

LIVE COURSE

Software Testing & Automation

Learn manual and automation testing with Selenium, API testing, Postman, Java and CI/CD in live mentor-led sessions. Build a job-ready portfolio with real projects and interview preparation.

Industry-Led Course

Batch details updating soon

Technical



DURATION
4 Months



LEVEL
Beginner to Advanced



MODE
Live Online



LANGUAGE
English



BATCH STARTS
To be announced



ENROLMENT CLOSES
Open

COURSE FEE

₹10,999

₹45,000 - save 76% OFF



LIVE COURSE BROCHURE

Inside this brochure

Everything on the Software Testing & Automation page, in one document. Tap a section to jump to it.

01		Course Overview	03
02		How Our Courses Work	04
03		Eligibility & Prerequisites	05
04		Skills You Will Gain	05
05		How we designed our Product?	06
06		What You Will Learn	08
07		Our Placement Support Services	37
08		Meet Our SME Board Members	39
09		Key Tools and Techniques	40
10		Projects You Will Work On	41
11		Software Testing & Automation	43
12		World's First Ed-Tech with 8C Appraisal	44
13		Frequently Asked Questions	45
14		Ready to take the next step?	46

01 · AT A GLANCE



Course Overview

Learn manual and automation testing with Selenium, API testing, Postman, Java and CI/CD in live mentor-led sessions. Build a job-ready portfolio with real projects and interview preparation.

This live, mentor-led course covers manual testing fundamentals, SQL, Java, Selenium WebDriver, API testing, and CI/CD basics. You learn through guided practice, assignments and nine hands-on projects, and finish with a job-ready automation portfolio and interview preparation for QA roles.



DURATION

4 Months

LEVEL

Beginner to Advanced

MODE

Live Online

LANGUAGE

English

BATCH STARTS

To be announced

ENROLMENT CLOSES

Open

LEARNERS

Trusted by ambitious learners nationwide

COURSE FEE

₹10,999

₹45,000 - save 76% OFF

Highlights

Industry-Led Upskilling Courses

Learn from industry professionals with years of real-world experience.

Personalized Mentorship

Get one-on-one guidance tailored to your career goals and aspirations.

Premium Resources

Access course materials, job portal, free events, challenges and learner 8C Appraisal reports.

Vibrant Community

Connect with like-minded peers and build lasting professional relationships.



02 · SIMPLE 3-STEP JOURNEY

How Our Courses Work

A structured, step-by-step path designed to take you from foundational learning to career success with industry guidance.

01

Step 1: Enroll & Get Started

Choose your course and complete the enrollment process. Access your learning materials instantly and join our community of learners.

✓ Instant access to course materials

✓ Personalized learning dashboard

✓ Welcome orientation session

02

Step 2: Learn & Practice

Engage with interactive lessons, live sessions, and hands-on projects. Learn from industry experts and apply your knowledge in real-world scenarios.

✓ Live classes with expert instructors

✓ Hands-on practical projects

✓ One-on-one mentorship support

03

Step 3: Complete & Get Certified

Finish your coursework, pass assessments, and receive your professional certificate. Launch your new career with confidence and credentials.

✓ Industry-recognized certificate

✓ Portfolio of completed projects

✓ Career placement assistance

03 · BEFORE YOU ENROLL



Eligibility & Prerequisites

Check who this course is designed for and what you should know before your first class.

WHO CAN JOIN

Eligibility

- ✓ Students from any degree or diploma stream
- ✓ Freshers looking to start a career in QA
- ✓ Working professionals planning a move into testing
- ✓ Manual testers who want to learn automation

BEFORE YOU START

Prerequisites

- ✓ Basic computer and internet skills
- ✓ Comfort with logical thinking
- ✓ A laptop with at least 8 GB RAM
- ✓ Willingness to practise coding regularly

04 · CAREER-READY SKILLS



Skills You Will Gain

Hands-on skills you will practise in live classes, assignments and projects throughout the course.

Manual Testing

Test Case Design

SDLC and STLC

Bug Reporting

JIRA

SQL for Testers

Core Java

Selenium WebDriver

TestNG

Page Object Model

Cucumber BDD

API Testing

Postman

REST Assured

Git and GitHub

Jenkins CI/CD

Performance Testing Basics

Test Reporting

05 · PRODUCT ECOSYSTEM ARCHITECTURE



How we designed our Product?

An all-in-one ed-tech ecosystem carefully engineered to bridge learning, practical skill assessment, career guidance, and employment.

WORLD'S FIRST FEATURE

8C Appraisal Report

Proprietary evaluation matrix measuring holistic performance across 8 career pillars for authentic proof-of-work.

Communication

Configuration

Chronicle

Completeness

Cognizance

Conception

Comprehension

Case Study

Industry Validated · 100% Data Score

SUBJECT MATTER EXPERT AI

Insyrio AI

Get instant 24/7 doubt solving and concept clarifications directly from our specialized AI trained on your enrolled courses.

- ✓ 24/7 Course-Specific SME Assistant
- ✓ Instant Step-by-Step Doubt Resolution

Instant Learning Support

GAMIFIED REWARDS

iSure Coins

Use iSure Coins for AI career tools, masterclasses and CertiQ - and part-pay your course fee with them.

- ✓ Earn Perks & Rewards · Pay with Coins

Redeem Rewards Store

AI ENGINE

AI Career Diagnosis

Algorithmic analysis mapping strengths, gaps, and tailored pathways for industry roles.

- ✓ Skill Gap Mapping

Personalized Roadmap

CURRICULUM

Industry-Led Courses

Handcrafted learning modules built directly alongside senior engineering, design, and management leads.

Practical Frameworks

LIVE

Live Classes

Interactive classrooms with live chat, code walk-throughs, and real-time engagement.

Real-time Q&A Support

AUTO-SCAN

AI Resume ATS

Instant ATS optimization scanner ensuring your resume passes screening software effortlessly.

High Parser Score

PRACTICAL**Assignments**

Industry-grade project briefs with detailed feedback and grading from experienced mentors.

Code & Work Reviews

WEEKLY**Challenges**

Weekly hackathons, coding sprints, and design contests to test your problem-solving skills.

Leaderboard Ranking

NETWORKING**Events**

Exclusive expert keynotes, career fairs, and interactive networking workshops.

Free Community Pass

1-ON-1**Live Mentorship**

Direct one-on-one video guidance sessions with experienced industry practitioners.

Personalized Growth

TRACKING**Attendance**

Class-by-class attendance tracking to ensure disciplined, consistent progression.

Attendance Reports

HIRING**Jobs**

Direct hiring pipeline connecting top-performing learners directly with recruiter managers.

Exclusive Placements

FREE ACCESS**Free Books**

Curated digital library filled with essential tech, management, and soft-skill books.

Instant Digital Library

ARTICLES**Blog**

In-depth technical tutorials, industry trend analysis, and career growth strategies.

Updated Weekly

FREE ACCESS**Free Study Material**

Downloadable cheat sheets, interview preparation guides, templates, and revision notes.

One-Click Downloads

06 · COURSE CURRICULUM



What You Will Learn

A carefully sequenced curriculum - every section breaks into modules and hands-on topics taught live by industry experts.

6

SECTIONS

36

MODULES

216

TOPICS

9

PROJECTS

01

SECTION 1

Software Testing Fundamentals and SDLC

Learners build a solid foundation in testing principles, the software development lifecycle, and the core vocabulary and practices used by professional QA teams.

6 modules · 36 topics

M1.1 Introduction to Software Testing · 6 topics

Understand what software testing is, why it matters, and how it supports product quality.

1

What Is Software Testing

Explore the definition and purpose of testing and how it protects users and businesses from defective software.

TESTING BASICS

QUALITY MINDSET

2

Why Testing Matters in Real Projects

Study real-world failures caused by missed defects and the cost of late bug discovery.

RISK AWARENESS

COST OF DEFECTS

3

Errors, Defects and Failures

Learn the difference between a human error, a defect in code, and a failure observed at runtime.

DEFECT CONCEPTS

TERMINOLOGY

4

Seven Principles of Testing

Review the seven widely accepted testing principles such as exhaustive testing being impossible and defect clustering.

