

Selenium WebDriver

DAY21 - Introduction to WebDriver & Setting up WebDriver Environment

DAY22 - Selenium Locators - ID, Name, LinkText, Partial LinkText, TagName

DAY23 - Selenium Locators - CSS Selector

DAY24 - Selenium Locators – Xpath

DAY25 - Selenium Locators - XPath Axes

DAY26 - WebDriver Methods - Get methods, Conditional methods & Browser methods

DAY27 - WebDriver Methods - Different types of Waits

DAY28 - WebDriver Methods - Browser Methods and Navigation methods

DAY29- Handle Checkboxes & Radio Buttons

DAY30 - Handle Different Types of Alerts & Frames

DAY31- Handle Different types of drop downs

DAY32 - Handle Static Web Table

DAY33 - Handle Dynamic & Pagination Web Table

DAY34 - Handle Date Pickers

DAY35 - Handle Mouse Events

DAY36 - Handle Keyboard Events, Tabs & Windows

DAY37 - Handle File Upload, Scrolling Page, JavascriptExecutor

DAY38 - Capture screenshots, HandleSSL,Headless browser, Blocking Ads,ChromeExtensions

DAY39 - Handle Broken Links, Handle SVG Elements & Shadow DOM Elements

[illegible]

Apache POI – Data Driven testing using MS excel

DAY40 - Reading Excel & Writing Data into Excel

DAY41 - Data Driven Testing using Microsoft Excel

[illegible]

TestNG Framework for Selenium

Day42 - Introduction to TestNG

Day43 - TestNG Annotations & Types of Assertions

Day44 - Dependency Methods & Grouping

Day45 - Data Providers and Parallel Testing

Day46 - TestNG Listeners & Extent Reports

[illegible]

Page Object Model

Day47 - Page Object Model Pattern - Page Factory,

[illegible]

Hybrid Driven Framework Development with Project (Ecommerce

Day48 - Automation Framework -Session1

Day49 - Automation Framework -Session2

Day50 - Automation Framework -Session3

Day51 - Automation Framework -Session4

Day52 - Automation Framework -Session5

Day53 - Automation Framework -Session6 (Run Tests on Selenium Grid)

Day54 - Automation Framework -Session7(Docker Integration with Selenium Grid)

[illegible]

Maven, GIT, GitHub & Jenkins

Day55 - Continuous Integration Part -1

Day56 - Continuous Integration Part -2 --- git project import

[illegible]

Cucumber BDD Framework

Day57 - Cucumber and BDD Concepts

Day58 - Cucumber Framework -Part1

Day59 - Cucumber Framework - Part2 – project import

[illegible]

Bonus Topics

Day60 - Git Options in Eclipse & IntelliJ IDE

Day61 - How to use ChatGPT in Software testing and automation

[illegible]

SelectorsHub

Day62 – Interview Preparation

Day63 –

Docker Introduction

Docker Installation

Docker Terminology

Docker Commands

How To Run Selenium Tests on Docker Containers

Selenium Grid Setup with Docker

How to Invoke Docker Grid using bat files

Docker with Selenium Grid Integration with Jenkins

Module-1: Overview on Automation & Selenium

- Introduction to Automation
- Challenges in Manual Testing
- Overcoming Challenges with Automation
- Importance of Programming in Automation
- Introduction to Selenium
- Selenium Components
- Advantages and Disadvantages of Selenium WebDriver

Module-2: Java Programming for Selenium

- **Java and Eclipse Environment Setup**
 - Setting up Java Development Kit (JDK)
 - Installing Eclipse IDE
 - Configuring Java and Eclipse Environment
- **How to Write Basic Java Program?**
 - Creating a Java Class
 - Main Method in Java
 - Writing the First Java Program
 - Compiling and Running Java Programs
- **Understanding Console Output**
 - Using System.out.println() for Output
 - Printing Variables and Text to Console
- **Java Variables and Data Types**
 - Declaring Variables in Java
 - Primitive Data Types (int, float, double, char, boolean)
 - Variable Naming Conventions
- **Java Operators**
 - Arithmetic Operators
 - Relational Operators
 - Logical Operators
 - Assignment Operators
 - Increment and Decrement Operators
 - Ternary Operators
- **Conditional Statements**
 - If Statement
 - If-Else Statement

