



TRAINING PROGRAMS

Software Testing & Automation — Complete Course Guide

Four complete, hands-on training programs in one guide — Playwright, Selenium, API Testing, and Manual Testing. Jump to any program using the contents page or the bookmarks panel in your PDF viewer.

100% Hands-On	Every class across every program is a practical, code-along session.
Repeat Classes Free	Didn't get a topic? We repeat it until you do — at no extra cost.
Flexible Schedule	Weekday & weekend batches, plus individual (1-on-1) training.
Support Beyond Training	Interview support and 2 months of free job support once placed.

Software Testing & Automation Coding Knowledge
connect@stackweb.in | 8754735073 | Kodungaiyur, Chennai

Table of Contents

1. Playwright Automation Testing →

Java / TypeScript basics to a full Playwright automation framework.

2. Selenium Automation Testing →

Java / Python basics to a full Selenium automation framework.

3. API Testing →

Postman fundamentals to automated API testing with RestAssured.

4. Manual Testing →

Testing fundamentals to a complete real-time manual testing project.

Tip: most PDF readers show a bookmarks/outline panel on the side — you can also navigate using that.

PLAYWRIGHT AUTOMATION TESTING

Complete Hands-On Training Program

From Java / TypeScript basics to building a complete automation framework — 100% practical, hands-on classes.

This program takes you from programming basics to building a complete, real-world Playwright automation framework. Every session is hands-on — you code along in every class, using real websites and real test scenarios.

Phase	Focus
Phase 1	Programming Foundation — Java Basics OR TypeScript Basics (choose one track)
Phase 2	Playwright From Scratch — install Node.js, first project, locators & actions
Phase 3	Intermediate Playwright — data-driven tests, API testing, debugging tools
Phase 4	Framework Development — Page Object Model, reporting, CI/CD, capstone project

[↑ Back to Table of Contents](#)

PHASE 1

Programming Foundation — Choose Your Track

Pick the track that matches your Playwright language of choice. Both are taught the same way — small hands-on programs every session.

Track A — Java Basics

#	Topic	Hands-On Activity
1	Setup & First Program	Install JDK + IntelliJ, write and run Hello World
2	Variables, Data Types, Operators	Small programs using int, String, boolean, arithmetic
3	Conditionals & Loops	if/else, switch, for/while — practice problems
4	Arrays & Strings	Store and manipulate test data using arrays and String methods
5	OOP Basics	Create classes, objects, constructors for a sample test entity
6	Methods & Access Modifiers	Write reusable methods, public/private practice
7	Collections (List, Map, Set)	Store and loop through test data sets
8	Exception Handling	try/catch practice with common runtime errors

Track B — TypeScript Basics

#	Topic	Hands-On Activity
1	Setup & First Script	Install Node.js + TypeScript, run your first .ts file
2	Variables & Types	string, number, boolean, arrays, union types — hands-on
3	Functions & Arrow Functions	Write and call functions, practice arrow syntax
4	Objects & Interfaces	Model test data using interfaces and typed objects
5	Classes & Constructors	Build simple classes used later in framework code
6	Async / Await & Promises	Practice asynchronous code — core to Playwright
7	Modules (import/export)	Split code across files and import between them
8	Error Handling	try/catch practice, handling failures gracefully

[↑ Back to Table of Contents](#)

PHASE 2**Playwright From Scratch**

#	Topic	Hands-On Activity
1	Installing Node.js & npm	Set up Node, verify install, npm basics
2	Creating Your First Playwright Project	Run npx create playwright, explore the generated project
3	Project Structure & Config	Walk through playwright.config.ts and folder structure
4	Writing & Running Your First Test	Write a real test script and execute it headed & headless
5	Locators	Practice getByRole, getByText, getByLabel, CSS & XPath on live pages
6	Core Actions	click, fill, type, hover, select, check/uncheck — live practice
7	Assertions	Use expect() to validate text, visibility, counts, URLs
8	Dropdowns, Alerts & iFrames	Handle native dropdowns, JS alerts, and embedded frames
9	Multiple Tabs & Windows	Automate flows that open new tabs/popups
10	File Upload & Download	Automate uploading and downloading files in a test
11	Dynamic Elements & Waits	Handle elements that load asynchronously

[↑ Back to Table of Contents](#)

