

Course Title: Core Java Programming with Selenium

Automation

Course Description:

This course explores foundational Java programming alongside Selenium WebDriver for automating web browsers. Students will learn to build robust and maintainable automated test suites and scripts for web applications, enhancing their ability to test and debug software.

Course Objectives:

- Understand the fundamentals of Java programming.
- Master object-oriented programming concepts in Java.
- Gain proficiency in using Selenium WebDriver for browser automation.
- Learn to design and implement scalable test frameworks.
- Develop skills to integrate automation scripts with continuous integration tools.

Prerequisites:

- Basic understanding of programming concepts.
- Familiarity with HTML and web technologies is helpful.

Weekly Syllabus Outline:

Week 1: Introduction to Java and Selenium

- Overview of Java and its features.
- Setting up the Java development environment.
- Introduction to Selenium and its role in automation.

Week 2: Java Basics

- Java syntax and basic programming constructs.
- Data types, variables, and operators.
- Control structures: if-else, switch, loops.

Week 3: Object-Oriented Programming in Java

- Classes, objects, and methods.

- Inheritance, polymorphism, and encapsulation.
- Constructors and method overloading.

Week 4: Advanced Java Features

- Interfaces and abstract classes.
- Exception handling and file I/O.
- Collections framework.

Week 5: Introduction to Selenium WebDriver

- Setting up Selenium and integrating with Java.
- Basic operations with WebDriver.
- Navigating pages and interacting with web elements.

Week 6: Advanced Selenium Techniques

- Synchronizing tests using explicit and implicit waits.
- Handling alerts, frames, and multiple windows.
- Executing JavaScript within Selenium.

Week 7: Building Test Frameworks

- TestNG for structuring tests and assertions.
- Annotations, grouping, and sequencing tests.
- Generating reports and logging.

Week 8: Data-Driven Testing

- Reading data from external sources like Excel and databases.
- Parameterizing tests with TestNG.
- Using Apache POI for Excel integration.

Week 9: Page Object Model (POM)

- Design pattern for enhancing test maintenance and reducing code duplication.
- Implementing POM in Selenium tests.
- Best practices for writing scalable test code.

Week 10: Integration with Build Tools and CI/CD

- Maven for managing project dependencies.
- Integrating Selenium tests with Jenkins for continuous integration.
- Automating the build and test cycle.

Week 11: Selenium Grid

- Distributed testing with Selenium Grid.
- Configuring and running tests on multiple browsers and environments.
- Optimizing execution time and resources.

Week 12: Advanced Browser Automation

- Advanced user interactions with Actions class.
- Capturing screenshots for bug reporting.
- Cookie management and browser session handling.

Week 13: Debugging and Optimization

- Debugging techniques for Selenium scripts.
- Profiling and optimizing Java code.
- Best practices for exception handling in automation scripts.

Week 14: Capstone Project

- Project planning and setup.
- Developing a comprehensive test suite using Java and Selenium.
- Implementing POM and integrating with CI/CD pipelines.

Week 15: Project Presentations and Course Wrap-Up

- Final presentations of capstone projects.
- Peer reviews and feedback.
- Course review and final assessment.

Assessment Methods:

- Weekly hands-on assignments to build Java and Selenium skills.
- Regular quizzes to test theoretical knowledge.
- A capstone project that involves creating a full-scale automated testing framework.