Table of Contents

[1.0 Automation Testing Overview & Importance 12](#_Toc32812809)

[ What is Automation Testing 12](#_Toc32812810)

[ Why to learn Automation Testing 12](#_Toc32812811)

[ What is the main objective of Automation Testing 12](#_Toc32812812)

[ Difference between Manual vs. Automation Testing 12](#_Toc32812813)

[ What are the advantages of Selenium Automation Testing 12](#_Toc32812814)

[ Selenium vs. Other Automation Tools 12](#_Toc32812815)

[ When to start Automation Testing 12](#_Toc32812816)

[ Why I have to learn Selenium Automation Testing only 12](#_Toc32812817)

[1.1 Selenium Components 12](#_Toc32812818)

[1. Selenium IDE 12](#_Toc32812819)

[2. Selenium RC 12](#_Toc32812820)

[3. Selenium WebDriver & Web Driver Architecture flow diagram 12](#_Toc32812821)

[4. Selenium Grid 12](#_Toc32812822)

[ Selenium Official Website blog - https://selenium.dev/downloads/ 12](#_Toc32812823)

[ Latest stable version of Selenium 3.141.59 12](#_Toc32812824)

[ Latest Selenium 4 Alpha version 4.0.0-alpha-4 12](#_Toc32812825)

[1.0 Selenium Installation guide (Java,Eclipse,TestNG,WebDriver) 12](#_Toc32812826)

[ Install Java/JDK 12](#_Toc32812827)

[ Install Eclipse 12](#_Toc32812828)

[ Import/Create Maven Project 12](#_Toc32812829)

[ Update Build Path from JRE 1.5 or 1.6 to Latest Version of JRE installed 12](#_Toc32812830)

[ Update POM.xml with dependencies 12](#_Toc32812831)

[ Verify JRE System Libraries 12](#_Toc32812832)

[ Verify Maven Dependencies Libraries 12](#_Toc32812833)

[ TestNG Setup 12](#_Toc32812834)

[ Run Basic Test and verify that Chrome Browser launched automatically 12](#_Toc32812835)

[1.1 How to launch Web browser by using System.setProperty 13](#_Toc32812836)

[ Chrome 13](#_Toc32812837)

[ Internet Explorer 13](#_Toc32812838)

[ Microsoft Edge 13](#_Toc32812839)

[ Opera 13](#_Toc32812840)

[ FireFox 13](#_Toc32812841)

[1.2 How to launch Web browser by using Git Hib WebDriver Manager class 13](#_Toc32812842)

[ WebDriverManager.*chromedriver*(); 13](#_Toc32812843)

[ WebDriverManager.*edgedriver*(); 13](#_Toc32812844)

[ WebDriverManager.*firefoxdriver*(); 13](#_Toc32812845)

[ WebDriverManager.*iedriver*(); 13](#_Toc32812846)

[ WebDriverManager.*operadriver*(); 13](#_Toc32812847)

[1.3 Browser Navigation commands 13](#_Toc32812848)

[ driver.navigate().back(); 13](#_Toc32812849)

[ driver.navigate().forward(); 13](#_Toc32812850)

[ driver.navigate().to(""); 13](#_Toc32812851)

[ driver.navigate().refresh(); 13](#_Toc32812852)

[ driver.get("") 13](#_Toc32812853)

[1.4 Steps to write First Basic Test in Selenium 13](#_Toc32812854)

[ Created Package 13](#_Toc32812855)

[ Created a Class 13](#_Toc32812856)

[ Created method inside class 13](#_Toc32812857)

[ @Test 13](#_Toc32812858)

[ WebDriver driver 13](#_Toc32812859)

[ Four line of code inside main method 13](#_Toc32812860)

[2.0 How to write Selenium Basic tests by using methods 13](#_Toc32812861)

[ Static methods 13](#_Toc32812862)

[ Non Static methods 13](#_Toc32812863)

[ Static methods with parameters 13](#_Toc32812864)

[ Non static methods with parameters 13](#_Toc32812865)

[ Highlight Web Element border 13](#_Toc32812866)

[3.0 Locators in Selenium 14](#_Toc32812867)

[1. ID 14](#_Toc32812868)

[2. Name 14](#_Toc32812869)

[3. XPath 14](#_Toc32812870)

[4. Link Text 14](#_Toc32812871)

[5. Partial Link Text 14](#_Toc32812872)