ISTQB PRINCIPLES

TEST THINKING

5

Verification versus Validation

Distinguish between building the product right and building the right product with practical examples.

VERIFICATION

VALIDATION

QUALITY ASSURANCE

6

Roles in a Testing Team

Meet the roles of tester, QA analyst, automation engineer, and test lead and see how they collaborate.

TEAM ROLES

COLLABORATION

M1.2

Software Development Life Cycle Essentials · 6 topics

Learn the phases of the SDLC and where testing fits into each one.

1

Phases of the SDLC

Walk through requirements, design, development, testing, deployment, and maintenance.

SDLC

PROCESS OVERVIEW

2

Requirement Analysis for Testers

Learn how testers read and question requirements to find gaps and ambiguities early.

REQUIREMENTS

ANALYSIS

3

Design and Development Touchpoints

See how testers engage with design reviews and developer builds before formal testing begins.

DESIGN REVIEW

EARLY TESTING

4

Waterfall Model

Examine the sequential Waterfall model along with its strengths, limits, and testing approach.

WATERFALL

SEQUENTIAL MODELS

5

V-Model and Test Alignment

Map each development phase to a matching test level using the V-Model.

V-MODEL

TEST LEVELS

6

Deployment and Maintenance Testing

Understand release checks, hotfix validation, and regression needs after software goes live.

RELEASE TESTING

MAINTENANCE

M1.3 Agile and Modern Delivery Models · 6 topics

Explore iterative methodologies and how testing works within fast-moving teams.

1 Agile Principles and Mindset

Learn the Agile manifesto values and what they mean for testers day to day.

[AGILE](#)
[MINDSET](#)

2 Scrum Framework for Testers

Understand sprints, backlogs, ceremonies, and the tester contribution in a Scrum team.

[SCRUM](#)
[SPRINT PLANNING](#)

3 User Stories and Acceptance Criteria

Practice reading user stories and turning acceptance criteria into testable conditions.

[USER STORIES](#)
[ACCEPTANCE CRITERIA](#)

4 Kanban and Continuous Flow

Compare Kanban with Scrum and see how testing fits a continuous flow of work.

[KANBAN](#)
[WORKFLOW](#)

5 DevOps and Continuous Testing

Get an introduction to DevOps culture and why testing is embedded across the delivery pipeline.

[DEVOPS](#)
[CONTINUOUS TESTING](#)

6 Shift Left Testing

Learn how moving testing earlier in the lifecycle reduces defects and rework.

[SHIFT LEFT](#)
[EARLY FEEDBACK](#)

M1.4 Software Testing Life Cycle · 6 topics

Follow the structured phases testers use to plan, execute, and close testing work.

1 Overview of the STLC

See how the testing life cycle runs alongside the SDLC and the entry and exit criteria for each phase.

[STLC](#)
[ENTRY AND EXIT CRITERIA](#)

2 Requirement Analysis in STLC

Identify testable requirements and prepare a requirement traceability starting point.

[TESTABILITY](#)
[TRACEABILITY](#)

3

Test Planning and Strategy

Learn what goes into a test plan including scope, resources, schedule, and risks.

TEST PLAN

SCOPE

RISK

4

Test Case Development Process

Understand how test scenarios become detailed test cases and test data.

TEST CASES

TEST DATA

5

Test Environment Setup

Learn how test environments are configured and verified before execution starts.

ENVIRONMENTS

SMOKE CHECK

6

Test Execution and Closure

Execute tests, record results, report status, and complete closure activities with lessons learned.

EXECUTION

REPORTING

CLOSURE

M1.5

Levels and Types of Testing · 6 topics

Classify testing by level, purpose, and technique to choose the right approach.

1

Unit and Integration Testing

Compare testing of individual components with testing of interactions between modules.

UNIT TESTING

INTEGRATION TESTING

2

System and Acceptance Testing

Learn how complete systems are validated against requirements and business needs.

SYSTEM TESTING

UAT

3

Functional versus Non-Functional Testing

Distinguish what the system does from how well it performs, using clear examples.

FUNCTIONAL

NON-FUNCTIONAL

4

Black Box, White Box and Grey Box

Understand the three testing perspectives and when each is most useful.

BLACK BOX

WHITE BOX

GREY BOX

5

Smoke, Sanity and Regression Testing

Know the purpose and timing of these commonly confused testing types.

SMOKE TESTING

SANITY TESTING

REGRESSION

6

Static versus Dynamic Testing

Compare reviews and walkthroughs with executing code to find defects.

STATIC TESTING

REVIEWS

DYNAMIC TESTING

M1.6

Defects and Quality Foundations · 6 topics

Learn to report, track, and measure defects and understand core quality concepts.

1

Defect Life Cycle

Trace a bug through states such as new, assigned, fixed, retested, and closed.

DEFECT LIFE CYCLE

BUG TRACKING

2

Writing Effective Bug Reports

Write clear, reproducible bug reports with steps, evidence, and expected versus actual results.

BUG REPORTING

CLARITY

REPRODUCIBILITY

3

Severity and Priority

Learn how to classify the technical impact and business urgency of a defect.

SEVERITY

PRIORITY

TRIAGE

4

Introduction to Bug Tracking Tools

Get hands-on with a defect tracking tool to log, assign, and follow up on issues.

JIRA

DEFECT TRACKING

5

Quality Assurance versus Quality Control

Differentiate process-focused QA from product-focused QC and see how both support quality.

QA

QC

PROCESS QUALITY

6

Basic Test Metrics and Reporting

Learn simple metrics such as defect density and test coverage and how to present them to stakeholders.

METRICS

TEST COVERAGE

REPORTING

02

SECTION 2

Test Design, JIRA and SQL for Testers

Learners master test design techniques, defect and project tracking in JIRA, and SQL skills for validating application data.

6 modules · 36 topics

M2.1 Test Case Design Techniques · 6 topics

Learners apply black box design techniques to create effective and efficient test cases.

1 Equivalence Partitioning

Learn to divide input data into valid and invalid classes to reduce the number of test cases. Practice with real form field examples.

EQUIVALENCE PARTITIONING

TEST DESIGN

INPUT ANALYSIS

2 Boundary Value Analysis

Identify boundary conditions where defects commonly occur. Design tests for minimum, maximum and edge values.

BOUNDARY TESTING

EDGE CASES

TEST DESIGN

3 Decision Table Testing

Model complex business rules using decision tables. Derive test cases from combinations of conditions and actions.

DECISION TABLES

BUSINESS RULES

TEST COVERAGE

4 State Transition Testing

Represent application behavior as states and transitions. Create tests that validate valid and invalid state changes.

STATE DIAGRAMS

TRANSITIONS

TEST DESIGN

5 Use Case and Pairwise Testing

Derive scenarios from use cases and reduce combinations using pairwise techniques. Compare coverage against effort.

USE CASES

PAIRWISE

SCENARIO DESIGN

6 Error Guessing and Exploratory Testing

Use experience and intuition to uncover hidden defects. Run time-boxed exploratory sessions with session notes.

ERROR GUESSING

EXPLORATORY TESTING

SESSION NOTES

M2.2 Writing and Managing Test Cases · 6 topics

Learners write clear test artifacts and maintain them for traceability and reuse.

1 Anatomy of a Good Test Case

Understand the fields, structure and clarity needed in a professional test case. Review good and poor examples.

TEST CASE WRITING

CLARITY

DOCUMENTATION

2 Writing Test Scenarios and Test Data

Convert requirements into high-level scenarios and prepare suitable test data sets. Cover positive and negative data.

TEST SCENARIOS

TEST DATA

REQUIREMENTS ANALYSIS

3 Requirement Traceability Matrix

Build a traceability matrix linking requirements to test cases and defects. Use it to find coverage gaps.

RTM

TRACEABILITY

COVERAGE

4 Test Case Review and Peer Feedback

Conduct walkthroughs and peer reviews to improve test quality. Apply review checklists to sample cases.

PEER REVIEW

QUALITY CHECKS

COLLABORATION

5 Test Suites and Regression Selection

Organize test cases into suites and choose regression sets based on risk and change impact.

