STLabsIndia
SOFTWARE TRAINING INSITITUTE

Full Stack Software Testing Training Program

"From Noida's premier software training institute - a course built to launch your tech career."

Ministry of MSME, Govt. of India

About STLabs

STLabs is a reputed technology training institute committed to delivering high-quality, industry-oriented education in software testing and development. With over 10 years of consistent excellence, STLabs has built a strong reputation for its dedication to student success and career growth.

We focus on nurturing our students through practical learning, expert mentorship, and continuous guidance, ensuring they are well-prepared for real-world challenges. Our commitment goes beyond training—we actively support each student in achieving their career goals by providing strong placement assistance, backed by an impressive track record of successful placements.

At STLabs, every effort is made to empower students with the skills, confidence, and opportunities needed to build a successful future.

Job-Oriented Software Testing Training Program

with GEN AI

Offered by STLabs India

Industry-Relevant Job Oriented Software Testing Courses to Get You Hired!

BHIM G Pay Paytm PhonePe VISA MasterCard

Address: Amba Tower C-58, Sector 2, Noida, U.P. **INDIA:** 0120-4850016 || **USA:** +1-(762) 332-9766

STLabsIndia
SOFTWARE TRAINING INSITITUTE

STLABS INDIA 10-years legacy of success

Empowering coders to achieve their dreams

- **20K+ Learners**
  cracked dream roles at top tech companies
- **5k+ Alumni in MNC**
  & more in 103/111 Unicorne
- **Regular Assignments**
  after completing the modules
- **Interview Preparation**
  via our placement cell
- **Expert Mentorship**
  cracked dream roles at top tech companies
- **Flexible Learning**
  via our placement cell
- **Career Guidance**
  cracked dream roles at top tech companies
- **Flexible Learning**
  via our placement cell
- **Career Guidance**
  cracked dream roles at top tech companies
- **Placement Assistance**
  via our placement cell

BHIM G Pay Paytm PhonePe VISA MasterCard

Tools & Technologies

Software Testing relies on a powerful ecosystem of tools and technologies that ensure the quality, functionality, and performance of software applications. At the core of this ecosystem is Selenium and Java, the most widely used tools in the industry: Selenium is a powerful open-source automation framework, ideal for automating web applications across different browsers, while Java, known for its versatility and reliability, is the preferred programming language for building robust and scalable test scripts. Together, these tools enable testers to automate repetitive tasks, execute test cases, and generate detailed reports with efficiency with efficiency.

Let 's take a closer look at these two essential tools in software testing:

Defect Management Tools & Agile Process

Selenium

A powerful open-source tool for automating web application testing across different browsers.

A widely used programming tool for planning, tracking, and reporting test execution.