PHASE 3 Intermediate Playwright

#	Topic	Hands-On Activity
1	Test Hooks & Structure	Organize tests using describe, beforeEach, afterEach
2	Data-Driven Testing	Run the same test with data from JSON / CSV / Excel
3	Environment & Config Management	Manage URLs/credentials across environments
4	API Testing Basics	Send GET/POST requests using Playwright's request context
5	Combining UI + API Tests	Set up test data via API, verify it through the UI
6	Visual Testing & Screenshots	Capture and compare screenshots in a test run
7	Debugging Tools	Use Trace Viewer, codegen, and headed mode to debug failures

[↑ Back to Table of Contents](#)

PHASE 4**Framework Development (Capstone)**

#	Topic	Hands-On Activity
1	Page Object Model (POM)	Refactor existing tests into a POM structure
2	Reusable Utilities & Helpers	Build shared helper functions used across tests
3	Framework Folder Structure	Set up a real-world framework: pages, tests, utils, config
4	Reporting	Add HTML / Allure reporting to your test run
5	Git & GitHub Basics	Push your framework to GitHub, commit workflow practice
6	CI/CD Basics	Run your test suite automatically using GitHub Actions
7	Capstone Project	Build and run a complete test suite on a real application, end to end

[↑ Back to Table of Contents](#)

Our Promise

- We support you till you understand the daily job routine, not just the syllabus.
- Every class is hands-on — you leave each session having written working code.
- Didn't get a topic the first time? We repeat the class, free of cost.
- Weekday and weekend batches, plus individual (1-on-1) training, based on your availability.

[↑ Back to Table of Contents](#)

SELENIUM AUTOMATION TESTING

Complete Hands-On Training Program

From Java / Python basics to building a complete Selenium automation framework — 100% practical, hands-on classes.

This program takes you from programming basics to building a complete, real-world Selenium automation framework. Every session is hands-on — you code along in every class, using real websites and real test scenarios.

Phase	Focus
Phase 1	Programming Foundation — Java Basics OR Python Basics (choose one track)
Phase 2	Selenium From Scratch — WebDriver setup, locators, actions, waits
Phase 3	Intermediate Selenium — TestNG/PyTest, data-driven tests, cross-browser testing
Phase 4	Framework Development — Page Object Model, reporting, CI/CD, capstone project

[↑ Back to Table of Contents](#)

PHASE 1

Programming Foundation — Choose Your Track

Pick the track that matches your Selenium language of choice. Both are taught the same way — small hands-on programs every session.

Track A — Java Basics

#	Topic	Hands-On Activity
1	Setup & First Program	Install JDK + IntelliJ, write and run Hello World
2	Variables, Data Types, Operators	Small programs using int, String, boolean, arithmetic
3	Conditionals & Loops	if/else, switch, for/while — practice problems
4	Arrays & Strings	Store and manipulate test data using arrays and String methods
5	OOP Basics	Create classes, objects, constructors for a sample test entity
6	Methods & Access Modifiers	Write reusable methods, public/private practice
7	Collections (List, Map, Set)	Store and loop through test data sets
8	Exception Handling	try/catch practice with common runtime errors

Track B — Python Basics

#	Topic	Hands-On Activity
1	Setup & First Script	Install Python + VS Code, run your first .py file
2	Variables & Data Types	Practice with strings, numbers, booleans, lists
3	Conditionals & Loops	if/elif/else, for/while — practice problems
4	Lists, Tuples & Dictionaries	Store and manipulate test data collections
5	Functions	Write and call reusable functions
6	OOP Basics	Create classes, objects, constructors for a sample test entity
7	Exception Handling	try/except practice with common runtime errors
8	File Handling	Read test data from a file into your script

[↑ Back to Table of Contents](#)

PHASE 2 Selenium From Scratch

#	Topic	Hands-On Activity
1	Installing Selenium WebDriver	Set up Selenium + browser drivers (Chrome/Firefox)
2	First Project Setup	Create a project in IntelliJ/PyCharm, run your first script
3	WebDriver Basics	Launch browser, navigate, and close a session — hands-on
4	Locators	Practice id, name, class, CSS selectors & XPath on live pages
5	Core Actions	click, sendKeys, clear, select dropdown — live practice
6	Waits	Implicit & explicit waits practiced on slow-loading elements
7	Alerts, Frames & Multiple Windows	Handle JS alerts, iframes, and new browser windows
8	Checkboxes, Radio Buttons & Dropdowns	Automate common form elements
9	File Upload & Download	Automate uploading and downloading files in a test
10	Dynamic Elements	Handle elements that load asynchronously