[6. Class Name 14](#_Toc32812873)

[7. CSS Selector 14](#_Toc32812874)

[8. Tag Name 14](#_Toc32812875)

[ Find Element & Find Elements 14](#_Toc32812876)

[4.0 How to handle dropdown 14](#_Toc32812877)

[ Select class in Selenium 14](#_Toc32812878)

[ selectByIndex 14](#_Toc32812879)

[ selectByValue 14](#_Toc32812880)

[ selectByVisibleText 14](#_Toc32812881)

[ How to select single value in dropdown 14](#_Toc32812882)

[ How to select multiple values in dropdown 14](#_Toc32812883)

[5.0 TestNG Data Provider 15](#_Toc32812884)

[ TestNG Data Provider Lab 1– Data driven test, managing all input data inside Data Provider 15](#_Toc32812885)

[ TestNG Data Provider Lab 2 – Data driven test, managing all input data in Excel Sheet 15](#_Toc32812886)

[ TestNG Data Provider Lab 3– Each Test Execution runs on different browser 15](#_Toc32812887)

[6.0 Reports in Selenium 15](#_Toc32812888)

[ Email able Report 15](#_Toc32812889)

[ Custom Email able Report 15](#_Toc32812890)

[ Extent Reports 15](#_Toc32812891)

[7.0 Katalon Recorder 15](#_Toc32812892)

[ Generate Test Script automatically by using Katalon Recorder 15](#_Toc32812893)

[8.0 TestNG Parameters 15](#_Toc32812894)

[ Give all input data in TestNG.xml and Run the program 15](#_Toc32812895)

[ Data driven Test by giving all input data in TestNG.xml 15](#_Toc32812896)

[ Data driven Test in single session 15](#_Toc32812897)

[9.0 parallel Test Execution by using TestNG.xml batch file 15](#_Toc32812898)

[ Parallel - Tests 15](#_Toc32812899)

[ Parallel - methods 15](#_Toc32812900)

[ Parallel - classes 15](#_Toc32812901)

[ Parallel – None (Executes one by one) 15](#_Toc32812902)

[10.0 Test Execution based on priority order 15](#_Toc32812903)

[ Test Execution based on @Test(priority=1) 15](#_Toc32812904)

[ Test Execution without priority order 15](#_Toc32812905)

[ Test Execution based on alphabetical order 15](#_Toc32812906)

[11.0 Synchronization 15](#_Toc32812907)

[ Implicitly Wait 15](#_Toc32812908)

[ Explicit Wait 15](#_Toc32812909)

[ Fluent Wait 15](#_Toc32812910)

[ Thread. Sleep 15](#_Toc32812911)

[ Different Expected conditions in Web Driver 15](#_Toc32812912)

[12.0 How to switch driver, when multiple windows / tabs 16](#_Toc32812913)

[ driver.getWindowHandles() 16](#_Toc32812914)

[ driver.close() 16](#_Toc32812915)

[ driver.quit() 16](#_Toc32812916)

[ How to switch driver when there are two winodws/tabs 16](#_Toc32812917)

[ How to switch driver when there are three or more winodws/tabs 16](#_Toc32812918)

[13.0 JavascriptExecutor 16](#_Toc32812919)

[ sendKeys operation by using java Script executor 16](#_Toc32812920)

[ Click operation by using java Script executor 16](#_Toc32812921)

[ Scrolling by using java Script executor 16](#_Toc32812922)

[ Open blank New tab by using java Script executor 16](#_Toc32812923)

[ js.executeScript(script, args) 16](#_Toc32812924)

[ js.executeAsyncScript(script, args) 16](#_Toc32812925)

[14.0 Action Class 16](#_Toc32812926)

[ Mouse Hover action / Handling Menu operations 16](#_Toc32812927)

[ Double click 16](#_Toc32812928)

[ Right Click & Context Menu 16](#_Toc32812929)

[ Drag and drop 16](#_Toc32812930)

[15.0 Robot Class 16](#_Toc32812931)

[ File Upload and download by using Robot class 16](#_Toc32812932)

[16.0 Advanced XPATH technics to locate a web element 16](#_Toc32812933)

[1. Absolute XPATH 16](#_Toc32812934)

[2. Relative XPATH 16](#_Toc32812935)

[3. OR and AND logical operators 16](#_Toc32812936)

[4. Text 16](#_Toc32812937)

[5. Contains 16](#_Toc32812938)

