures and errors gracefully.
- Differentiating between assertion errors and test failures.
- Employing strategies to investigate and resolve test failures.
- Introduction to PyTest's xfail and xpass outcomes.
- Lab

PyTest Assertions and Assumptions

- Master the art of assertions and assumptions for robust test cases.
- Advanced assertions for more comprehensive tests.
- Understanding and using the assume plugin for non-critical assertions.
- Customizing assertion messages for clarity.
- Lab

Practical Test Case Development

- Bring together all the concepts learned to develop practical, real-world test cases.
- Developing tests for common application functionalities.
- Strategies for incrementally building test suites alongside development.
- Tips for keeping tests readable and easy to understand.
- Lab

For More Information

Please contact us or call 844-475-4559 toll free for more information about our training services (instructor-led, self-paced or blended), coaching and mentoring services, public course enrollment or questions, partner programs, courseware licensing options and more.