

SOFTWARE QA TESTING + AI ENGINEER COURSE STRUCTURE 2026

PROGRAMMING LANGUAGE FUNDAMENTALS: PYTHON

60 hours

Basics • Hardware and Software Architecture • Computer and Networking Basics • Software Infrastructure and Applications • Numbering Systems Notation • Open Source and Open Standards • Development Platforms • Integrated Development Environment (IDE) Basics • Compiler and Builder • Programming Language Basics • Data Types • Operators • Statements • Functions • Symbol Presentation and Coding • Array • Basic Programming Patterns • Practicing • Sequential and Binary Search • Sorting • Reversing • Merging • Object-Oriented Programming (OOP Principles: Encapsulation, Inheritance) • Polymorphism • Classes and Objects • Practicing in Python • Using AI tools for programming and training

MANUAL TESTING — CORE TOPICS

60 hours

Software Testing Fundamentals • Testing Types and Levels • Software Quality, SDLC/STLC and the Seven Principles of Testing • Test Documentation (Test Plans, Test Cases, Checklists, Reports, Defect Reports) • Developer / QA Tools (DevTools, Inspectors, Console) • Test Design Techniques (Equivalence Partitioning, Boundary Values, Decision Tables, State Transition, Pairwise, Use Cases) • HTTP & API Fundamentals • Testing Postman for Manual API Testing (Collections, Variables, Environments) • AI in Manual Testing (Test Data Generation, Test Case Generation, Prioritization, Exploratory Testing Assistance)

PYTHON FOR QA, DATABASE & TEST AUTOMATION (UI & API)

128 hours

Python Essentials for QA (syntax refresh, data structures, modules) • SQL for QA (DDL/DML, Joins, Aggregations, Subqueries, Keys, Indexes, Transactions) • Automation Project Architecture— Page Object Model (structure, layers, fixtures) • Git Fundamentals (branching, PR flow) • Locators: HTML/DOM, CSS Selectors, XPath • Pytest (fixtures, parametrization, markers, Selenium WebDriver (Selenium 4, Waits, Actions, JavaScript Executor) • API Testing with Python (requests / HTTPX; contract & schema checks) • Running Tasks and Schedulers (CLI, Make/Invoke; basic orchestration) • Continuous Integration — GitHub Actions (pipelines, caching, matrix) • AI in Automation (test generation aids, flakiness triage, code reviews)

TOTAL THEORETICAL HOURS

248 hours

PRACTICAL WORK ON REAL PROJECTS

80 hours

TOTAL HOURS

328 hours*

* — We may adjust program content based on technology updates while maintaining the total study hours. Changes may occur without prior notice to students.

Free Weekly Webinars. We run additional free webinars every week, reinforcing class topics and reviewing homework. These sessions are not included in the core syllabus and are provided at no cost.