TEST SUITES

REGRESSION

RISK BASED SELECTION

6 Test Management Tools Overview

Explore how test management tools store, execute and report on test cases. Compare common tool features.

TEST MANAGEMENT

TOOL SELECTION

REPORTING

M2.3 Defect Life Cycle and Bug Reporting · 6 topics

Learners report, track and communicate defects in a way developers can act on quickly.

1 Defect Life Cycle Stages

Walk through every defect status from new to closed and the reasons for reopening or rejecting. Map the flow to team roles.

DEFECT LIFE CYCLE

WORKFLOW

STATUS TRACKING

2 Writing Effective Bug Reports

Write reports with clear titles, steps, expected and actual results and evidence. Practice rewriting weak reports.

BUG REPORTING

REPRODUCIBILITY

COMMUNICATION

3

Severity and Priority Assessment

Distinguish severity from priority and assign both using business impact. Debate realistic scenarios in groups.

SEVERITY

PRIORITY

IMPACT ANALYSIS

4

Root Cause Analysis and Defect Triage

Run triage meetings and apply root cause techniques to find why defects occur. Use findings to prevent repeats.

ROOT CAUSE ANALYSIS

TRIAGE

PREVENTION

5

Defect Metrics and Reports

Calculate defect density, leakage and removal efficiency. Present trends to stakeholders in simple summaries.

DEFECT METRICS

REPORTING

DATA INTERPRETATION

6

Defect Reporting Workshop

Test a sample application and log defects end to end. Receive feedback on report quality and accuracy.

HANDS ON PRACTICE

BUG LOGGING

FEEDBACK

M2.4

JIRA for Testing Teams · 6 topics

Learners use JIRA to manage projects, test work and defects in Agile teams.

1

JIRA Basics and Navigation

Get familiar with projects, issue types, boards and the dashboard. Create and view issues in a practice project.

JIRA

NAVIGATION

ISSUE TYPES

2

Creating and Managing Issues

Create bugs, tasks and stories with the right fields, labels and links. Assign and update issues through their life.

ISSUE MANAGEMENT

LABELS

LINKING

3

Workflows and Custom Fields

Understand how workflows control status changes and how custom fields capture extra data for testing needs.

WORKFLOWS

CUSTOM FIELDS

CONFIGURATION

4

Scrum and Kanban Boards

Use sprint and Kanban boards to plan and track testing work. Follow a story from backlog to done.

SCRUM BOARD

KANBAN

SPRINT TRACKING

5

JQL Filters and Dashboards

Write JQL queries to find issues and build dashboards with gadgets for daily test status.

JQL

FILTERS

DASHBOARDS

6

Test Management in JIRA

Manage test cases and executions using JIRA add-ons or native structures. Link tests to requirements and bugs.

TEST EXECUTION

ADD-ONS

TRACEABILITY

M2.5

SQL Fundamentals for Testers · 6 topics

Learners write core SQL queries to read and verify application data.

1

Database Concepts for Testers

Learn tables, keys, relationships and why testers need database knowledge. Explore a sample schema.

RDBMS

KEYS

SCHEMA READING

2

SELECT, WHERE and Sorting

Retrieve and filter records using SELECT, WHERE, ORDER BY and operators. Practice on realistic tables.

SELECT

WHERE

ORDER BY

3

Aggregate Functions and GROUP BY

Summarize data using COUNT, SUM, AVG, MIN and MAX with GROUP BY and HAVING clauses.

AGGREGATION

GROUP BY

HAVING

4

Joins Across Tables

Combine data using inner, left, right and full joins. Verify data spread over multiple tables.

JOINS

MULTI TABLE QUERIES

DATA VERIFICATION

5

INSERT, UPDATE and DELETE Safely

Modify test data with DML statements and understand transactions. Apply safe habits to avoid data loss.

DML

TRANSACTIONS

TEST DATA SETUP

6

Subqueries and Functions

Use nested queries and common string, date and null handling functions to answer complex questions.

SUBQUERIES

SQL FUNCTIONS

NULL HANDLING

M2.6

Database Testing in Practice · 6 topics

Learners apply SQL to validate data integrity, business logic and application behavior.

1

Data Validation Between UI and Database

Compare values shown in the application with stored records. Spot mismatches and report them clearly.

DATA VALIDATION

UI TO DB CHECKS

VERIFICATION

2 Testing Constraints and Data Integrity

Verify primary key, foreign key, unique and not null rules. Design tests that try to break them.

[CONSTRAINTS](#)
[DATA INTEGRITY](#)
[NEGATIVE TESTING](#)

3 Views, Indexes and Stored Procedures

Understand these database objects and how to test their results and behavior from a tester's view.

[VIEWS](#)
[INDEXES](#)
[STORED PROCEDURES](#)

4 Detecting Duplicates and Orphan Records

Write queries that find duplicate rows, missing references and inconsistent data after processing.

[DUPLICATE DETECTION](#)
[ORPHAN RECORDS](#)
[DATA QUALITY](#)

5 Data Migration and ETL Testing Basics

Check record counts, mappings and transformations when data moves between systems.

[DATA MIGRATION](#)
[ETL BASICS](#)
[RECONCILIATION](#)

6 Capstone Practice: Design, Log and Query

Complete a combined exercise that designs tests, logs defects in JIRA and validates results with SQL.

[END TO END PRACTICE](#)
[JIRA](#)
[SQL](#)

03

SECTION 3

Core Java for Automation

Learners build practical Java programming skills that form the foundation for writing maintainable test automation code.

6 modules · 36 topics

M3.1 Java Basics and Environment Setup · 6 topics

Learners set up their Java toolchain and write their first programs using core syntax.

1 Java Platform Overview and JDK Installation

Understand the roles of JDK, JRE and JVM and install Java on your machine. Verify the setup from the command line.

[JDK](#)
[JVM](#)
[SETUP](#)

2 IDE Setup with IntelliJ IDEA or Eclipse

Install an IDE and create a first project. Learn to run and debug a basic program.

[IDE](#)
[PROJECT SETUP](#)
[DEBUGGING](#)

3 Writing Your First Java Program

Explore class structure and the main method. Compile and run a simple console program.

[SYNTAX](#)
[MAIN METHOD](#)
[COMPILATION](#)

4 Variables and Data Types

Work with primitive and reference data types and declare variables correctly. Practice with literals and naming conventions.

[DATA TYPES](#)
[VARIABLES](#)
[NAMING](#)

5 Operators and Expressions

Use arithmetic, relational, logical and assignment operators in expressions. Understand operator precedence through examples.

[OPERATORS](#)
[EXPRESSIONS](#)
[LOGIC](#)

6 Type Casting and User Input

Convert between data types using implicit and explicit casting. Read user input with the Scanner class.

[TYPE CASTING](#)
[SCANNER](#)
[INPUT HANDLING](#)

M3.2 Control Flow and Loops · 6 topics

Learners control program execution using decisions and repetition, which are essential for test logic.

1 If Else Statements

Write conditional logic using if, else if and else. Apply it to simple validation scenarios.

[CONDITIONALS](#)
[VALIDATION](#)
[LOGIC](#)

2 Switch Statements

Use switch for multi-branch decisions such as choosing a browser. Understand break and default behavior.

[SWITCH](#)
[BRANCHING](#)
[BROWSER SELECTION](#)

3 For and Enhanced For Loops

Iterate over ranges and collections using for loops. Apply loops to repeat test steps.

[FOR LOOP](#)
[ITERATION](#)
[REPETITION](#)

4 While and Do While Loops

Use condition-based loops for scenarios such as waiting or retrying. Compare the loop types and their use cases.

[WHILE LOOP](#)
[RETRY LOGIC](#)
[CONDITIONS](#)

5 Break Continue and Nested Loops

Control loop flow with break and continue. Build nested loops for pattern and data grid problems.

[BREAK](#)
[CONTINUE](#)
[NESTED LOOPS](#)

6 Logic Building Practice for Testers

Solve common problems such as string reversal, palindromes and number checks. These exercises mirror typical interview questions.

[PROBLEM SOLVING](#)
[LOGIC BUILDING](#)
[INTERVIEW PREP](#)

M3.3 Arrays Strings and Methods · 6 topics

Learners handle common data structures and reusable code blocks used in everyday automation scripts.