[↑ Back to Table of Contents](#)

PHASE 3**Intermediate Selenium**

#	Topic	Hands-On Activity
1	TestNG / PyTest Basics	Annotations, assertions, and test grouping
2	Data-Driven Testing	Run the same test with data from Excel/CSV/JSON
3	Cross-Browser Testing	Run the same test across Chrome, Firefox, Edge
4	Screenshots on Failure	Auto-capture a screenshot when a test fails
5	Logging	Add logs to track test execution flow
6	Synchronization Strategies	Advanced waits for flaky, dynamic pages
7	Debugging Failed Tests	Step through and fix a broken test live

[↑ Back to Table of Contents](#)

PHASE 4**Framework Development (Capstone)**

#	Topic	Hands-On Activity
1	Page Object Model (POM)	Refactor existing tests into a POM structure
2	Page Factory / Reusable Utilities	Build shared helper functions used across tests
3	Framework Folder Structure	Set up a real-world framework: pages, tests, utils, config
4	Reporting	Add Extent Reports / Allure reporting to your test run
5	Git & GitHub Basics	Push your framework to GitHub, commit workflow practice
6	CI/CD Basics	Run your test suite automatically using Jenkins/GitHub Actions
7	Capstone Project	Build and run a complete test suite on a real application, end to end

[↑ Back to Table of Contents](#)

Our Promise

- We support you till you understand the daily job routine, not just the syllabus.
- Every class is hands-on — you leave each session having written working code.
- Didn't get a topic the first time? We repeat the class, free of cost.
- Weekday and weekend batches, plus individual (1-on-1) training, based on your availability.

[↑ Back to Table of Contents](#)

API TESTING

Complete Hands-On Training Program

From API fundamentals in Postman to a fully automated RestAssured framework — 100% practical, hands-on classes.

This program takes you from API fundamentals to building a complete, automated API testing framework. You start hands-on in Postman, then move into code-driven automation with RestAssured.

Phase	Focus
Phase 1	API Fundamentals — hands-on with Postman: methods, requests, responses, scripts
Phase 2	API Automation with RestAssured (Java) — from setup to full CRUD test coverage
Phase 3	Intermediate API Testing — data-driven tests, schema validation, auth
Phase 4	Framework Development — reusable structure, reporting, CI/CD, capstone project

[↑ Back to Table of Contents](#)

PHASE 1**API Fundamentals (Postman)**

#	Topic	Hands-On Activity
1	What Is an API?	Explore real public APIs and inspect requests/responses
2	HTTP Methods	Practice GET, POST, PUT, DELETE in Postman
3	Request & Response Structure	Inspect headers, params, and body on live APIs
4	Status Codes & Validations	Validate response codes and body values in Postman
5	Postman Collections & Environments	Organize requests, switch between environments
6	Pre-request & Test Scripts	Write basic JavaScript checks inside Postman
7	Postman Runner & Newman	Run a full collection from the command line

[↑ Back to Table of Contents](#)

PHASE 2**API Automation with RestAssured (Java)**

#	Topic	Hands-On Activity
1	RestAssured Project Setup	Create a Maven project, add RestAssured dependencies
2	First Automated GET Request	Write and run your first RestAssured test
3	Validating Responses	Assert status code, response body, and headers in code
4	POST / PUT / DELETE Requests	Automate full CRUD operations against a live API
5	Handling JSON	Extract values using JsonPath and map responses to POJOs
6	Authentication	Automate requests using Basic Auth, Bearer Token, and API Keys
7	Chaining Requests	Use data from one API response in a dependent API call

[↑ Back to Table of Contents](#)

PHASE 3**Intermediate API Testing**

#	Topic	Hands-On Activity
1	Data-Driven API Testing	Run the same API test with data from JSON/CSV/Excel
2	Schema Validation	Validate response structure using JSON Schema
3	Query & Path Parameters	Automate requests using dynamic parameters
4	Mocking APIs (Basics)	Simulate an API response for isolated testing
5	Combining API + UI Tests	Set up test data via API, verify it through the UI
6	Logging & Debugging	Inspect request/response logs to debug failing tests