- Switch-Case Statement
- **Loops**
 - While Loop
 - Do-While Loop
 - For Loop
- **Jumping Statements**
 - Using break Statement
 - Using continue Statement
- **Arrays - Single & Two Dimensional**
 - Declaring and Initializing Arrays
 - Accessing Array Elements
 - Single-Dimensional Arrays
 - Two-Dimensional Arrays
 - Object Type Array
 - Coding Exercises on Arrays
- **String Class**
 - Introduction to the String Class
 - Creating Strings
 - String Concatenation
 - String Length
 - String Comparison
- **String Methods**
 - Common String Methods
 - Coding Exercises on Strings

Module 3: Java OOPS Concepts

- **Classes and Objects**
 - Introduction to Classes and Objects
 - Creating Classes
 - Declaring Object Instances
 - Class Members: Fields and Methods
- **Java Methods & Constructors**
 - Defining Methods
 - Method Signatures
 - Return Types
- **Passing Parameters to Methods**
 - Passing Parameters
 - Call by Value
 - Call by Reference (Reference Passing)
- **Java Constructor**
 - Constructor Basics
 - Parameterized Constructors
 - Default Constructors
- **Polymorphism**
 - Introduction to Polymorphism

- **Overloading Methods**
 - Method Overloading
 - Overloading with Different Parameter Types
 - Overloading with Different Number of Parameters
- **Overloading Constructors**
 - Constructor Overloading
 - Creating Constructors with Different Parameters
- **Encapsulation**
 - Encapsulation Overview
- **Setters and Getters**
 - Purpose of Setters and Getters
 - Encapsulation with Setters and Getters
- **this Keyword**
 - Understanding the this Keyword
 - Using this to Differentiate Instance and Local Variables
- **Static Variables and Methods**
 - Static Variables
 - Static Methods
 - Accessing Static Members
- **Inheritance**
 - Basics of Inheritance
 - Extending Classes
 - Superclass and Subclass
- **Types of Inheritance**
 - Single Inheritance
 - Multiple Inheritance
 - Multilevel Inheritance
 - Hierarchical Inheritance
- **Method Overriding**
 - Overriding Methods in Subclasses
 - Use of @Override Annotation
- **super Keyword**
 - Using super to Access Superclass Members
 - Invoking Superclass Methods
- **final Keyword**
 - Final Variables
 - Final Methods
 - Final Classes
- **Abstract Class**
 - Abstract Class Overview
 - Abstract Methods
 - Creating Abstract Classes
- **Java Interface**
 - Introduction to Interfaces
 - Implementing Interfaces
 - Multiple Interface Implementation
- **Java Packages**
 - Organizing Code with Packages

- Creating and Using Packages
- **Access Modifiers**
 - Public, Private, Protected, Default Access Modifiers
 - Access Modifier Scope
- **Wrapper Classes**
 - Primitive Data Types vs Wrapper Classes
 - Autoboxing and Unboxing
- **Data Conversion Functions**
 - Converting Data Types
 - Casting - Up Casting and Down casting
- **Reading Input from Keyboard**
 - Using Scanner Class for Input
- **Exception Handling**
 - Introduction to Exception Handling
 - Try, Catch, Finally Blocks
 - Types of Exceptions
- **Java Collections**
 - Overview of Java Collections Framework
 - Lists, Sets, Maps
 - Iterators
 - Collections Methods

Module-4: Selenium WebDriver (Latest Version)

- **Introduction to WebDriver and Architecture**
 - Overview of Selenium WebDriver
 - Architecture of Selenium WebDriver
- **Environment Setup & WebDriver Configuration**
 - Setting up WebDriver in Eclipse
 - Configuring WebDriver for Different Browsers
- **How to Create Automated Test Case in WebDriver?**
 - Setting Up a Basic Test Case Structure
 - Writing and Running Simple WebDriver Test Cases
- **Execute Test Case on Multiple Browsers**
 - Cross-Browser Testing
 - Running Tests on Different Browsers
- **Understanding HTML Tags and Attributes**
 - HTML Basics for Selenium Testers
 - Locating Elements Using HTML Tags and Attributes
- **Basic Locators**
 - Name Locator
 - ID Locator
 - LinkText Locator
 - PartialLinkText Locator