1 Single and Multidimensional Arrays

Declare, initialize and traverse arrays. Use two-dimensional arrays to represent tabular test data.

[ARRAYS](#)
[TRAVERSAL](#)
[TEST DATA](#)

2 String Class and Immutability

Understand how strings are stored and why they are immutable. Learn the string pool and comparison pitfalls.

[STRINGS](#)
[IMMUTABILITY](#)
[STRING POOL](#)

3 Common String Methods for Validation

Use methods like substring, split, contains, trim and equals. Apply them to verify text on web pages.

[STRING METHODS](#)
[TEXT VALIDATION](#)
[PARSING](#)

4 StringBuilder and StringBuffer

Build and modify strings efficiently. Compare mutable and immutable string options.

[STRINGBUILDER](#)
[PERFORMANCE](#)
[MUTABILITY](#)

5 Methods and Parameters

Create methods with parameters and return values. Organize repeated test steps into reusable methods.

[METHODS](#)
[PARAMETERS](#)
[REUSABILITY](#)

6 Method Overloading and Variable Scope

Define multiple methods with the same name and different signatures. Understand local and instance variable scope.

[OVERLOADING](#)
[SCOPE](#)
[SIGNATURES](#)

M3.4 Object Oriented Programming for Frameworks · 6 topics

Learners apply object oriented principles that underpin well-designed automation frameworks.

1 Classes Objects and Constructors

Model real entities as classes and create objects. Use constructors to initialize state.

CLASSES

OBJECTS

CONSTRUCTORS

2 Encapsulation and Access Modifiers

Protect data using private fields with getters and setters. Learn how access modifiers control visibility.

ENCAPSULATION

ACCESS MODIFIERS

GETTERS SETTERS

3 Inheritance and the Super Keyword

Reuse code by extending classes. Apply inheritance to create a base test class.

INHERITANCE

SUPER

BASE CLASS

4 Polymorphism and Method Overriding

Understand compile-time and runtime polymorphism. Override methods and relate this to WebDriver references.

POLYMORPHISM

OVERRIDING

WEBDRIVER CONCEPT

5 Abstraction with Abstract Classes and Interfaces

Hide implementation details using abstract classes and interfaces. Learn when to choose each approach.

ABSTRACTION

INTERFACES

DESIGN

6 Static Final and This Keywords

Use static members for shared utilities and final for constants. Understand the this keyword and static blocks.

STATIC

FINAL

UTILITIES

M3.5 Collections Exceptions and Generics · 6 topics

Learners manage groups of data and handle errors gracefully in automation code.

1 List and ArrayList

Store ordered data in lists and perform add, remove and search operations. Use lists to hold web elements and test data.

LIST

ARRAYLIST

WEB ELEMENTS

2 Set and HashSet

Work with unique collections and understand ordering differences. Use sets to remove duplicates from results.

SET

HASHSET

UNIQUENESS

3

Map and HashMap

Store key-value pairs and iterate over entries. Apply maps to manage configuration and test data.

MAP

HASHMAP

KEY VALUE

4

Generics and Iterators

Write type-safe collections using generics. Traverse collections with iterators.

GENERICS

ITERATOR

TYPE SAFETY

5

Exception Handling with Try Catch Finally

Handle runtime errors using try, catch and finally blocks. Understand checked and unchecked exceptions.

EXCEPTIONS

TRY CATCH

ERROR HANDLING

6

Throw Throws and Custom Exceptions

Raise exceptions deliberately and declare them in method signatures. Create custom exceptions for framework errors.

THROW

THROWS

CUSTOM EXCEPTIONS

M3.6

Java Utilities for Test Automation · 6 topics

Learners use Java features for file handling, data access and reusable utilities commonly needed in automation projects.

1

File Handling with File and Reader Classes

Read and write text files using Java IO classes. Use files to store logs and test output.

FILE IO

READER WRITER

LOGS

2

Reading Properties Files for Configuration

Load configuration values such as URLs and browser names from properties files. Keep environment settings out of code.

PROPERTIES

CONFIGURATION

ENVIRONMENTS

3

Reading Excel Data with Apache POI

Add the POI library and read test data from Excel sheets. Feed the data into data-driven tests.

APACHE POI

EXCEL

DATA DRIVEN

4

Date Time and Random Data Utilities

Generate timestamps and random values for unique test data. Format dates for reports and screenshots.

DATE TIME

RANDOM DATA

UTILITIES

5

Introduction to Maven and Dependencies

Create a Maven project and manage libraries through the pom file. Understand the standard project structure.

MAVEN

DEPENDENCIES

POM

6

Packages Git Basics and Code Organization

Organize code into packages and follow naming standards. Commit and push a Java project using Git.

PACKAGES

GIT

CODE STANDARDS

04

SECTION 4

Selenium WebDriver and TestNG Framework

Learners build robust browser automation using Selenium WebDriver with Java and organize it into a scalable TestNG based test framework.

6 modules · 36 topics

M4.1

Selenium Setup and Browser Automation Basics · 6 topics

Introduces the Selenium ecosystem and the first steps of driving a real browser from Java code.

1

Introduction to Test Automation and Selenium Suite

Understand when to automate and how Selenium WebDriver, IDE and Grid fit together. Review the automation pyramid and tool selection criteria.

AUTOMATION STRATEGY

SELENIUM ECOSYSTEM

2

Setting Up Eclipse, Maven and Selenium Dependencies

Create a Maven project and add Selenium and browser driver dependencies. Verify the setup with a first running script.

MAVEN

ECLIPSE

PROJECT SETUP

3

WebDriver Architecture and Driver Management

Explore how WebDriver communicates with browsers through drivers. Configure Chrome, Firefox and Edge using Selenium Manager.

WEBDRIVER ARCHITECTURE

BROWSER DRIVERS

4

Launching Browsers and Basic Browser Commands

Use get, navigate, back, forward, refresh, close and quit commands. Retrieve page title and URL for simple validations.

BROWSER COMMANDS

NAVIGATION

5 Your First Selenium Test Script

Write an end to end script that opens a site, performs an action and verifies the result. Learn to read and fix common startup errors.

SCRIPTING

DEBUGGING

6 Browser Window Handling and Options

Manage window size, position and maximize operations. Apply ChromeOptions for incognito, headless and notification settings.

BROWSER OPTIONS

HEADLESS MODE

WINDOW MANAGEMENT

M4.2 Locators and Web Element Interaction · 6 topics

Teaches how to find elements reliably and interact with every common type of web control.

1 Inspecting the DOM and Using Developer Tools

Read HTML structure and attributes using browser developer tools. Identify stable properties for automation.

DOM

DEVTOOLS

2 Basic Locators: ID, Name, Class, Tag and Link Text

Apply the eight Selenium locator strategies to find elements. Compare their speed and reliability.

LOCATORS

FINDELEMENT

3 XPath Fundamentals and Axes

Write absolute and relative XPath expressions with attributes, functions and axes. Handle dynamic elements with contains and starts-with.

XPATH

DYNAMIC ELEMENTS

4 CSS Selectors for Fast and Clean Locators

Build CSS selectors using attributes, classes and hierarchy. Decide when CSS is better than XPath.

CSS SELECTORS

LOCATOR STRATEGY

5 Working with Text Boxes, Buttons, Checkboxes and Radio Buttons

Perform sendKeys, click, clear and state checks on common elements. Validate isDisplayed, isEnabled and isSelected.

WEBELEMENT METHODS

FORM HANDLING

6 Handling Dropdowns, Tables and Lists of Elements

Use the Select class for dropdowns and findElements for groups of elements. Read data from static and dynamic web tables.

SELECT CLASS

WEB TABLES

FINDELEMENTS

M4.3 Advanced Browser Interactions and Synchronization · 6 topics

Covers complex user actions and the waiting techniques that make scripts stable.

1 Implicit and Explicit Waits

Understand why timing failures occur and how to remove them. Use WebDriverWait with ExpectedConditions.

IMPLICIT WAIT

EXPLICIT WAIT

2 Fluent Wait and Handling Stale Elements

Configure polling and ignored exceptions with FluentWait. Resolve StaleElementReferenceException in dynamic pages.