[6. Following 16](#_Toc32812939)

[7. Following Siblings 16](#_Toc32812940)

[8. Parent 16](#_Toc32812941)

[9. Child 16](#_Toc32812942)

[10. Ancestor 16](#_Toc32812943)

[11. Descendant 16](#_Toc32812944)

[12. Preceding 16](#_Toc32812945)

[13. Self 16](#_Toc32812946)

[14. Starts-with 16](#_Toc32812947)

[17.0 Read or Write operations on Excel (.xls or .xlsx) 17](#_Toc32812948)

[1. Read data from .xls file 17](#_Toc32812949)

[2. write data into into .xls file 17](#_Toc32812950)

[3. Read data from .xlsx file 17](#_Toc32812951)

[4. write data into into .xlsx file 17](#_Toc32812952)

[5. .xls file supporting class is HSSFWorkbook,HSSFSheet,HSSFRow,HSSFCell 17](#_Toc32812953)

[6. .xlsx file supporting class is XSSFWorkbook,XSSFSheet,XSSFRow,XSSFCell 17](#_Toc32812954)

[18.0 Take Screen Shot 17](#_Toc32812955)

[ How to take screenshot and save in drive 17](#_Toc32812956)

[19.0 Alerts 17](#_Toc32812957)

[ Alert alert = *driver*.switchTo().alert(); 17](#_Toc32812958)

[ alert.sendKeys("Selenium Sudhakar") 17](#_Toc32812959)

[ alert.accept(); 17](#_Toc32812960)

[ alert.dismiss(); 17](#_Toc32812961)

[20.0 Important validation checks on Web Elements 17](#_Toc32812962)

[ isDispalyed() 17](#_Toc32812963)

[ isEnabled() 17](#_Toc32812964)

[ isSelected () 17](#_Toc32812965)

[21.0 Frames 17](#_Toc32812966)

[ Find Frame by Index 17](#_Toc32812967)

[ Frame by Name 17](#_Toc32812968)

[ Find frame by Web Element 17](#_Toc32812969)

[22.0 Handling Web Tables 17](#_Toc32812970)

[ Export Web Table data into excel 17](#_Toc32812971)

[23.0 TestNG Groups 17](#_Toc32812972)

[ Execute some specific group related Test cases only from Testng.xml 17](#_Toc32812973)

[24.0 TestNG Annotations 17](#_Toc32812974)

[1. @BeforeSuite 17](#_Toc32812975)

[2. @BeforeTest 17](#_Toc32812976)

[3. @BeforeClass 17](#_Toc32812977)

[4. @BeforeMethod 17](#_Toc32812978)

[5. @Test 17](#_Toc32812979)

[6. @AfterMethod 17](#_Toc32812980)

[7. @AfterClass 17](#_Toc32812981)

[8. @AfterTest 17](#_Toc32812982)

[9. @AfterSuite 18](#_Toc32812983)

[10. @DataProvider 18](#_Toc32812984)

[25.0 TestNG Listeners 18](#_Toc32812985)

[ ITestListener 18](#_Toc32812986)

[ ITestContext 18](#_Toc32812987)

[ ITestResult 18](#_Toc32812988)

[26.0 Event Firing Web Driver 18](#_Toc32812989)

[ EventFiringWebDriver 18](#_Toc32812990)

[27.0 Execute Failed Test Cases only 18](#_Toc32812991)

[ Execute Failed Test Cases 18](#_Toc32812992)

[ Test Suite Reports 18](#_Toc32812993)

[ TestNG-Failed.xml 18](#_Toc32812994)

[28.0 Broken Links 18](#_Toc32812995)

[ Find the Hyperlinks where 404 Page Not Found Exception 18](#_Toc32812996)

[29.0 Date Time picker 18](#_Toc32812997)

[ Handling Calendar pick list and populate Date Time Picker 18](#_Toc32812998)

[30.0 Method Overloading Test Scenario 18](#_Toc32812999)

[ Method Overloading Test Scenario 18](#_Toc32813000)

[31.0 Method overriding Test Scenario 18](#_Toc32813001)

[ Method overriding Test Scenario 18](#_Toc32813002)

[32.0 Smart Search Test Scenario Automation 18](#_Toc32813003)

[ Find the Keywords in Smart Search Test Scenario 18](#_Toc32813004)

[33.0 Cookies 18](#_Toc32813005)

[ driver.manage().getCookies(); 18](#_Toc32813006)