Defect Management Tools & Agile Process

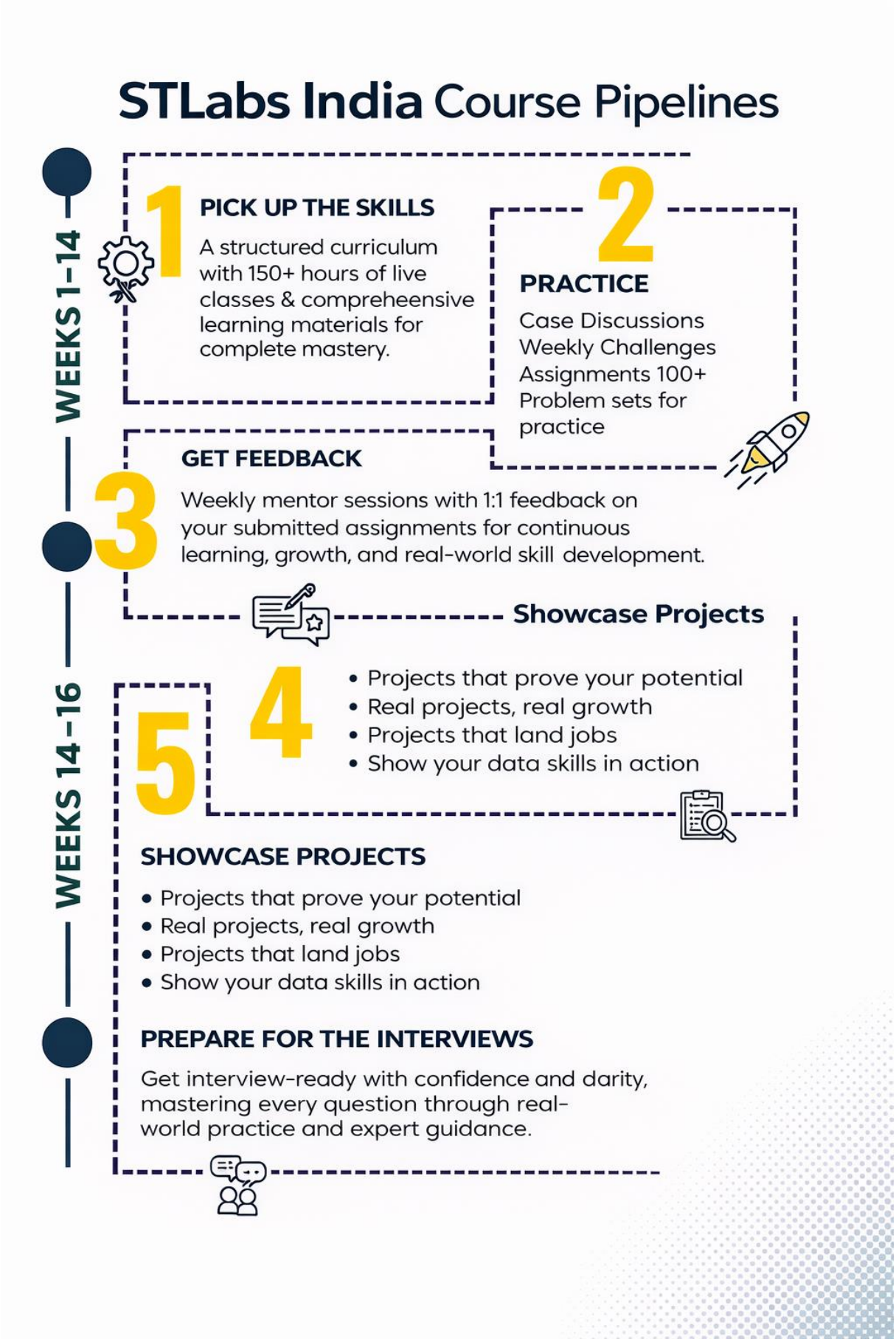
Jira

A widely used tool for bug and issue tracking, fostering colaboration and efficient defect mgnt,

Zephyr

An advanced test management tool for planning, tracking, and reporting test execution.

Agile/Scrum Process
A flexible and iterative approach to software development emphaszing teamwork-sprints, and con-tinuous improvement.


STLabs India Course Pipelines
WEEKS 1–14
1
PICK UP THE SKILLS
A structured curriculum with 150+ hours of live classes & compreheensive learning materials for complete mastery.
2
PRACTICE
Case Discussions
Weekly Challenges
Assignments 100+
Problem sets for practice
3
GET FEEDBACK
Weekly mentor sessions with 1:1 feedback on your submitted assignments for continuous learning, growth, and real-world skill development.
4
Showcase Projects
• Projects that prove your potential
• Real projects, real growth
• Projects that land jobs
• Show your data skills in action
WEEKS 14–16
5
SHOWCASE PROJECTS
• Projects that prove your potential
• Real projects, real growth
• Projects that land jobs
• Show your data skills in action
PREPARE FOR THE INTERVIEWS
Get interview-ready with confidence and darity, mastering every question through real-world practice and expert guidance.