FLUENTWAIT

EXCEPTION HANDLING

3 Alerts, Pop-ups and Authentication Dialogs

Accept, dismiss and read JavaScript alerts and prompts. Handle basic authentication through the URL.

ALERTS

POP-UPS

4 Frames, Windows and Tabs

Switch between iframes, child windows and tabs using window handles. Return control to the parent context safely.

SWITCHTO

WINDOW HANDLES

IFRAMES

5 Actions Class for Mouse and Keyboard Events

Automate hover, drag and drop, double click, right click and key combinations. Build chained action sequences.

ACTIONS CLASS

MOUSE EVENTS

6 JavaScript Executor, Scrolling and File Upload

Run JavaScript to scroll and click hidden elements. Automate file uploads and capture screenshots.

JAVASCRIPTEXECUTOR

SCREENSHOTS

FILE UPLOAD

M4.4 TestNG Fundamentals · 6 topics

Introduces TestNG as the engine for organizing, running and reporting automated tests.

1 Introduction to TestNG and Installation

Learn why TestNG is preferred over plain Java main methods. Install the plugin and run the first test class.

TESTNG

SETUP

2 TestNG Annotations and Execution Order

Use BeforeSuite, BeforeTest, BeforeClass, BeforeMethod, Test and their After counterparts. Trace the exact order of execution.

ANNOTATIONS

LIFECYCLE

3

Test Attributes: Priority, Enabled, Timeout and Dependencies

Control the order and conditions of test execution. Use dependsOnMethods and invocationCount effectively.

PRIORITY

DEPENDENCIES

TEST CONTROL

4

Assertions: Hard Assert and Soft Assert

Validate expected results using Assert methods. Compare the behavior of hard and soft assertions with SoftAssert.

ASSERTIONS

VALIDATION

5

testng.xml, Groups and Parameters

Configure suites, tests and classes in testng.xml. Run groups and pass parameters from XML.

TESTNG.XML

GROUPS

PARAMETERS

6

Data Driven Testing with DataProvider

Feed multiple data sets into one test using DataProvider. Separate test data from test logic.

DATAPROVIDER

DATA DRIVEN TESTING

M4.5

Automation Framework Design with Page Object Model · 6 topics

Guides learners in structuring reusable and maintainable framework code.

1

Framework Types and Architecture Overview

Compare linear, modular, data driven and hybrid frameworks. Plan the folder structure for a scalable project.

FRAMEWORK DESIGN

ARCHITECTURE

2

Page Object Model Concepts

Represent each web page as a Java class with locators and actions. Understand the benefits for maintenance.

POM

ENCAPSULATION

3

Page Factory and FindBy Annotations

Initialize elements using PageFactory and the FindBy annotation. Compare it with standard By locators.

PAGEFACTORY

FINDBY

4

Base Class, Utilities and Reusable Methods

Create a base test class for setup and teardown. Build utilities for waits, screenshots and common actions.

BASE CLASS

UTILITIES

REUSABILITY

5

Reading Data from Excel, Properties and JSON Files

Use Apache POI and Java libraries to read external data. Store configuration values in properties files.

APACHE POI

PROPERTIES FILE

JSON

6 Building a Complete E-commerce Test Suite

Apply POM to automate login, search, cart and checkout flows. Review the code for design improvements.

[END TO END TESTING](#)[POM IMPLEMENTATION](#)**M4.6 Reporting, Logging and Execution at Scale** · 6 topics

Shows how to produce clear results and run tests efficiently across environments.

1 Logging with Log4j2

Add structured logs to track test execution and failures. Configure log levels and output files.

[LOG4J2](#)[LOGGING](#)**2 Extent Reports Integration**

Generate rich HTML reports with steps, status and screenshots. Attach reports to TestNG listeners.

[EXTENT REPORTS](#)[REPORTING](#)**3 TestNG Listeners and Retry Analyzer**

Capture screenshots on failure using ITestListener. Rerun flaky tests with IRetryAnalyzer.

[LISTENERS](#)[RETRY ANALYZER](#)**4 Cross Browser and Parallel Execution**

Run tests on multiple browsers in parallel using testng.xml. Use ThreadLocal to keep drivers thread safe.

[PARALLEL TESTING](#)[THREADLOCAL](#)[CROSS BROWSER](#)**5 Selenium Grid and Cloud Execution Basics**

Understand hub and node architecture and remote execution with RemoteWebDriver. Explore cloud testing platforms.

[SELENIUM GRID](#)[REMOTEWEBDRIVER](#)**6 Version Control and Jenkins Integration**

Push the framework to Git and trigger builds from Jenkins using Maven. Schedule runs and publish reports.

[GIT](#)[JENKINS](#)[CI BASICS](#)

05

SECTION 5

API Testing and BDD with Cucumber

Learners test REST APIs manually and with automation tools, then adopt Behavior Driven Development using Cucumber to build readable and maintainable test suites.

6 modules · 36 topics

M5.1 REST API and Web Services Basics · 6 topics

Builds a solid understanding of how web services and REST APIs work before any testing begins.

1 Introduction to Web Services and APIs

Understand what APIs are and how client and server applications communicate. Compare SOAP and REST at a high level.

API CONCEPTS

WEB SERVICES

SOAP VS REST

2 HTTP Methods and Status Codes

Learn GET, POST, PUT, PATCH and DELETE and the meaning of common response codes. Map each method to real application actions.

HTTP METHODS

STATUS CODES

CRUD

3 Request and Response Anatomy

Examine URLs, headers, query parameters, path parameters and request bodies. Read responses carefully to spot defects.

HEADERS

PARAMETERS

PAYLOADS

4 JSON and XML Data Formats

Read and write JSON and XML structures used in API payloads. Practice navigating nested objects and arrays.

JSON

XML

DATA PARSING

5 Authentication and Authorization Basics

Explore API keys, basic authentication, bearer tokens and OAuth concepts. Understand how access control affects test scenarios.

AUTHENTICATION

OAUTH

TOKENS

6 Planning API Test Scenarios

Design positive, negative and boundary test cases for API endpoints. Prepare an API test checklist from documentation.

API TEST DESIGN

NEGATIVE TESTING

DOCUMENTATION REVIEW

M5.2 Manual API Testing with Postman · 6 topics

Teaches learners to explore, validate and organize API tests using Postman.

1 Postman Setup and First Requests

Install Postman and send the first requests to a public API. Understand the workspace layout and response viewer.

[POSTMAN](#)
[API REQUESTS](#)
[WORKSPACE](#)

2 Collections and Folders

Organize requests into collections and folders for reuse. Document requests so teammates can follow them.

[COLLECTIONS](#)
[ORGANIZATION](#)
[DOCUMENTATION](#)

3 Environments and Variables

Use environment, global and collection variables to avoid hardcoded values. Switch between test environments quickly.

[ENVIRONMENTS](#)
[VARIABLES](#)
[REUSABILITY](#)

4 Writing Tests and Assertions in Postman

Add JavaScript test scripts to verify status codes, response times and body values. Interpret test results in the runner.

[ASSERTIONS](#)
[TEST SCRIPTS](#)
[VALIDATION](#)

5 Pre-request Scripts and Chaining Requests

Pass data from one response into the next request. Automate token generation and dynamic payloads.

[PRE-REQUEST SCRIPTS](#)
[REQUEST CHAINING](#)
[DYNAMIC DATA](#)

6 Data Driven Runs and Newman

Run collections with CSV or JSON data files. Execute collections from the command line using Newman and generate reports.

[COLLECTION RUNNER](#)
[NEWMAN](#)
[DATA DRIVEN TESTING](#)

M5.3 API Automation with Rest Assured · 6 topics

Moves from manual API checks to coded automation using Java and Rest Assured.

1 Rest Assured Setup and Project Structure

Create a Maven project and add Rest Assured and supporting libraries. Organize packages for API tests.

[REST ASSURED](#)
[MAVEN](#)
[PROJECT SETUP](#)

2 Given When Then Request Syntax

Write the first automated GET and POST tests using the fluent syntax. Understand how each part of a request maps to code.

[GIVEN WHEN THEN](#)
[GET AND POST](#)
[FLUENT API](#)

3 Validating Responses and JSON Paths

Assert on status, headers and body fields using matchers and JsonPath. Extract values for later use.