[↑ Back to Table of Contents](#)

PHASE 4**Framework Development (Capstone)**

#	Topic	Hands-On Activity
1	Reusable Framework Structure	Organize requests, payloads, and tests into a clean structure
2	Config & Environment Management	Manage base URLs and credentials across environments
3	Reporting	Add Extent/Allure reporting to your API test run
4	Git & GitHub Basics	Push your framework to GitHub, commit workflow practice
5	CI/CD Basics	Run your API test suite automatically using GitHub Actions/Jenkins
6	Capstone Project	Build and run a complete API test suite for a real service, end to end

[↑ Back to Table of Contents](#)

Our Promise

- We support you till you understand the daily job routine, not just the syllabus.
- Every class is hands-on — you leave each session having tested a real API.
- Didn't get a topic the first time? We repeat the class, free of cost.
- Weekday and weekend batches, plus individual (1-on-1) training, based on your availability.

[↑ Back to Table of Contents](#)

MANUAL TESTING

Complete Hands-On Training Program

From testing fundamentals to a full real-time project — 100% practical, hands-on classes.

This program takes you from testing fundamentals to running a complete real-time testing project. Every session is hands-on — you analyze requirements, write test cases, execute them, and log defects on a real sample application.

Phase	Focus
Phase 1	Testing Fundamentals — SDLC/STLC, testing types & levels, hands-on on a sample app
Phase 2	Test Case Design — requirement analysis, test case writing, design techniques
Phase 3	Test Execution & Defect Management — execution, bug reporting, defect life cycle
Phase 4	Tools & Real-Time Project — JIRA, test management tools, full project cycle

[↑ Back to Table of Contents](#)

PHASE 1

Testing Fundamentals

#	Topic	Hands-On Activity
1	What Is Software Testing?	Explore a sample app and identify what could go wrong
2	SDLC & STLC	Map real project phases onto a sample project timeline
3	Types of Testing	Identify functional vs non-functional tests on a sample app
4	Levels of Testing	Walk through unit, integration, system, and UAT with examples
5	Verification vs Validation	Practice distinguishing the two with real examples

[↑ Back to Table of Contents](#)

PHASE 2**Test Case Design**

#	Topic	Hands-On Activity
1	Requirement Analysis	Read a sample user story / SRS and extract test conditions
2	Test Scenario Writing	Write test scenarios for a real sample application
3	Test Case Writing	Write detailed, step-by-step test cases with expected results
4	Design Techniques	Apply BVA, Equivalence Partitioning, Decision Tables — hands-on
5	Test Data Preparation	Prepare valid/invalid data sets for your test cases
6	Traceability Matrix	Build a requirement-to-test-case traceability matrix

[↑ Back to Table of Contents](#)

PHASE 3**Test Execution & Defect Management**

#	Topic	Hands-On Activity
1	Test Case Execution	Execute your written test cases on a live sample app
2	Logging Defects	Write clear, reproducible bug reports
3	Defect Life Cycle	Track a defect through New → Fixed → Retest → Closed
4	Severity & Priority	Practice classifying real defect examples
5	Regression & Retesting	Re-execute impacted test cases after a fix

[↑ Back to Table of Contents](#)

PHASE 4**Tools & Real-Time Project (Capstone)**

#	Topic	Hands-On Activity
1	JIRA	Log, assign, and track issues hands-on in JIRA
2	Test Management Tools	Write and organize test suites in TestRail/Zephyr
3	Automation Awareness	See how manual test cases feed into automation scripts
4	Agile / Scrum Basics	Walk through sprint planning, stand-ups, and retros
5	Capstone Project	Full cycle: analyze requirements, write cases, execute, log defects, report

[↑ Back to Table of Contents](#)

Our Promise

- We support you till you understand the daily job routine, not just the syllabus.
- Every class is hands-on — you leave each session having worked on a real sample project.
- Didn't get a topic the first time? We repeat the class, free of cost.
- Weekday and weekend batches, plus individual (1-on-1) training, based on your availability.

[↑ Back to Table of Contents](#)

S.T.A.C.K — Software Testing & Automation Coding Knowledge

connect@stackweb.in | 8754735073 | Kodungaiyur, Chennai

TRAIN | SUPPORT | SUCCEED