STLabsIndia
SOFTWARE TRAINING INSITITUTE

Course Curriculum

Course Overview	*This comprehensive **Full Stack Software Testing program** is designed to equip learners with end-to-end testing skills, covering manual testing, automation testing, programming, tools, and real-world workflows.* *The course focuses on hands-on learning, real-time projects, and industry-relevant practices to make you job-ready.* ***Course Highlights:*** ✓*Live instructor-led sessions* ✓*100% practical training* ✓*Real-time projects & case studies* ✓*Industry tools & frameworks* ✓*Resume & interview preparation* ✓*Placement assistance*
Introduction to Software Testing	• *What is Software?* • *What is Software Testing?* • *Client-Server Architecture and Types of software applications.* • *Reason of Defects in Software Application.* • *QA vs QC vs Testing* • *AI-Powered Testing Tools Overview*
SDLC & STLC	• *Phases of SDLC* • *Models: Waterfall, Agile, V-Model* • *STLC phases and deliverables* • *Entry & Exit criteria*
Testing Types	• *Deployment Process* • *Dev, QA, SIT, UAT and Production Environments* • *Types of Testing:* ✓ *Manual & Automation* • *Functional vs Non-functional testing* • *Black Box Vs White Box testing* • *Deployment, Smoke, Sanity, Retesting, Regression, Exploratory,*

	Integration, Cross browser, Compatibility, Installation, Adhoc, Monkey and other testing • *UAT (User Acceptance Testing)*
Test Design Techniques	• *Test Scenarios & Test Cases* • *Boundary Value Analysis* • *Equivalence Partitioning* • *Decision Table Testing*
Defect Management	• *What is Bug/Defect?* • *Bug Life Cycle* • *Severity vs Priority* • *Writing effective bug reports* • *Defect tracking process* • *Common defect types*
Documentation & Test Artifacts	• *Test Plan* • *Test Strategy* • *Test Case & Test Scenario documents* • *Requirement Traceability Matrix (RTM)* • *Test Summary Report*
Introduction to Java | JDK & Eclipse Installation	• *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* • *Writing the First Java Program* • *Compiling and Running Java Programs* • *Understanding Console Output*
Working with Java Variables & Data Types	• *Variables & Naming Conventions* • *Data Types* • *Declaring Variables in Java*

	• *Primitive Data Types:*
	  ✓ *To Store Numbers: int, byte, short, long*
	  ✓ *To Store Decimal numbers: float, double*
	  ✓ *To Store Single Character: char*
	  ✓ *To Store True and False: Boolean*
	• *Non Primitive Data Types:*
	  ✓ *Strings*
	  ✓ *Arrays*
	  ✓ *Classes & Objects*
	  ✓ *Interfaces*
	  ✓ *Collections*
	  ▪ *Lists (ArrayList, LinkedList)*
	  ▪ *Sets (HashSet, TreeSet)*
	  ▪ *Maps (HashMaps)*
	  ✓ *Enums (Enumerations)*

Working with Java Operators & Expressions	• *What are Operators in Java?*
	• *Binary Operators:*
	  ✓ *Arithmetic Operators (+, -, *, /, %)*
	  ✓ *Relational/Comparison Operators (>, >=, <, <=, !=, ==)*
	  ✓ *Logical Operators (&&, ||, !)*
	  ✓ *Increment/Decrement Operators (++, --)*
	  ✓ *Assignment Operators (=, +=, -=, *=, /=, %=)*
	• *Ternary Operators:*
	  ✓ *Conditional/Ternary Operator (?)*

Working with Java Conditional Statements	• *If Statement → one condition* • *If-else statement → either-or* • *If-else-if Ladder → multiple conditions* • *Nested if Statement → condition inside condition* • *Switch Statement → multiple fixed options*

Working with Loops and Jump Statements	• *Loop Statements:*
	  ✓ *For loop → fixed iterations*
	  ✓ *While loop → condition-based, may run 0 times*
	  ✓ *Do-while loop → condition-based, runs at least once*
	  ✓ *For-each loop → best for arrays/collections*
	• *Loop Control Statements:*
	  ✓ *break → Exit the loop immediately.*
	  ✓ *continue → Skip the current iteration and move to the next.*

