TechyEdz Solutions

Training | Consulting | Developement | Outsourcing

Software Testing

Software Testing

➕ **Learning Path:**

1. **Core Java basics**
2. **Manual Testing**
3. **Automation Testing**
   - I. **QTP**
   - II. **Selenium**

➕ **Course Overview:**

Software Testing Tools program is designed for any graduate with any stream. It is process oriented in Software Development Life Cycle. Through Software Testing Life Cycle, a Quality Analyst, analyze the software and providing assurance to client/end user.

STLC mainly focuses on Understanding Requirements, Test Planning, Test Design (Scenarios and Test Cases), Test Execution, Result Analysis, Defect Tracking and Reporting and etc. In Contemporary IT industry Quality is playing a major role.

In Testing Industry Test Automation Tools are playing major role, with the help of third party tools tester can able to perform retesting and regression testing. There are several popular tools in **QA Industry** like **UFT, QTP, Quality Center, LoadRunner, SoapUI, Selenium etc...**

➕ **Course Outline:**

Core Java

1. **Getting Started with Java SE**

   ➢ What is Java?

- ➢ Installing Java
- ➢ The jdk Directory Structure
- ➢ Sdk structures
- ➢ OOPS Concept
- ➢ Java Language
- ➢ Java Virtual Machine
- ➢ World Wide Web and Java
- ➢ Java Platforms

2. First Java Programs

- ➢ Writing your first Java program "Hello, World", your first small step towards learning Java
- ➢ Program Structure
- ➢ Output in Java
- ➢ Variables and Expressions

3. Data types and Variables

- ➢ Primitive Data types
- ➢ Variable Names
- ➢ Numeric Literals
- ➢ Character Literals
- ➢ String
- ➢ String Literals
- ➢ Arrays

4. Introduction to Objects

- ➢ Object Models
- ➢ Classes and Objects
- ➢ Abstract methods and Classes
- ➢ Input in Java
- ➢ Input Wrapper Class
- ➢ Packages

5. Data Types and Operators

- ➢ Strong Typing

- ➢ Character Codes
- ➢ ASCII and Unicode
- ➢ String Class
- ➢ String Input and Output
- ➢ String Methods

12. Modular Programming

- ➢ Monolithic Programs
- ➢ Static Variables and Methods
- ➢ Functional Modularity
- ➢ Object Modularity
- ➢ Top-Down and Bottom-Up Development
- ➢ Pass-By-Value and Pass-By-Reference
- ➢ Nested Classes

13. Exception Handling and More Flow Control

- ➢ Exceptions Overview
- ➢ Exceptions
- ➢ Declaring Exceptions
- ➢ Defining and Throwing Exceptions
- ➢ Errors and Runtime Exceptions
- ➢ Catching Exceptions
- ➢ The finally Block
- ➢ Exception Methods
- ➢ I/O Exceptions vs Runtime Exceptions

14. Input/Output Streams

- ➢ Overview of Streams
- ➢ Bytes VS Characters
- ➢ Converting Byte Streams to Character Streams
- ➢ File Object
- ➢ Binary Input and Output
- ➢ Print Writer Class
- ➢ Reading and Writing Objects

> Basic and Filtered Streams

15. Core Collection Classes

> The Collections Framework
> The Set Interface
> Set Implementation Classes
> The List Interface
> List Implementation Classes
> The Queue Interface
> Queue Implementation Classes
> Implementing a Stack
> The Map Interface
> Map Implementation Classes

16. Collection Sorting and Tuning

> New Features in JSE 6
> Changing in I/O(JSE 6)
> Using Java 6.0 Features with Collections
> Sorting with Comparable
> Sorting with Comparator
> Sorting Lists and Arrays
> Collections Utility Methods
> Tuning Array List
> Navigable Map and Navigable Set
> Tuning Hash Map and Hash Set

17. Inner Classes

> Inner Classes
> Member Classes
> Local Classes
> Anonymous Classes
> Instance Initializes
> Static Nested Classes

18. Introduction to Threads

- ➢ Overview of thread
- ➢ Life Cycle of thread
- ➢ Creating Threads
- ➢ Multithreading
- ➢ Deadlock
- ➢ Inter-Thread Communication
- ➢ Thread States
- ➢ Runnable Threads
- ➢ Coordinating Threads
- ➢ Interrupting Threads
- ➢ Runnable Interface
- ➢ Thread Groups

Manual Testing

1. Introduction

- ➢ Importance of Software System
- ➢ Common problems in software development
- ➢ Why software has bugs?
- ➢ What is quality?
- ➢ Testing objectives
- ➢ What are errors, bugs, defect?

