

SDET

Automation Testing Course: Software Development Engineer in Test (SDET)

COURSE OVERVIEW

SDET (Software Development Engineer in Test) course is geared towards an onboarding orientation that will train participants in all the tools they will encounter during their projects. From UI automation and Mobile App testing to backend API/webservices testing, this program will train and expose them to the powerful tools in the industry.

SDETs are the new key power player in automation testing for today's rapid software development. Combining testing skills with developer knowledge, they quickly write effective test script framework will help migrate you from manual testing to automation testing. Our SDET program has designed for manual testers who have little to no experience in writing code.

MAJOR TAKEAWAYS

- Core Java Programming
- Selenium WebDriver, TestNG, Maven & CI Tools
- BDD with Cucumber
- Mobile App Testing with Appium
- API/Webservices Testing
- JavaScript End to End Testing Framework Cypress

SCHEDULE

Duration of the course: 10 Weeks (100+ Hours)

1.5 Hours/day, 5 Days of week (Mon-Fri)

Module 1: Overview on Automation

- What is Software testing?
- Ways of Software Testing
- Challenges in Manual testing
- Use of automation Testing
- Tools for Automation Testing
- What is Selenium?
- Features of Selenium
- Limitations in Selenium

Module 2: Java Essentials for Selenium

- Download and install Java
- Setup Java Environment
- How to use Eclipse/IntelliJ IDEA
- Basic Java Program
- Compile and run a Java program
- Understanding console output
- Java Variables and Data Types
- Java Operators
- Conditional statements
- Looping statements
- Single Dimensional array
- Double Dimensional array
- Object class
- String Class
- String methods

Module 3: Java OOPS Concepts

- Classes and Objects
- Java methods
- Passing parameters to the methods
- Call by value and call by reference
- Java Constructor
- Method Overloading
- Constructor Overloading
- this, final keywords
- Static variables and methods
- Java Inheritance
- Method Overriding
- Java Interfaces
- Access Modifiers
- Java Packages
- Exception Handling
- Array List
- Hash Map
- JDBC

Module 4: Selenium WebDriver

- Download WebDriver API
- Create WebDriver Project in Eclipse
- Create WebDriver test case
- Execute test case on multiple browsers
- Locators
 - What are locators?
 - Types of Locators
 - Capturing XPath in different ways
 - How to use ChroPath in Chrome
 - Types XPath
 - Writing XPath
 - Built-in functions in XPath
 - Handling Dynamic elements using XPath
- WebDriver commands
 - Browser commands
 - Get commands
 - Switching commands
 - Navigation commands
 - Conditional commands
 - Implicit, Explicit and Fluent Waits
- Handling Web Elements
 - Textbox/Input box
 - Links
 - Radio Button/Checkbox
 - Dropdown box/Combo box
 - Multi Select dropdown
 - List box
 - Calendar/Date Picker
 - Web Links
 - HTML frames/Iframes
 - Web/HTML Tables
- Mouse actions using Actions class
 - Mouseover
 - Mouse double click
 - Mouse right clicks
 - Drag and Drop
 - Handling Slider/scroll bar
 - Resizing
 - Handling Tooltips
- Working with Robot Class
- Working with Sikuli
- How to use JavaScript Executor
- Browser profiles
- Desired Capabilities

Module 5: Data Driven Testing

- What is data driven testing?
- Apache POI API
- Data Driven Testing using Excel/csv files
- Data Driven Testing using Java HashMap
- Data Driven testing using Databases
- Data Driven testing Text files/properties file

Module 6: TestNG

- What is TestNG? Features of TestNG
- Install TestNG in Eclipse
- How to write TestNG Test case
- Annotations in TestNG
- Understanding testng.xml
- TestNG html Report
- Prioritizing tests
- dependsOnMethods
- Skipping tests
- Grouping methods
- TestNG batch testing
- Parameterization
- Passing parameters using xml
- Parallel testing
- Data Provider
- Invocation methods
- TestNG Listeners

Module 7: Advanced Concepts

- Handling cookies
- Headless browser testing
- Generating Extent Reports
- Capturing Screenshots
- Log4j for logging
- Page Object Model - Creating Page objects
- Page Factory - @FindBy annotation

Module 8: Maven Integration with Selenium

- What is Maven and Why Maven?
- Installing/Configuring Maven
- Creating Maven Project
- Importing Maven Project into Eclipse
- What is POM.xml?
- Adding Dependencies to POM.xml

Module 9: Automation Frameworks

- What is Framework?
- Various Types of Frameworks
- Data Driven framework
- Keyword Driven framework
- Hybrid Driven framework
- Prerequisites for designing frameworks
- Implementation of Hybrid Driven Framework from the scratch
 - Creating Maven Project
 - Update pom.xml with dependencies
 - Creating page objects and Object repository
 - Creating utility files
 - Setting up configuration files
 - Creating automation test scripts
 - Creating data driven test scripts
 - Generating extent reports
 - Generating logs using log4j
 - Execute test scripts
 - Emailing test reports
 - Execute test scripts through Maven
 - Execute test scripts using bat file