- **Advanced Locators**
 - CSS Locators
 - XPath Locators
 - XPath Axes
- **Relative Locators**
 - .above, .below, .toLeftOf, .toRightOf, .near
- **How to Use SelectorsHub & Features of Selectors Hub**
 - Overview of SelectorsHub
 - Features and Benefits
- **WebDriver Methods**
 - Opening Page URL
 - Verifying Page Title
 - Getting Page Source
 - Getting Page URL
 - Conditional Methods
 - Visibility of WebElements
 - Enable/Disable Status of Web Elements
- **Waits**
 - Implicit Waits
 - Explicit Waits
 - Fluent Waits
 - Page Load Time
- **Navigation Methods**
 - Navigating Back & Forward
 - Refreshing Page
- **Browser Commands**
 - close() and quit()
 - getWindowHandle() and getWindowHandles()
- **Working with Web Elements**
 - Handling Input Boxes
 - Handling Checkboxes & Radio Buttons
 - Handling Links
- **Handling Alerts/Popups**
 - Handling Alert Boxes
 - Handling Confirmation Boxes
 - Handling Prompt Boxes
- **Handling Iframes & Inner Iframes**
 - Switching to Iframes
 - Working with Nested Iframes
- **Handling Drop Downs**
 - Select type Drop down
 - Bootstrap Drop Downs
 - Multi Select Drop Down
 - Auto Suggest Drop Down
- **Working with Browser Windows**
 - Switching Between Browser Windows
 - Handling Tabs and Windows
- **Handling Static & Dynamic Web Tables**

- Strategies for Handling Tables
 - Handling Static Tables
 - Handling Dynamic Tables
 - Pagination Tables
- **Handling Date Pickers**
 - Strategies for Handling Date Pickers
- **Handling Mouse Actions**
 - Using Actions Class
 - Right Click, Double Click, Mouse Over
 - Drag and Drop Actions
- **Handling Keyboard Actions and Shortcuts**
 - Using Actions Class for Keyboard Actions
- **Working with JavascriptExecutor**
 - Executing JavaScript Code in Selenium
- **Scrolling Web Page**
 - Scrolling Up and Down the Page
- **Zoom In & Zoom Out, Maximize, Minimize Web Page**
 - Managing Browser Window Size
- **Opening Links in New Tab**
 - Handling New Tabs
- **Working with Tabs and Windows**
 - Switching Between Tabs and Windows
- **Capturing Screenshots**
 - Taking Screenshots in Selenium
- **Working with Cookies**
 - Managing Cookies in Selenium
- **Headless Browser Testing**
 - Overview of Headless Testing
 - Advantages and Disadvantages
- **SSL Certificate Handling**
 - Handling SSL Certificates
- **Browser Settings with ChromeOptions**
 - Customizing Chrome Browser Settings
- **Enable Extensions on Browser at Runtime**
 - Managing Browser Extensions
- **Blocking Ads on WebPage at Runtime**
 - Blocking Ads Using WebDriver
- **Run Tests in Incognito Mode**
 - Running Tests in Incognito/Private Mode
- **How to Handle Upload Files (Single and Multiple)**
 - Uploading Files in Selenium
- **Handling Broken Links**
 - What is Broken link?
 - How to find Broken links in web page
- **Handling shadowDOM, SVG Elements**
 - Strategies for Handling shadowDOM and SVG Elements
- **Selenium WebDriver Exceptions**
- **Working with Excel Files Using Apache POI API**

- Reading and Writing Excel Files
- **Data-Driven Testing**
 - Strategies for Data-Driven Testing
 - Implementing Data-Driven Tests in Selenium
- **Reading Properties File**

Module-5: TestNG

- **Introduction to TestNG and Advantages**
 - Overview of TestNG
 - Advantages of TestNG over JUnit
 - TestNG Features
- **Configuring TestNG in Project**
 - Setting up TestNG in Eclipse or IntelliJ
 - Configuring TestNG Dependencies in the Project
- **Writing TestNG Test Cases**
 - Basics of Writing TestNG Test Cases
 - Annotations in TestNG
- **Prioritizing Tests**
 - Setting Test Case Priority
 - Running Tests in Priority Order
- **Understanding Testng.xml**
 - Structure of TestNG XML File
 - Configuring Test Suites and Test Cases
- **TestNG Report**
 - TestNG HTML Reports
 - Viewing and Analyzing TestNG Reports
- **TestNG Annotations**
 - @BeforeSuite, @AfterSuite
 - @BeforeTest, @AfterTest
 - @BeforeClass, @AfterClass
 - @BeforeMethod, @AfterMethod
 - @Test Annotation
- **Types of Assertions (Hard and Soft Assertions)**
 - Using Hard Assertions in TestNG
 - Using Soft Assertions in TestNG
- **dependsOnMethods**
 - Defining Dependencies Between Test Methods
 - Ensuring Test Methods Run in a Specific Order
- **Skipping and Ignoring Tests**
 - Skipping Tests in TestNG
 - Ignoring Tests in TestNG
- **Grouping Tests**
 - Grouping Tests in TestNG