JSONPATH

MATCHERS

RESPONSE VALIDATION

4 Request Specifications and Reusable Builders

Create request and response specifications to remove duplication. Centralize base URI and common headers.

REQUESTSPECIFICATION

REUSABILITY

CLEAN CODE

5 Serialization and Deserialization with POJOs

Convert Java objects to JSON payloads and parse responses back into objects. Use Jackson or Gson for mapping.

POJO

JACKSON

SERIALIZATION

6 Integrating Rest Assured with TestNG

Run API tests as TestNG suites with groups, data providers and reports. Handle authentication and test data cleanup.

TESTNG INTEGRATION

DATA PROVIDERS

API SUITES

M5.4 Behavior Driven Development and Gherkin · 6 topics

Introduces BDD principles and teaches learners to write clear Gherkin feature files.

1 Introduction to BDD and the Three Amigos

Learn why BDD improves collaboration between business, developers and testers. Understand the Three Amigos conversation.

BDD

COLLABORATION

THREE AMIGOS

2 Gherkin Syntax and Feature Files

Write features and scenarios using Given, When, Then, And and But. Follow naming conventions for clarity.

GHERKIN

FEATURE FILES

SCENARIOS

3 Writing Good Scenarios

Apply declarative style and avoid UI-heavy or overly long steps. Refactor poor scenarios into clean ones.

SCENARIO DESIGN

DECLARATIVE STYLE

REFACTORING

4 Background, Scenario Outline and Examples

Reduce repetition with Background and run one scenario against many data sets. Structure Examples tables effectively.

BACKGROUND

SCENARIO OUTLINE

EXAMPLES TABLE

5 Tags and Organizing Feature Files

Use tags to group smoke, regression and module scenarios. Plan a folder structure for large projects.

TAGS

TEST ORGANIZATION

REGRESSION SUITES

6 From User Stories to Acceptance Criteria

Convert JIRA user stories into testable Gherkin scenarios. Review scenarios with a peer for completeness.

USER STORIES

ACCEPTANCE CRITERIA

JIRA TO GHERKIN

M5.5 Cucumber with Java and Selenium · 6 topics

Implements Gherkin scenarios in code and connects Cucumber with Selenium and TestNG.

1 Cucumber Setup with Maven

Add Cucumber, JUnit or TestNG dependencies and configure the IDE plugin. Run the first feature file.

CUCUMBER

MAVEN

CONFIGURATION

2 Step Definitions and Glue Code

Generate and write step definitions that match Gherkin steps. Use parameters and regular expressions in steps.

STEP DEFINITIONS

GLUE CODE

CUCUMBER EXPRESSIONS

3 Test Runner and Execution Options

Configure the runner class to select features, tags and plugins. Run subsets of scenarios from the command line.

TEST RUNNER

TAGS EXECUTION

CLI RUN

4 Hooks and Dependency Injection

Use Before and After hooks for setup and teardown. Share state between steps using PicoContainer.

HOOKS

PICOCONTAINER

SHARED STATE

5 Integrating Selenium with Cucumber and Page Objects

Drive browser scenarios through step definitions that call page object methods. Keep steps thin and maintainable.

SELENIUM INTEGRATION

PAGE OBJECT MODEL

MAINTAINABILITY

6 Combining API Tests in Cucumber Scenarios

Use Rest Assured inside step definitions for API setup and validation. Mix API and UI steps in one scenario.

API IN BDD

HYBRID TESTING

REST ASSURED

M5.6 BDD Reporting and API Framework Practice · 6 topics

Wraps up the section with reporting, parallel execution and a hands-on framework build.

1 Cucumber Reports and Extent Reports

Generate HTML and JSON reports from Cucumber runs. Add screenshots and logs for failed scenarios.

[CUCUMBER REPORTS](#)[EXTENT REPORTS](#)[FAILURE EVIDENCE](#)**2 Parallel Execution in Cucumber**

Run scenarios in parallel with TestNG and manage thread safe drivers. Diagnose flaky results caused by shared state.

[PARALLEL EXECUTION](#)[THREAD SAFETY](#)[FLAKY TESTS](#)**3 Configuration and Test Data Management**

Externalize URLs, credentials and environments using property files. Manage test data for API and UI scenarios.

[PROPERTIES FILES](#)[TEST DATA](#)[ENVIRONMENTS](#)**4 Logging and Debugging API and BDD Tests**

Add logging with Log4j and inspect request and response logs. Troubleshoot common step and API failures.

[LOG4J](#)[DEBUGGING](#)[TROUBLESHOOTING](#)**5 Hands-on Project: API and BDD Framework**

Build a small framework that combines Rest Assured, Cucumber and reporting against a sample application. Work in guided pairs.

[FRAMEWORK DESIGN](#)[PROJECT WORK](#)[INTEGRATION](#)**6 Project Review and Interview Preparation**

Present the project and review it against best practices. Practice common API testing and BDD interview questions.

[CODE REVIEW](#)[INTERVIEW PREP](#)[PRESENTATION](#)

06

SECTION 6

CI/CD, Performance Testing and Career Preparation

Learners integrate automation suites into CI/CD pipelines, measure application performance, and prepare for interviews and real testing roles.

6 modules · 36 topics

M6.1 Version Control with Git and GitHub · 6 topics

Learners manage automation code collaboratively using Git workflows and GitHub repositories.

1 Git Basics and Repository Setup

Understand version control concepts and create local repositories. Practice init, add, commit and log commands on a test project.

GIT

VERSION CONTROL

COMMITTS

2 Branching and Merging Strategies

Create feature branches and merge changes into the main line. Learn how branching supports parallel test development.

BRANCHING

MERGING

WORKFLOW

3 Resolving Merge Conflicts

Identify and fix conflicts that arise when multiple testers edit the same files. Use diff tools and best practices to avoid repeat issues.

CONFLICT RESOLUTION

DIFF

COLLABORATION

4 GitHub Remote Repositories and Pull Requests

Push automation projects to GitHub and raise pull requests for review. Learn code review etiquette for test code.

GITHUB

PULL REQUESTS

CODE REVIEW

5 Git Stash, Rebase and Tagging

Use stash to save work in progress and rebase to keep history clean. Apply tags to mark framework releases.

STASH

REBASE

TAGS

6 Project Structure and Gitignore Best Practices

Organize an automation framework repository with proper ignore rules and README documentation. Prepare the project for pipeline integration.

GITIGNORE

REPOSITORY DESIGN

DOCUMENTATION

M6.2 Build Management with Maven · 6 topics

Learners automate builds and dependency management for test frameworks using Maven.

1 Introduction to Maven and Project Object Model

Learn Maven architecture and the structure of pom.xml. Create a Maven based automation project from scratch.

[MAVEN](#)
[POM](#)
[PROJECT SETUP](#)

2 Managing Dependencies and Repositories

Add Selenium, TestNG and reporting libraries as dependencies. Understand local and central repositories and version handling.

[DEPENDENCIES](#)
[REPOSITORIES](#)
[VERSIONING](#)

3 Maven Lifecycle and Build Phases

Walk through clean, compile, test, package and install phases. See how each phase affects test execution.

[LIFECYCLE](#)
[BUILD PHASES](#)
[GOALS](#)

4 Surefire Plugin and Running TestNG Suites

Configure the Surefire plugin to run TestNG XML suites from the command line. Pass parameters and groups at runtime.

[SUREFIRE](#)
[TESTNG SUITES](#)
[CLI EXECUTION](#)

5 Maven Profiles and Environment Configuration

Use profiles to switch between QA, staging and production settings. Control browsers and URLs without changing code.

[PROFILES](#)
[ENVIRONMENTS](#)
[CONFIGURATION](#)

6 Multi Module Projects and Build Troubleshooting

Structure larger frameworks into modules and diagnose common build failures. Read Maven logs to find root causes quickly.

[MULTI MODULE](#)
[DEBUGGING](#)
[BUILD LOGS](#)

M6.3 Continuous Integration with Jenkins · 6 topics

Learners build automated pipelines that run test suites on every code change.