[ driver.manage().getCookieNamed(arg0); 18](#_Toc32813007)

[ driver.manage().addCookie(arg0); 18](#_Toc32813008)

[ driver.manage().deleteCookie(arg0); 18](#_Toc32813009)

[ driver.manage().deleteCookieNamed(arg0); 18](#_Toc32813010)

[ driver.manage().deleteAllCookies(); 18](#_Toc32813011)

[34.0 Exception Handling 19](#_Toc32813012)

[ Try-catch-Finally block 19](#_Toc32813013)

[ TC001\_NoSuchElementException 19](#_Toc32813014)

[ TC002\_StaleElementException 19](#_Toc32813015)

[ TC003\_ElementNotVisibleException 19](#_Toc32813016)

[ TC004\_NoSuchWindowException 19](#_Toc32813017)

[ TC005\_NoSuchFrameException 19](#_Toc32813018)

[ TC006\_NoAlertPresentException 19](#_Toc32813019)

[ TC007\_InvalidSelectorException 19](#_Toc32813020)

[ TC008\_ElementNotSelectableException 19](#_Toc32813021)

[ TC009\_TimeoutException 19](#_Toc32813022)

[ TC010\_NoSuchSessionException 19](#_Toc32813023)

[35.0 Selenium Grid 19](#_Toc32813024)

[ Execute Test cases in Remote system 19](#_Toc32813025)

[ Hub & Note concepts 19](#_Toc32813026)

[ Remote Web Driver 19](#_Toc32813027)

[ Desired Capabilities 19](#_Toc32813028)

[36.0 Create Maven Project in Eclipse 19](#_Toc32813029)

[ How to Import an Existing Maven Project into eclipse 19](#_Toc32813030)

[ How to create a new maven Project in eclipse 19](#_Toc32813031)

[ How to update Maven Dependencies in POM.xml 19](#_Toc32813032)

[ How to add Plug-ins of TestNG entry in POM.xml 19](#_Toc32813033)

[37.0 Page Object Model 19](#_Toc32813034)

[ Test Script automation by using Page Object Model 19](#_Toc32813035)

[38.0 Page Factory Model 19](#_Toc32813036)

[ Test Script automation by using Page Factory Model 19](#_Toc32813037)

[39.0 Key Word Driven Framework Approach 19](#_Toc32813038)

[ Driver Script 19](#_Toc32813039)

[ Actions 19](#_Toc32813040)

[ Keywords 19](#_Toc32813041)

[ Automate Test Case by using Key word driven framework approach 19](#_Toc32813042)

[40.0 Hybrid Framework Approach 19](#_Toc32813043)

[ Integrating Data driven Framework + Page Object Model 19](#_Toc32813044)

[ One Live Project Implementation by using Hybrid Framework 19](#_Toc32813045)

[41.0 Selenium with Cucumber (BDD Framework) 20](#_Toc32813046)

[ Why use Cucumber with Selenium 20](#_Toc32813047)

[ Prerequisite for using Cucumber with Selenium 20](#_Toc32813048)

[ Jar files For Cucumber 20](#_Toc32813049)

[ Automation Testing Using Cucumber with Selenium 20](#_Toc32813050)

[ @Given, @When , @Then 20](#_Toc32813051)

[ Test Execution & Results 20](#_Toc32813052)

[42.0 Git HUB 20](#_Toc32813053)

[ How to create Local Repository 20](#_Toc32813054)

[ How to create Remote Repository 20](#_Toc32813055)

[ Push & Commit changes from Local Repository to Remote Repository 20](#_Toc32813056)

[ How to download Remote Repository 20](#_Toc32813057)

[43.0 Jenkins 20](#_Toc32813058)

[ maven-compiler-plugin 20](#_Toc32813059)

[ maven-surefire-plugin 20](#_Toc32813060)

[ testng.xml 20](#_Toc32813061)

[ Install Jenkins and configure it to Run Maven with TestNg 20](#_Toc32813062)

[ Scheduling Jenkins for automatic execution. 20](#_Toc32813063)

[44.0 Selenium API Automation Testing 20](#_Toc32813064)

[ JSON – Request & Response 20](#_Toc32813065)

[ SOAP UI – XML Request & Response 20](#_Toc32813066)

[ GET / POST / PUT / PATCH / DELETE  requests. 20](#_Toc32813067)

[45.0 JDBC – Database Testing 20](#_Toc32813068)