2. SDLC

- ➢ Feasibility study/Requirement Gathering
- ➢ Analysis
- ➢ Designing
- ➢ Coding
- ➢ Testing
- ➢ Delivery and maintenance
- ➢ Testing Stage in SDLC

3.Software development Models

- ➢ Waterfall model
- ➢ Prototype model
- ➢ v-model

4.Kinds of Testing

- ➢ Un-conventional Tesing
- ➢ Conventional Testing

5.Methods Of Testing

- ➢ Black Box Testing
- ➢ White Box Testing
- ➢ Grey Box Testing

6.Static Testing

- ➢ Reviews
- ➢ Inspections
- ➢ Audit
- ➢ Walkthrough

7.Dynamic Testing

- ➢ Unit Testing
- ➢ Module testing
- ➢ Integration Testing
- ➢ System testing
- ➢ User Acceptance Testing

8.Software Environments

- ➢ Stand alone Environment
- ➢ Client/Server Environment
- ➢ Web Environment
- ➢ Distributed Environment

9.Software Testing Life Cycle and Implementation with project support

- ➢ Requirements Reviews

- ➢ Test Plan Preparation—Real Time Scenerio

- ➢ Test Strategy Preparation

- ➢ Test Case Identification

10. Black Box Testing design Technique

- ➢ ECP

- ➢ BVA

- ➢ Error Guessing

11. White Box testing Techniques

- ➢ Basis path Testing

- ➢ Cyclomatic Testing

- ➢ Condition Testing

- ➢ Code Coverage

12. What is use Case

- ➢ UCd

- ➢ Identify Use case Based Test Cases

13. Test Case design Technique In Project Implementation

- ➢ Introduction

- ➢ Test Case design template

- ➢ Types of Test Cases

- ➢ Test Case design Reviews

- ➢ Requirement Traceability Matrix

14. Test data Setup

- ➢ Importance of Test data in Testing

- ➢ Approach for gathering Test data

- ➢ Benefits of Test data gathering

15. Test Execution

- ➢ Test Execution Life Cycle Process

- ➢ Re-Testing
- ➢ Regression Testing
- ➢ Database Testing

16. Result Analysis

- ➢ Comparing Expected and Actual Values
- ➢ Status of the Result

17. Defect Tracking

- ➢ What is Defect?
- ➢ Categories of defects
- ➢ Preparation of dpd(defect profile document)

18. Test Closure Activity

- ➢ Test Execution Stop Criteria
- ➢ Test Summary Report Preparation

19. User Acceptance Testing

- ➢ Importance Of UAT
- ➢ Scope of UAT
- ➢ Alpha Testing
- ➢ Beta Testing

20. Other Testing Technique

- ➢ Smoke testing
- ➢ Sanity testing
- ➢ Re-testing
- ➢ Regression Testing
- ➢ Static Testing
- ➢ Usability Testing
- ➢ Dynamic testing
- ➢ Compatibility Testing
- ➢ System testing

- ➢ User Interface testing
- ➢ Security Testing
- ➢ Functional Testing
- ➢ Usability Testing
- ➢ End-End Testing
- ➢ Exploration Testing
- ➢ Installation Testing/System Testing.

Quick Test Professionals (QTP)

1. Automation Testing
- ➢ What is Automation Testing
- ➢ Benefits of Automation testing
- ➢ Manual Testing VS Automation Testing
- ➢ Types off Automation Tools
- ➢ When test case can be automated
- ➢ When test cannot be automated
- ➢ Things you need to before automation

2. Introduction to QTP 11.00
- ➢ Overview of Quick Test Pro
- ➢ Starting Quick Test
- ➢ Quick Test Window
- ➢ Quick Test Window Layout
- ➢ Using Quick Test Commands

3. Working with Test Objects and Object Repositories
- ➢ How QTP learns & identifies Objects
- ➢ Working with Test Objects
- ➢ Object Spy
- ➢ Configuring Object Identification
- ➢ Smart Identification
- ➢ Merging Shared Object Repositories
- ➢ Comparing shared objects Repositories
- ➢ Associating shared objects Repositories to a Test

➢ Working with Test objects methods & Properties

4. User Defined Functions
➢ Creating User Defined Functions
➢ Advantages of Using Functions
➢ Creating Generic Functions
➢ Creating Function Libraries
➢ Associating Function Libraries
➢ Preparing Regression Test Automation Suites
➢ Function Decomposition Implementation

5. VB Script Language Overview
➢ VB Script Data Types
➢ VB Script Variables
➢ VB Script Constants
➢ Conditional Statements
➢ Loops
➢ Select case
➢ Arrays
➢ Functions