1 CI/CD Concepts and Jenkins Installation

Understand continuous integration and delivery principles and the tester's role in them. Install and configure Jenkins locally.

[CI/CD](#)
[JENKINS](#)
[SETUP](#)

2 Creating Freestyle Jobs for Test Execution

Configure a Jenkins job that pulls code from GitHub and runs Maven tests. Review console output and build status.

[FREESTYLE JOBS](#)
[MAVEN INTEGRATION](#)
[CONSOLE OUTPUT](#)

3

Build Triggers and Scheduling

Set up scheduled builds, polling and webhook triggers. Run nightly regression suites automatically.

TRIGGERS

CRON SCHEDULING

WEBHOOKS

4

Parameterized Builds and Jenkins Plugins

Pass browser, environment and suite choices as build parameters. Install plugins for reports and notifications.

PARAMETERS

PLUGINS

REPORTS

5

Jenkins Pipeline as Code

Write a declarative Jenkinsfile with checkout, build, test and report stages. Store the pipeline alongside the project code.

JENKINSFILE

DECLARATIVE PIPELINE

STAGES

6

Email Notifications and Test Reporting in Pipelines

Publish test reports and send build results to the team by email. Handle failures with clear alerts and retries.

NOTIFICATIONS

REPORTING

FAILURE HANDLING

M6.4

Cross Browser, Parallel and Cloud Execution · 6 topics

Learners scale test execution across browsers, machines and cloud platforms.

1

Selenium Grid Architecture

Learn the hub and node model for distributed execution. Set up a basic grid on a local network.

SELENIUM GRID

HUB AND NODE

DISTRIBUTED TESTING

2

Running Tests on Remote WebDriver

Configure RemoteWebDriver and capabilities to target grid nodes. Execute the same suite on multiple browsers.

REMOTEWEBDRIVER

CAPABILITIES

CROSS BROWSER

3

Parallel Execution with TestNG

Run tests in parallel at method, class and suite levels. Make the framework thread safe using ThreadLocal.

PARALLEL TESTING

THREADLOCAL

THREAD SAFETY

4

Docker for Test Environments

Use Docker containers to spin up browsers and grid nodes quickly. Understand images, containers and compose files for testers.

DOCKER

CONTAINERS

DOCKER COMPOSE

5 Cloud Testing Platforms Overview

Explore cloud device and browser farms such as BrowserStack and Sauce Labs. Run an existing suite on cloud infrastructure.

[CLOUD TESTING](#)
[BROWSERSTACK](#)
[DEVICE COVERAGE](#)

6 Flaky Test Management and Execution Optimization

Diagnose unstable tests and apply retry logic and smart waits. Reduce total suite runtime without losing coverage.

[FLAKY TESTS](#)
[RETRY ANALYZER](#)
[OPTIMIZATION](#)

M6.5 Performance Testing with JMeter · 6 topics

Learners design and run load tests and analyze performance results using Apache JMeter.

1 Performance Testing Fundamentals

Understand load, stress, spike and endurance testing and key metrics such as response time and throughput. Learn when performance testing fits in the lifecycle.

[LOAD TESTING](#)
[METRICS](#)
[PERFORMANCE TYPES](#)

2 JMeter Installation and Test Plan Basics

Install JMeter and explore its interface. Build a first test plan with a thread group and HTTP sampler.

[JMETER](#)
[TEST PLAN](#)
[THREAD GROUP](#)

3 Recording Scripts and Using Samplers

Record user journeys with the HTTP test script recorder. Edit samplers to model realistic web traffic.

[SCRIPT RECORDING](#)
[HTTP SAMPLER](#)
[USER JOURNEYS](#)

4 Parameterization, Correlation and Assertions

Feed test data from CSV files and extract dynamic values from responses. Add assertions to validate correctness under load.

[CSV DATA](#)
[CORRELATION](#)
[ASSERTIONS](#)

5 Listeners, Reports and Result Analysis

Use listeners and the HTML dashboard report to study results. Identify bottlenecks from percentiles and error rates.

[LISTENERS](#)
[DASHBOARD REPORT](#)
[BOTTLENECK ANALYSIS](#)

6 Running JMeter in Non GUI Mode and in Pipelines

Execute load tests from the command line and integrate them into Jenkins. Apply good practices for distributed load generation.

[NON GUI MODE](#)
[JENKINS INTEGRATION](#)
[DISTRIBUTED LOAD](#)

M6.6 Career Preparation and Capstone · 6 topics

Learners consolidate their skills through a capstone project and prepare for interviews and job roles.

1 Capstone Project Planning and Framework Design

Plan an end to end automation project covering UI, API and reporting. Define scope, architecture and deliverables.

[PROJECT PLANNING](#)
[FRAMEWORK DESIGN](#)
[SCOPE DEFINITION](#)

2 Capstone Build and Pipeline Integration

Implement the capstone framework and connect it to GitHub, Maven and Jenkins. Demonstrate a fully automated run with reports.

[INTEGRATION](#)
[END TO END AUTOMATION](#)
[REPORTING](#)

3 Resume Building for Testing Roles

Craft a resume that highlights tools, projects and measurable results. Tailor it to manual, automation and SDET openings.

[RESUME WRITING](#)
[PORTFOLIO](#)
[ROLE TARGETING](#)

4 Technical Interview Preparation

Practice commonly asked questions on testing concepts, Java, Selenium, API and SQL. Learn how to explain framework design clearly.

[INTERVIEW QUESTIONS](#)
[FRAMEWORK EXPLANATION](#)
[PROBLEM SOLVING](#)

5 Mock Interviews and Live Coding Practice

Take part in mock technical rounds with live coding and scenario questions. Receive structured feedback on answers and communication.

[MOCK INTERVIEWS](#)
[LIVE CODING](#)
[FEEDBACK](#)

6 LinkedIn Profile, Job Search and Workplace Readiness

Build a professional online presence and learn effective job search strategies. Understand agile team practices and expectations in the first months on the job.

[LINKEDIN](#)
[JOB SEARCH](#)
[AGILE TEAMS](#)

07 · PLACEMENT TRAINING



Our Placement Support Services

Placement Training comes with the Placement Assistance plan. Once your course is completed, our placement trainers take you through these six live sessions one by one - finish all six to earn the Placement Training Completion Certificate.

01

Complete your course

Placement Training starts after course completion

02

6 sessions, one by one

Each step is scheduled after the previous one is done

03

Get certified

Placement Training Completion Certificate

STEP 1 OF 6 · LIVE SESSION

1:1 Mock Interview

A live one-to-one mock interview with an industry expert, followed by honest feedback on your answers, confidence and presence.

- ✓ Technical & HR round simulation
- ✓ Questions from your course domain
- ✓ Body language & confidence pointers
- ✓ Personal improvement checklist

1:1 Live

Online

Feedback

After course completion · live with a placement trainer

STEP 2 OF 6 · EXPERT REVIEW

Resume Optimization

Your resume reviewed line by line and reshaped into a clear, ATS-friendly profile that highlights your projects and skills.

- ✓ Structure & format review
- ✓ Impact-focused bullet points
- ✓ Course projects showcased
- ✓ Keywords for your target roles

ATS-Ready

Projects

Rewrite

After course completion · live with a placement trainer

STEP 3 OF 6 · PROFILE MAKEOVER

LinkedIn Optimization

Headline, About, skills and featured work tuned so recruiters can find you and quickly see what you can do.

- ✓ Headline & About section
- ✓ Skills & endorsements plan
- ✓ Featured projects & certificates
- ✓ Networking & outreach tips

Discoverable

Skills

Network

After course completion · live with a placement trainer

STEP 4 OF 6 · INDUSTRY-LED

Career-Led Workshop

A practical workshop on job search strategy, applications, interviews and how hiring works in your domain.

- ✓ Finding the right openings
- ✓ Applications that get replies
- ✓ Interview rounds explained
- ✓ Offer & joining essentials

Job Search

Live

Hiring Process

After course completion · live with a placement trainer

STEP 5 OF 6 · MENTOR-LED

Career-Led Mentorship

Personal guidance from an industry mentor on roles, growth paths and your next career moves.