	✓ *Return → Exit method (optionally return value)*
Working with Arrays | Single and Two Dimensional	• *Introduction to Arrays* • *Declaring and Initializing Arrays* • *Accessing Array Elements* • *Traversing Arrays* • *Array Properties* • *Multidimensional Arrays* • *Array Methods (via java.util.Arrays Class)* • *Common Operations* ✓ *Find largest/smallest element* ✓ *Reverse array* • *Limitations of Arrays*
Working with Java Strings	• *Introduction* • *Declaring Strings* ✓ *Using string literal (stored in String pool)* ✓ *Using new keyword (stored in heap)* • *String Concatenation* • *String Methods (Commonly Used)* ✓ *Length ()* ✓ *CharAt ()* ✓ *SubString ()* ✓ *Contains ()* ✓ *Equals () and EqualsIgnoreCase ()* ✓ *Case Conversion (toLowerCase (), toUpperCase ())* ✓ *Replace ()* ✓ *Trim ()* ✓ *Split ()* • *StringBuilder and StringBuffer*
Working with Strings- Coding Practice	• *Common String Programs (Class Exercises)* ✓ *Reverse a string (toCharacterArray, charAt, StringBuffer, StringBuilder)* ✓ *Check if a string is a palindrome* ✓ *Count vowels and consonants* ✓ *Find frequency of characters* ✓ *Remove duplicates from a string* ✓ *String Comparison (== and equals)*

Java OOPs Concepts- Class, Objects and Methods	• *Object Oriented Programming* ✓ *Class* ✓ *Object* ✓ *Polymorphism* ✓ *Encapsulation* ✓ *Inheritance* ✓ *Data Abstraction*
Java OOPs Concepts- Java Methods & Constructors	• *Methods* ✓ *No Parameters- No Return Value* ✓ *No Parameters- Return Value* ✓ *Takes Parameters- No Return Value* ✓ *Takes Parameters- Return Value* • *How many ways data can be stored in a Variable:* ✓ *Using reference variable* ✓ *Method* ✓ *Constructor* ▪ *Default Constructor* ▪ *Parameterized Constructor*
Java OOPs Concepts- Polymorphism & Encapsulation- Overloading	• *Polymorphism* • *Rules of Polymorphism* • *Encapsulation* • *Rules of Encapsulation* • *Can we overload main () method?*
Java OOPs Concepts- Use of This and Static Keyword in Java	• *Types of Variables:* ✓ *Class Variables/Instance Variable* ✓ *Local Variables* ✓ *Rules of Static Variable and Method*
Java OOPs Concepts- Inheritance and Types of Inheritance	• *Inheritance* • *Objective of Inheritance (Reusability and Avoid Duplication)* • *Types of Inheritance:* ✓ *Single* ✓ *Multi-level* ✓ *Hierarchy*

	✓ *Multiple*
Java OOPs Concepts- Method Overriding, Final & Super Keywords	• *Method Overriding* • *Method Overloading* • *Final- Variables, Methods, Class* • *Usage of Super Keywords*
Data Conversion, Packages, Access Modifiers Interface	• *Wrapper Class* • *Data Conversion* • *Packages (Built in/User Defined)* • *Access Modifiers* ✓ *Public* ✓ *Protected* ✓ *Default* ✓ *Private*
Java- Exception	• *Checked Exception- Which are identified by Java Complier* ✓ *FileNotFound* ✓ *IOException etc..* • *Unchecked Exception- Which are not identified by Java Complier* ✓ *ArithMaticExcepton* ✓ *NullPointerException* ✓ *ArrayIndexOutOfBoundsException etc..* ✓ *Try..catch* ✓ *Try..catch..finally* ✓ *Throws*
Selenium with Java- Introduction & Environment Setup	• *Selenium WebDriver* • *Environment Setup: Downloading jars and configuring them with java project (Manually)* • *Creating Maven Project* • *Practical Assignment:* ✓ *Launch Browser* ✓ *Open URL* ✓ *Validate Title* ✓ *Close Page*

Selenium with Java- Locators/Object Identifiers	• *What are locators and how selenium interacts with locators?* ✓ *ClassName, CssSelector, Id, LinkText, Name, PartialLinkText, TagName, Xpath* • *FindElement ()* • *FindElements ()*
Selenium with Java- WebDriver Methods	• *Introduction to WebDriver methods in Selenium* • *WebDriver Methods Overview* ✓ *Get Methods* ✓ *Conditional Methods* ✓ *Browser Methods* ✓ *Navigational Methods* ✓ *Wait Methods* • *Understanding the usage of get window handles method in Selenium with Java* • *Handling synchronization in Selenium WebDriver* • *Understanding how to switch between browser windows using window ID* • *WebDriver Methods for Web Elements* • *WebDriver Methods: isDisplayed, isEnabled* • *WebDriver method is enabled checks if element is enabled* • *Using WebDriver's select method to interact with radio buttons* • *Selecting and checking status of radio buttons using Selenium WebDriver in Java* • *Difference between close and quit commands in Selenium WebDriver* • *Difference between close and quit command*
Selenium with Java- WebDriver Wait Methods	• *Synchronization* • *Thread. Sleep ()* • *Implicit Wait ()* • *Explicit Wait/Fluent Wait ()* • *NoSuchElementException and ElementNotFoundException*
Selenium with Java- Browser and Navigation Commands	• *Navigational Commands:* ✓ *Navigate().to()* ✓ *Navigate().back()* ✓ *Navigate().forward()* ✓ *Navigate().refresh()*