6. Advanced VB Scripting
➢ Working with Internet Explorer Application
➢ Working with File system Objects
➢ Working with Data Base Connection
➢ Working with Record Set Object
➢ Working with COM Object
➢ Working with Word Application Object
➢ Error handling with Err Object
➢ Working with Dictionary Object
➢ Working with XML Util Object
➢ Working with QC Util Object
➢ Working with DOM Object
➢ WSH Scripting Basics
➢ Working with Shell

7. Working with test object and object Repositories
➢ Object Introduction
➢ How objects are added in to OR

- ➤ Object Repository modes
- ➤ Identify the Objects
- ➤ Pre Action V Shared object repository
- ➤ Viewing the object properties
- ➤ Object SPY
- ➤ OI Introduction
- ➤ User Defined Objects
- ➤ Smart Identification
- ➤ Managing Object Repositories
- ➤ Merging Object Repositories

8. Synchronization

- ➤ What is Synchronization
- ➤ Need of Synchronization
- ➤ Different ways of Synchronizing the Tests
- ➤ Configuring Default Synchronization Time Out
- ➤ Synchronization Methods
- ➤ Adding Wait Property Method
- ➤ Adding Wait Method
- ➤ Adding Exit Method
- ➤ Adding Sync Method
- ➤ Adding Synchronization Setting
- ➤ Browser VS Page Sync
- ➤ Web Event Recording Configuration
- ➤ Synchronizing Point

9. Checkpoints

- ➤ Checkpoint Introduction
- ➤ Adding Checkpoints to a Test
- ➤ Types of Checkpoints
- ➤ Standard Checkpoint
- ➤ Image Checkpoint
- ➤ Bitmap Checkpoint
- ➤ Table Checkpoint
- ➤ Text Checkpoint
- ➤ Text Area Checkpoint
- ➤ Accessibility Checkpoint
- ➤ Page Checkpoint

- ➢ Database Checkpoint
- ➢ XML Checkpoint
- ➢ Checkpoints Supported Environment
- ➢ Validating the Above Checks through

10. Transactions
- ➢ Purpose
- ➢ Inserting Transactions
- ➢ Inserting Ending Transactions
- ➢ Analyzing Transaction Results

11. Working with Actions
- ➢ Introduction to Actions
- ➢ Benefits of Actions
- ➢ Creating Tests with Multiple Actions
- ➢ Guidelines for working with Actions
- ➢ Creating new Actions
- ➢ Insert Actions
- ➢ Inserting Copy of Actions
- ➢ Inserting Existing Actions
- ➢ Splitting Actions
- ➢ Action Parameters
- ➢ Test Modularity Implementation
- ➢ Creating an Action Template
- ➢ Difference Between Reusable Actions and Function
- ➢ Introduction to Parameterize Tests
- ➢ Parameterize a Test Manually
- ➢ Parameterize a Test using Data table
- ➢ Parameterization through XML Files, Excel Files, Flat Files
- ➢ Parameterize a Test using Environment Variables
- ➢ Data Driven Test
- ➢ Create Data Driven Tests
- ➢ Local and Global Data Tables

12. Regular Expressions
- ➢ Introduction to Regular Expressions
- ➢ Defining Regular Expressions
- ➢ Working with RegExp Object

13. Debugging

- ➢ Need of Debugging
- ➢ Break Points
- ➢ Step Into
- ➢ Step Over
- ➢ Step Out
- ➢ Debug Viewer
- ➢ Watch
- ➢ Variable
- ➢ Command

14. Recovery Scenarios

- ➢ Introduction to Recovery Scenarios
- ➢ When to use Recovery Scenarios
- ➢ Defining Recovery Scenarios
- ➢ Recovery Scenario Manager
- ➢ Managing Recovery Scenarios
- ➢ Associating Recovery Scenarios to your Test
- ➢ Programmatically Controlling the Recovery Mechanism

15. Descriptive Programming

- ➢ Introduction to Descriptive Programming
- ➢ When to use Descriptive Programming
- ➢ Descriptive Programming Approach
- ➢ Static Representation
- ➢ Dynamic Representation
- ➢ Handling Dynamic Objects
- ➢ Handling Similar Objects
- ➢ Developing Generic Scripts

16. Virtual Objects

- ➢ Define Virtual Object
- ➢ Manage Virtual Objects

17. Working with WebTables AOM Scripts

- ➢ Developing Start Up Scripts
- ➢ Launching QTP
- ➢ Setting Tool Options
- ➢ Setting Test Options

- ➢ Opening a Test
- ➢ Running a Test
- ➢ Store Results in Required Location
- ➢ Closing a Test
- ➢ Closing QTP

18. API Programming
- ➢ API Introduction
- ➢ Windows API Library
- ➢ Registering API Library
- ➢ Calling the Library Functions in API