[ JDBC Driver. 20](#_Toc32813069)

[ Connection. 20](#_Toc32813070)

[ Statement. 20](#_Toc32813071)

[ PreparedStatement. 20](#_Toc32813072)

[ CallableStatement. 20](#_Toc32813073)

[ ResultSet. 20](#_Toc32813074)

[ Batch Updates. 20](#_Toc32813075)

[ Transactions. 20](#_Toc32813076)

[46.0 Career Guidance 21](#_Toc32813077)

[ Sample Selenium Profile Resume Assistance 21](#_Toc32813078)

[ Personal Lab Assistance will be taken care 21](#_Toc32813079)

[ We will ensure that you are comfortable in all 40 labs 21](#_Toc32813080)

[ 2 Live Projects Automation 21](#_Toc32813081)

[ Selenium Interview FAQ’s 21](#_Toc32813082)

[ 2 rounds of Mock Interview Preparation & Improvements 21](#_Toc32813083)

[ Job Assistance 21](#_Toc32813084)

[ Technical Skills : Java, jdbc, Selenium 3.141.59 , Eclipse, Git Hub, Jenkins,HP ALM 12.53 ,TestNG 6.14.3 21](#_Toc32813085)

[ Framework Expertized in : Page Object Model Design pattern, Page Factory Design Pattern, Data driven frame work , Key word driven frame work , BDD Cucumber frame work, Hybrid Framework 21](#_Toc32813086)

[ After completion of course you can able to automate any web sites independently 21](#_Toc32813087)

## 1.0 Automation Testing Overview & Importance

### What is Automation Testing

### Why to learn Automation Testing

### What is the main objective of Automation Testing

### Difference between Manual vs. Automation Testing

### What are the advantages of Selenium Automation Testing

### Selenium vs. Other Automation Tools

### When to start Automation Testing

### Why I have to learn Selenium Automation Testing only

## 1.1 Selenium Components

### Selenium IDE

### Selenium RC

### Selenium WebDriver & Web Driver Architecture flow diagram

### Selenium Grid

### Selenium Official Website blog - <https://selenium.dev/downloads/>

### Latest stable version of Selenium [3.141.59](https://bit.ly/2TlkRyu)

### Latest Selenium 4 Alpha version [4.0.0-alpha-4](https://selenium-release.storage.googleapis.com/4.0-alpha4/selenium-server-4.0.0-alpha-4.jar)

## 1.0 Selenium Installation guide (Java,Eclipse,TestNG,WebDriver)

### Install Java/JDK

### Install Eclipse

### Import/Create Maven Project

### Update Build Path from JRE 1.5 or 1.6 to Latest Version of JRE installed

### Update POM.xml with dependencies

### Verify JRE System Libraries

### Verify Maven Dependencies Libraries

### TestNG Setup

### Run Basic Test and verify that Chrome Browser launched automatically

## 1.1 How to launch Web browser by using System.setProperty

### Chrome

### Internet Explorer

### Microsoft Edge

### Opera

### FireFox

## 1.2 How to launch Web browser by using Git Hib WebDriver Manager class

### WebDriverManager.*chromedriver*();

### WebDriverManager.*edgedriver*();

### WebDriverManager.*firefoxdriver*();

### WebDriverManager.*iedriver*();

### WebDriverManager.*operadriver*();

## 1.3 Browser Navigation commands

### driver.navigate().back();

### driver.navigate().forward();

### driver.navigate().to("");

### driver.navigate().refresh();

### driver.get("")

## 1.4 Steps to write First Basic Test in Selenium

### Created Package

### Created a Class

### Created method inside class

### @Test

### WebDriver driver

### Four line of code inside main method

## 2.0 How to write Selenium Basic tests by using methods

### Static methods

### Non Static methods

### Static methods with parameters

### Non static methods with parameters

### Highlight Web Element border

## 3.0 Locators in Selenium

### ID

### Name

### XPath

### Link Text

### Partial Link Text

### Class Name

### CSS Selector

### Tag Name

### Find Element & Find Elements

* driver.findElement(By.cssSelector(""));
* driver.findElement(By.className(""));
* driver.findElement(By.id(""));
* driver.findElement(By.name(""));
* driver.findElement(By.linkText(""));
* driver.findElement(By.partialLinkText(""));
* driver.findElement(By.tagName(""));
* driver.findElement(By.xpath(""));