Module 10: Continuous Integration (CI) - Maven, Jenkins & GIT

- What is Continuous Integration?
- Continuous Integration Tools
- Download and install Jenkins
- Maven integration with Jenkins
- Run selenium test scripts through Jenkins
- Downloading and Installing GIT
- Installing GIT and GITHUB plug-ins for Jenkins
- Git commands
- Uploading project to Github

Module-11: Cucumber BDD Framework

- Cucumber Introduction
- TDD (Test Driven Development) Vs BDD (Behavior Driven Development)
- Set Up Cucumber with Eclipse
- Install Cucumber Eclipse Plugin
- Download Cucumber JVM for Eclipse
- Cucumber Selenium Java Test
- Cucumber Environment
- Setting Up Cucumber with Selenium

- Cucumber Gherkins languages
- Cucumber – Features file
- Cucumber – Scenarios
- Cucumber – Annotations
- Cucumber - Scenario Outline
- Cucumber – Tag
- Cucumber - Data Tables
- Parameterization in Cucumber
- Cucumber - Comments
- Cucumber - Hooks
- Cucumber - Command Line Options
- Cucumber - JUnit Runner
- Cucumber – Reports generation both Html report and Json report

Module-12: Appium for Mobile App Testing

- Introduction to Mobile Testing & Appium tool
- Emulators, Simulators and Real devices
- Environment setup
- Working with Android Studio and Appium Server
- Running the scripts on Emulators
- Running the scripts on Real devices
- Understanding the API Levels and Appium Concepts
- Client/Server Architecture, Session, Desired Capabilities, Appium Server and Appium Clients
- Installing JDK, Maven and Android SDK
- Taking a look at available Android Virtual Devices
- Downloading Appium for Windows
- Downloading Appium jars and other dependencies
- Setting up Android environment variables
- Starting Appium Server
- Connecting real device to USB and running ADB commands
- Starting the Appium server programmatically Windows OS
- Starting the Appium server through code / programmatically on Windows
- How to Kill all nodes
- UI Automator Viewer

- Opening Automator Viewer through SDK tools kit
- Understanding different locator strategies
- Taking the Device screenshot
- Appium Inspector
- Record and Play using Appium Inspector
- Locating elements with the help of Appium Inspector
- Object Repository
- Locating Elements on the App
- Testing Android Native, Hybrid & Web Apps on Windows
- Native App Test – Making a call with Internal Dialer Application
- Native App Test – Adding a new contact to Phonebook
- Native App Test – Sending SMS from a real device
- Native App Test – Automating a standard App
- Touch Actions and Events – Handling user gestures
- WebApp Test – Automating Web based testing on Chrome Browser
- Hybrid App Test – Automating the mobile messaging app

Module-13: API/Webservices Testing- Postman & Rest Assured

- Basics of API & Web services Testing
- What is Client and Server?
- Client Server architecture
- Presentation, Business & Database Layer
- What is Request & Response
- What is API Testing
- API Testing VS Unit Testing
- What is Web Service
- Difference between API & Web service
- Components of Web services (WSDL & UDDI)
- Types of Web services
- SOAP VS Rest services
- URI, URL & URN
- API/Web service testing challenges
- Web Service API testing tools
- Web services API Testing process
- What to test in Webservices /API Testing?
- HTTP Methods (GET/POST/PUT/DELETE)
- Status Codes

- Examples of Web service API's

Postman

- Download and Install Postman
- Creating Requests & Analyzing Response
- Request Parameters in Postman
- Postman Collections
- End to End test case (GET, POST, PUT, DELETE)
- Test and Collection Runner in Postman
- Workflows in Postman
- Variables in POSTMAN
- Environment/Global variables
- How to share collections in workspaces
- Data Driven testing using Json & CSV files
- Run Postman tests in command Line
- Generating Reports in Postman
- How to handle SOAP Web service Requests in Postman

Swagger UI overview

Rest Assured

- Introduction to HTTP methods
- REST Assured Setting up environment (Eclipse, Maven & TestNG)
- Send GET request using REST-Assured
- Send POST request using REST-Assured
- Send PUT request using REST-Assured
- Send DELETE request using REST-Assured
- REST-Assured Extracting values from the response.
- Validating Response codes and status line
- JSON Schema Validation
- Validating Headers
- Rest Assured End-to-End Test Cases
- Data Driven test cases
- How to run API tests using Maven CLI
- Generating reports
- Run Rest Assured tests in Jenkins

Module-14: JavaScript End to End Testing Framework Cypress

Java Scripting

- What is JavaScript? overview on JS
- Environment setup

- Variables
- Arrays
- Conditional Statements:
- IF, Else, Else IF (Example)
- Loops: For, While and Do While LOOP
- Functions

Overview on JavaScript Frameworks

Cypress.io Vs Selenium

Functional UI testing with Cypress

- Introduction to Cypress
- Installing Cypress
- Writing and Organizing Tests
- Interacting with Elements
- Variables and Aliases
- Conditional Testing
- Data driven testing
- The Test Runner
- Dashboard Service
- Debugging
- Screenshots and Videos
- Launching Browsers
- Reporters
- Assertions