	• *Conditional Commands:* ✓ *isDisplayed()* ✓ *isEnabled()* ✓ *isSelected()* • *Get Commands:* ✓ *get ()* ✓ *getTitle ()* ✓ *getCurrentURL ()* ✓ *getPageSource ()* ✓ *getWindowHandle ()* ✓ *getWindowHandles ()* • *close () and quit ()* • *Assignment*
Selenium with Java- Handling Check Boxes and Alerts	• *Handling checkboxes in Selenium* • *Selective deselection of checkboxes in Selenium* • *Handling different types of popups and alerts in Selenium*
Selenium with Java- Handling Frames & Nested iFrames	• *What is a Frame or IFrame?* • *Why Switch to a Frame in Selenium?* • *Switching to a Frame - Methods* ✓ *driver.switchTo().frame(int index)* ✓ *driver.switchTo().frame(String nameOrId)* ✓ *driver.switchTo().frame(WebElement)* ✓ *driver.switchTo().parentFrame();* ✓ *driver.switchTo().defaultContent();* • *Handling Nested Frames (Frames inside Frames)* • *Frame Count*
Handling Different Types of Drop-downs	• *What is a Dropdown in HTML?* • *Selenium Support for Dropdowns* • *Creating a Select Object* • *Selecting Options* ✓ *selectByVisibleText()* ✓ *selectByValue() (from HTML)* ✓ *selectByIndex() (0-based)* ✓ *select.isMultiple();* • *Deselecting Options*

	✓ *select.deselectByVisibleText("India");* ✓ *select.deselectAll();*
Handling Dynamic Pagination WebTable	• *What Are Dynamic Web Tables?* • *What Is Pagination in Tables?* • *Challenges with Dynamic Tables + Pagination* ✓ *Table content changes per page* ✓ *Dynamic row/column count* ✓ *Data loading time after page click* • *Assignment*
Mouse Actions, Action Vs Actions	• *What Are Mouse Actions in Selenium?* ✓ *click()* *Left-click* ✓ *doubleClick()* *Double-click* ✓ *contextClick()* *Right-click (context menu)* ✓ *moveToElement()* *Hover (mouse-over)* ✓ *dragAndDrop()* *Drag and drop* ✓ *clickAndHold()* *Press and hold mouse* ✓ *release()* *Release the hold* • *What Is the Actions Class?* • *What is the Action Interface?* • *Difference: Action vs Actions* • *What Are Mouse Actions in Selenium?* ✓ *actions.sendKeys(Keys.ENTER)* *Send a single key* ✓ *actions.sendKeys("username")* *Send text* ✓ *actions.keyDown(Keys.CONTROL).sendKeys("a")* ✓ *keyUp(Keys.CONTROL)* *Press combo keys*
JavaScriptExecutor, Scrolling Pages and Upload Files	• *Why JavaScriptExecutor in Selenium: Selenium doesn't support all browser features directly. JavaScriptExecutor lets you run raw JavaScript in the browser to overcome those limitations.* • *How to Use JavaScriptExecutor?* • *Scroll to Bottom of Page* • *Scroll to a WebElement* • *Scroll down by pixels* • *Click using JS (if normal .click() fails)* • *Set value of hidden input* • *File Upload* • *Assignment*