- Running Tests by Groups
- **Parameterizing Tests (XML File, DataProviders)**
 - Parameterizing Tests Using TestNG XML File
 - Parameterizing Tests Using DataProviders
- **@DataProvider**
 - Understanding @DataProvider Annotation
 - Providing Test Data to Test Methods
- **Invocation Count**
 - Repeating Test Execution with Invocation Count
- **Parallel Testing**
 - Running Tests in Parallel
 - Configuring Parallel Execution in TestNG
- **TestNG Listeners**
 - Overview of TestNG Listeners
 - Implementing Listeners in TestNG
 - Types of TestNG Listeners (e.g., ITestListener, IAnnotationTransformer)

Module-6: Extent Reports, Log4j2 & Page Object Model Pattern

- Generating Extent Reports
- Log4j2 for Logging
- Page Object Model (POM) Pattern
- Creating Page Object Classes without Page Factory
- Creating Page Object Classes using Page Factory
- @FindBy annotation

Module-7: Selenium Grid

- Introduction to Selenium Grid
- Setting Up Grid Environment (Stand Alone, Hub with Nodes)
- Running Selenium Tests on Grid

Module-8: Docker

- What is Docker
- What is Docker Image
- What is Docker Container

- How to create account on Docker Hub
- Download Images from Docker Hub
- Docker UI Application
- Selenium Grid Configuration on Docker Hub
- Run your test cases on Dockerized Selenium Grid
- Setup Infrastructure using Docker Compose yml
- Scale up the browsers at the run time
- Shut down the Docker Infrastructure

Module-9: Introduction to Automation Frameworks

- Introduction to Automation Frameworks
- Types of Frameworks
- Phases in Designing Framework
- Choosing Test Cases for Automation

Module-10: Hybrid Framework with POM (Page Object Model) – Project

- Creating Maven Project and update pom.xml with dependencies
- Creating Folder Structure
- Creating Page Objects and Object Repository
- Creating Utility files
- Generating Extent Reports
- Generating Logs using Log4j2
- Setting Up Configuration Files
- Creating Automation Test Cases
- Creating Data-Driven Test Cases
- Creating Database Test Cases
- Executing Test Scripts using TestNG XML File
- Emailing Test Reports
- Executing Test Scripts through Command Line
- Executing Test Scripts using Batch File
- Executing Test Scripts on Selenium Grid
- Executing Script on Docker Container

Module-11: Cucumber BDD Framework

- Introduction to BDD and Cucumber
- Gherkin Language Keywords
- Feature Files

- Step Definitions
- TestRunner Class
- Data Tables and Scenario Outline
- Tags, Background
- Cucumber Hooks
- Cucumber Reports
- Cucumber Extent Reports

Module-12: CI & CD Concepts(Git, Github & Jenkins)

- Version Control System Overview
- Git and GitHub
- Git Commands
- Continuous Integration (CI) and Continuous Delivery (CD)
- Jenkins Overview
- Maven Integration with Jenkins
- Running Selenium Tests through Jenkins

Who this course is for:

- Any Fresh Graduates, Students, Working professionals, Manual testers & who wish to learn this course in order to transform as an Automation Test Engineer can go for this course.

Show less

Instructor

Pavan Kumar

Tech Educator | Trainer | YouTuber

As a dedicated Tech Educator and Trainer, I leverage my extensive experience in Software Development and Quality Assurance to empower professionals with the skills needed to thrive in today's dynamic tech landscape.

Key Highlights:

- 18+ years of experience in various domains within the IT industry and 15+ years as an Instructor & Corporate Trainer.

- Designed and delivered 17+ impactful courses on Udemy, reaching over 200,000+ students worldwide.
- Content creator on YouTube channel "SDET-QA," with over 715K+ subscribers, where many have benefited from learning various tools & technologies related to software testing & automation.
- Specialized in test automation frameworks and web automation using Selenium, Cypress, and Playwright.
- Expertise in programming languages including Java, Python, JavaScript, and TypeScript.
- Proficient in web testing, API testing, mobile app testing, performance testing, and CI/CD practices.
- Emphasis on JavaScript-based frameworks and the latest test automation tools for efficient, reliable, and scalable solutions.

Beyond my teaching roles, I actively engage with the tech community through my blog and YouTube content, where I share valuable insights, practical tips, and the latest trends in software testing and automation. My mission is to deepen understanding and drive innovation, helping professionals excel in their careers.

Show less

Requirements

- Software Testing Concepts
- Understanding of Computers and Web Sites
- No prior knowledge of Java, everything will be covered in this course