19. Scripting in Real-time Environment
- ➢ Quick Test Pro Coding Standards &BestPractices
- ➢ Naming Conventions
- ➢ Coding Rules

20. Automation Frameworks
- ➢ Approach
- ➢ Planning and Design Frame Work
- ➢ Library Architecture Framework
- ➢ Keyword-Driven Framework
- ➢ Data-Driven Framework
- ➢ Hybrid Framework

21. Working with Quality Center
- ➢ Connecting to Quality Center
- ➢ Disconnecting from Quality Center
- ➢ Integrating Quickest with Quality Center
- ➢ Uploading Automation Resources in Quality Center
- ➢ Saving Tests into Quality Center
- ➢ Opening Tests from Quality Center
- ➢ Executing Tests from Quality Center

22. Database Verification
- ➢ Introduction
- ➢ Database Checkpoint
- ➢ Database Output Value
- ➢ Database Connectivity

23. Run-Time Error Handling
- ➤ Recovery Scenario Manager
- ➤ Err Object
- ➤ Customized Error Handling
- ➤ On Error Statement

24. Additional Coverage
- ➤ Utility Objects
- ➤ Step Generator
- ➤ Function Definition Generator
- ➤ Examples & Scenarios Relevant to Major Businesses

Selenium

Introduction:
- ➤ Introduction of selenium
- ➤ Selenium components
- ➤ How differs from other automation tools
- ➤ Advantages of selenium
- ➤ Overview of the Testing framework
- ➤ Overview of Eclipse

Selenium Architecture
- ➤ Selenium RC Architecture
- ➤ Webdriver Architecture
- ➤ Webdriver Vs Selenium RC
- ➤ A brief explanation about the advantages of webdriver

Locator Techniques
- ➤ Introduction about locator concept
- ➤ A brief explanation of different locator techniques
- ➤ Xpath techniques explanation with different kind of real-time scenarios

Selenium IDE
- ➤ Download and Installation
- ➤ Record and playback techniques

- ➢ Modifying the script using IDE
- ➢ Convert the IDE script into RC script
- ➢ Validate the locator value using IDE

Selenium setup
- ➢ Explanation of step by step instructions for setup
- ➢ Integration of eclipse and TestNG

Selenium RC
- ➢ Start the selenium Selenium server
- ➢ Basics of Annotations in Testing
- ➢ Write the basic script of Selenium RC
- ➢ How to execute the scripts (different ways) and look at the results
- ➢ A brief explanation of commands in selenium RC

WebDriver
- ➢ Webdriver setup
- ➢ A detailed discussion about web driver commands
- ➢ Handling different browsers
- ➢ Create our own methods in webdriver
- ➢ More programming techniques in webdriver
- ➢ Using RC commands from webdriver project
- ➢ Migration of selenium1 code to selenium2
- ➢ Project structure organization

TestNG
- ➢ Brief about the Testing framework and Testing
- ➢ Annotation concept and different types
- ➢ Testing Configuration
- ➢ Suite execution from command prompt without eclipse
- ➢ Parallel script execution
- ➢ Grouping the test cases
- ➢ Parameter configurations
- ➢ Suite creation methodology
- ➢ Report generation
- ➢ Assertions
- ➢ Verifications

- ➢ Listeners in Testing
- ➢ Create a customized report using Testing

AutoIT
- ➢ The need for AutoIT tool
- ➢ Basics of AutoIT tool and create a simple script
- ➢ Converting AU3 file to exe file
- ➢ Calling AutoIT script from selenium project

Selenium Grid
- ➢ Introduction of Selenium Grid
- ➢ Installation and setup of Hub and node
- ➢ Script changes for Selenium grid execution
- ➢ Simple script execution using grid
- ➢ Parallel execution using grid

✚ Prerequisites:

- Basic Programming knowledge

✚ Who Can attend:
- Software testing can be learned by the people who want to start their career in the IT industry but not interested in coding. It can be learned by the people who want to learn various testing tools and enjoy finding bugs.
- If you are a just college pass-out, this is EXACTLY what you are looking for to open the doors for your dream career
- If you are an experienced professional from ANY other field but wanted to be in software testing, this course will help you make this switch smoothly
- If you are an experienced testing professional, you will be amazed by the new things and advanced tactics you will learn to work efficiently and smartly in this field.

✚ Number of Hours: 70hrs

✚ Certification:

Key Features:

- ➢ One to One Training
- ➢ Online Training
- ➢ Fastrack & Normal Track
- ➢ Resume Modification
- ➢ Mock Interviews
- ➢ Video Tutorials
- ➢ Materials
- ➢ Real Time Projects
- ➢ Virtual Live Experience
- ➢ Preparing for Certification