Working with Excel	• *Introduction to Data Driven Testing* ✓ *What is DDT?* ✓ *Why use Excel?* ✓ *Test logic vs. test data separation* • *Introduction to Apache POI* ✓ *What is Apache POI?* ✓ *Working with XLS vs XLSX* ✓ *Basic structure: Workbook → Sheet → Row → Cell* • *Reading from Excel* ✓ *Handling data types (strings, numbers, blanks)* ✓ *Reading multiple rows and columns* • *Integrating with TestNG or JUnit* ✓ *Data Providers (@DataProvider)* ✓ *Running tests with different inputs* • *Best Practices* ✓ *Keep Excel data clean* ✓ *Don't hardcode paths* ✓ *Log errors if cell is empty or invalid*
TestNG Integration & Setup	• *Introduction to TestNG* • *Installing & Setting Up TestNG* • *Writing Your First TestNG Test* • *TestNG Annotations* • *Running TestNG Tests* • *Integrating TestNG with Selenium* • *Grouping, Prioritization, and Dependencies* • *Parallel Test Execution (Optional for Advanced Class)* • *TestNG Listeners and Generating Test Reports (Extent Report)*
Page Object Model (POM)	• *Introduction to Page Object Model (POM)* • *Why Use POM?* ✓ *Code Reusability* ✓ *Easy Maintenance* ✓ *Better Readability* ✓ *Scalability* ✓ *Cleaner TestNG Tests* • *Basic Structure of a POM Framework* • *Test Case with TestNG Using POM* • *Adding a Driver Factory (Reusable Setup)* • *Using PageFactory*

	• *Integration with testng.xml*
GitHub	• *Introduction to Git & GitHub* • *Setting Up Repositories* • *Cloning & Managing Projects* • *Commit, Push & Pull Operations* • *Branching & Merging Techniques* • *Resolving Merge Conflicts* • *Collaborating with Teams on GitHub* • *Using GitHub for Project Version Control* • *Hosting & Showcasing Data Science Projects*

Interview Preparation

Our dedicated placement teams have a decade-plus of experience transitioning students from classroom to workplace. Here's how it goes:

Resume Preparation Session

Our mentors will guide you through preparing your personalized resumes connecting your talents, skills, previous experience, etc.

LinkedIn Profile Refinement

Linkedin is the platform where you get the tickets to your dream job, and we help you optimize your profile there in the finest way so you bag the best salary in the market.

Interview Preparatory Sessions

We provide intensive interview preparation right from Day 1 to prepare candidates for interviews with our network of 2000+ hiring partners.

Interview Preparatory Sessions

We provide intensive interview preparation right from Day 1 to prepare candidates for interviews with our network of 2000+ hiring partners.

Mock Interviews

A panel of industry experts will conduct a SWOT analysis and present you with the feedback and ratings of how you performed in the interview and the areas you need to improve.

Hands on Experience

Why **STLabs India?**

Top-Notch Faculty Our faculty is our strength. We have certified software testing trainers, with over 12 years of experience, who have a strong passion for training. The trainers are always at your aisposal to assist you throughout the learning process.

Industry-Based Course Curriculum Considering the unmet skill gaps in industry, our course modules are aligned to the latest technological trends, and cover all the essential topics and tools.

Dedicated Post-Training Support Learners gain hands-on experience when they start on their assignments and case studies. We provide dedicated mentors to help with any queries during this phase.

Job Assistance Program Our software testing training doesn't stop with theory and practice. We also take responsibility for making you job-ready. Our placement support is included with the course.

Google ★★★★★

Job Star Reviews on Google We have had many happy learners over theory and practice. We also take responsibility for making you job-ready. Our placement support is included with the course.

5/5 Star Reviews on Google We have had many happy learners over the years, and our ratings reflect our ethos of striving to always provide a high-quality learning experience.

Top Hiring Companies
Synechron
EXL
DXC TECHNOLOGY
OPTUM
BDO
amazon
CAM COM
Coforge
WNS
VNG
paloalto
Birlasoft
Deloitte.
salesforce
vmware
Tech Mahindra
BDO
amazon
DXC TECHNOLOGY
Optum
WNS
VNG
CAM COM
vmware
Deloitte.
salesforce
accenture
MetLife
FIS
MetLife
BLACK LIGHT
wipro
Deloitte.
salea
NUCLEUS SOFTWARE
NUCLEUS SOFTWARE