- ✓ Role & domain fit discussion
- ✓ Personal career roadmap
- ✓ Skill gaps to close next
- ✓ Answers to your career questions

1:1 Mentor

Career Path

Growth

After course completion · live with a placement trainer

STEP 6 OF 6 · COMMUNICATION

Industrial Communication Training

Workplace communication that employers expect: emails, meetings, presentations and professional etiquette.

- ✓ Professional emails & messages
- ✓ Meetings & stand-up updates
- ✓ Presenting your work clearly
- ✓ Workplace etiquette & teamwork

Emails

Speaking

Workplace

After course completion · live with a placement trainer

Placement Training Completion Certificate

Complete all six sessions to receive a verifiable certificate listing every training you finished - share it with recruiters along with your course certificate.

08 · SUBJECT MATTER EXPERTS



Meet Our SME Board Members

Expert professionals guiding your journey through real-world industry knowledge and practical experience.

**SME BOARD MEMBER · SENIOR SME**

Harshavardhan Kumar

Lead Consultant

Intern Stump Solutions · Subject Matter Expert

Computer Science professional specializing in AI, software development, education, and UI/UX consulting.

Harshavardhan Kumar Jajam is a technology professional, educator, and Lead UI/UX Consultant at Intern Stump Solutions, specializing in Artificial Intelligence, Python, Data Science, Full-Stack Development, cloud computing, and industry-oriented technical education.

09 · TOOLS AND TECHNIQUES



Key Tools and Techniques

Master industry-standard methodologies used by top-tier professionals, gaining practical exposure to tools, frameworks, and strategies for results.

INDUSTRY STANDARD

01

Selenium WebDriver

Automates browser actions to test web applications across browsers.

CORE FRAMEWORK

02

TestNG

Java testing framework for organising, grouping, parameterising and reporting tests.

API ESSENTIAL

03

Postman

Designs, runs and automates API requests with assertions and collections.

API AUTOMATION

04

REST Assured

Java library for writing readable automated tests for REST services.

WIDELY USED

05

JIRA

Tracks bugs, tasks and sprints the way QA teams do in industry.

BDD

06

Cucumber BDD

Writes human-readable Gherkin scenarios that link business requirements to automated tests.

BUILD TOOL

07

Maven

Manages project dependencies and builds for Java automation frameworks.

MUST KNOW

08

Git and GitHub

Versions your test code and enables collaboration and portfolio sharing.

CI/CD

09

Jenkins

Runs automated test suites on schedule or on every code change in a CI pipeline.

Many More Tools Integrated in Curriculum

10 · COURSE PROJECTS



Projects You Will Work On

Build a strong portfolio by working on industry-relevant projects that simulate real job scenarios, helping you gain confidence and career-ready skills.

INDUSTRY-LED PROJECT

01

E-commerce Manual Test Suite

Write test scenarios, test cases and a traceability matrix for a shopping website. Log defects in JIRA with clear steps and severity.

Duration: 6-10 hours

INDUSTRY-LED PROJECT

02

Banking App Defect Tracking

Test a demo banking application and report bugs through a full defect life cycle. Prepare a summary report for stakeholders.

Duration: 6-10 hours

INDUSTRY-LED PROJECT

03

SQL Data Validation Exercise

Write SQL queries to verify backend data against UI results for a sample order system. Identify mismatches and document them.

Duration: 6-8 hours

INDUSTRY-LED PROJECT

04

Selenium Login and Form Automation

Automate login, registration and form validation flows using Selenium WebDriver with Java. Handle waits, alerts and dropdowns reliably.

Duration: 8-12 hours

INDUSTRY-LED PROJECT

05

Page Object Model Framework

Build a reusable TestNG framework using the Page Object Model and Maven. Add data-driven testing with Excel and property files.

Duration: 10-15 hours

INDUSTRY-LED PROJECT

06

REST API Testing with Postman

Create Postman collections with assertions, environments and variables for a public REST API. Run them with Newman and export reports.

Duration: 6-10 hours

INDUSTRY-LED PROJECT

07

REST Assured API Automation

Automate CRUD API tests in Java using REST Assured and validate JSON responses. Integrate the tests into the TestNG suite.

Duration: 8-12 hours

INDUSTRY-LED PROJECT

08

BDD Cucumber Test Suite

Write Gherkin feature files and step definitions for an e-commerce checkout flow. Generate readable reports for business teams.

Duration: 8-12 hours

INDUSTRY-LED PROJECT

09

Jenkins CI Pipeline for Tests

Push the framework to GitHub and configure a Jenkins job to run tests on every commit. Publish HTML reports after each build.

Duration: 8-12 hours



11 · COURSE ENROLLMENT

Software Testing & Automation

Learn manual and automation testing with Selenium, API testing, Postman, Java and CI/CD in live mentor-led sessions. Build a job-ready portfolio with real projects and interview preparation.

Starts	To be announced	Enrolment closes	Batch details soon
Duration	4 Months	Timings	To be announced
Batch	Batch details updating soon	Batch status	Batch open · Batch dates announced soon

Your plan

One-time fee · inclusive of GST

Plan 1

Course + Live Work Experience

Build real projects and earn a live work experience certificate.

₹10,999

₹45,000 · 76% OFF

Live classes & completion certificate	✓ Included
---------------------------------------	------------

Live projects	✓ Included
---------------	------------

Live work experience certificate	✓ Included
----------------------------------	------------

8C appraisal report	✓ Included
---------------------	------------

SME AI (Insyrio)	✓ Included
------------------	------------

Live chat with SME	✗ Not included
--------------------	----------------

Placement assistance	✗ Not included
----------------------	----------------

Plan perks	15% off across Intern Stump - coins, self-paced courses, masterclasses, mock interviews 20% off membership plans
------------	---

Also included

- Live classes with industry SMEs
- Course completion certificate
- Live projects reviewed by your SME

Good to know

Completion certificate

Verifiable ID - a sample certificate can be previewed on the course page.

8C Appraisal System

How the eight Cs work - see the 8C Appraisal section of this brochure.

Part-pay with iSure Coins

Use your coin balance at checkout.

Need help choosing?

See the FAQs or book a free demo.



12 · 8C APPRAISAL

World's First Ed-Tech with 8C Appraisal

Verified 8C Appraisal Certificate: every candidate receives an official corporate readiness report shared directly with hiring partners upon course completion.

The 8C Appraisal is a proprietary evaluation matrix measuring holistic performance across 8 career pillars for authentic proof-of-work:

Communication

Configuration

Chronicle

Completeness

Cognizance

Conception

Comprehension

Case Study



Frequently Asked Questions

Find answers to common questions about our learning tracks, placement assistance, and 8C evaluation model, or connect directly with an academic consultant.

Q1

Do I need coding knowledge to join Software Testing & Automation?

No. The course starts from the basics and teaches Core Java from scratch, so beginners can follow comfortably with regular practice.

Q2

What is the duration of the course?

The course runs for 4 Months, with live mentor-led sessions plus assignments and project work.

Q3

When does the next batch start?

The upcoming batch is the next batch and it starts on soon. Sessions are held at to be announced.

Q4

What is the course fee?

The current fee is ₹10,999, against the actual fee of ₹45,000, which gives you a discount of 76%.

Q5

Will I work on real projects?

Yes. You complete nine hands-on projects covering manual testing, Selenium frameworks, API testing, BDD and CI pipelines, which you can add to your portfolio.

Q6

Are the classes live or recorded?

Classes are live and mentor-led so you can ask questions and get feedback. Doubt-clearing support helps you catch up if you miss a session.

Q7

Is this course suitable for manual testers?

Yes. Manual testers can use it to move into automation, with a strong focus on Java, Selenium, API automation and framework design.

Q8

Will I get career support after the course?

Yes. As part of the Technical programme you get resume guidance, mock interviews and QA interview preparation to help you apply with confidence.

Ready to take the next step?

Software Testing & Automation · Live Course



Scan to view details & enrol

internstump.com/courses/software-testing-automation



WEBSITE
internstump.com



EMAIL
info@internstump.com



CALL / WHATSAPP
+91 7396168247



Intern Stump Solutions · Generated on 25 Oct 2026, 05:07 PM IST from the live details on internstump.com. Prices, batches and schedules can change.