## 4.0 How to handle dropdown

### Select class in Selenium

### selectByIndex

### selectByValue

### selectByVisibleText

### How to select single value in dropdown

### How to select multiple values in dropdown

* Select dropdown1 = new Select(driver.findElement(By.name("fromMonth")));
* dropdown1.selectByIndex(1);
* dropdown1.selectByValue("06");
* dropdown1.selectByVisibleText("June");

## 5.0 TestNG Data Provider

### TestNG Data Provider Lab 1– Data driven test, managing all input data inside Data Provider

### TestNG Data Provider Lab 2 – Data driven test, managing all input data in Excel Sheet

### TestNG Data Provider Lab 3– Each Test Execution runs on different browser

## 6.0 Reports in Selenium

### Email able Report

### Custom Email able Report

### Extent Reports

## 7.0 Katalon Recorder

### Generate Test Script automatically by using Katalon Recorder

## 8.0 TestNG Parameters

### Give all input data in TestNG.xml and Run the program

### Data driven Test by giving all input data in TestNG.xml

### Data driven Test in single session

## 9.0 parallel Test Execution by using TestNG.xml batch file

### Parallel - Tests

### Parallel - methods

### Parallel - classes

### Parallel – None (Executes one by one)

## 10.0 Test Execution based on priority order

### Test Execution based on @Test(priority=1)

### Test Execution without priority order

### Test Execution based on alphabetical order

## 11.0 Synchronization

### Implicitly Wait

### Explicit Wait

### Fluent Wait

### Thread. Sleep

### Different Expected conditions in Web Driver

## 12.0 How to switch driver, when multiple windows / tabs

### driver.getWindowHandles()

### driver.close()

### driver.quit()

### How to switch driver when there are two winodws/tabs

### How to switch driver when there are three or more winodws/tabs

## 13.0 JavascriptExecutor

### sendKeys operation by using java Script executor

### Click operation by using java Script executor

### Scrolling by using java Script executor

### Open blank New tab by using java Script executor

### js.executeScript(script, args)

### js.executeAsyncScript(script, args)

## 14.0 Action Class

### [Mouse Hover action / Handling Menu operations](https://www.toolsqa.com/selenium-webdriver/mouse-hover-action/)

### Double click

### Right Click & Context Menu

### Drag and drop

## 15.0 Robot Class

### [File Upload and download by using Robot class](https://www.toolsqa.com/selenium-webdriver/mouse-hover-action/)

## 16.0 Advanced XPATH technics to locate a web element

### Absolute XPATH

### Relative XPATH

### OR and AND logical operators

### Text

### Contains

### Following

### Following Siblings

### Parent

### Child

### Ancestor

### Descendant

### Preceding

### Self

### Starts-with

## 17.0 Read or Write operations on Excel (.xls or .xlsx)

### Read data from .xls file

### write data into into .xls file

### Read data from .xlsx file

### write data into into .xlsx file

### .xls file supporting class is HSSFWorkbook,HSSFSheet,HSSFRow,HSSFCell

### .xlsx file supporting class is XSSFWorkbook,XSSFSheet,XSSFRow,XSSFCell

## 18.0 Take Screen Shot

### How to take screenshot and save in drive

## 19.0 Alerts

### Alert alert = *driver*.switchTo().alert();

### alert.sendKeys("Selenium Sudhakar")

### alert.accept();

### alert.dismiss();

## 20.0 Important validation checks on Web Elements

### isDispalyed()

### isEnabled()

### isSelected ()

## 21.0 Frames

### Find Frame by Index

### Frame by Name

### Find frame by Web Element

## 22.0 Handling Web Tables

### Export Web Table data into excel

## 23.0 TestNG Groups

### Execute some specific group related Test cases only from Testng.xml

## 24.0 TestNG Annotations

### @BeforeSuite

### @BeforeTest

### @BeforeClass

### @BeforeMethod

### @Test

### @AfterMethod

### @AfterClass

### @AfterTest

### @AfterSuite

### @DataProvider

## 25.0 TestNG Listeners

### ITestListener

### ITestContext

### ITestResult

## 26.0 Event Firing Web Driver

### EventFiringWebDriver

## 27.0 Execute Failed Test Cases only

### Execute Failed Test Cases

### Test Suite Reports

### TestNG-Failed.xml

## 28.0 Broken Links

### Find the Hyperlinks where 404 Page Not Found Exception

## 29.0 Date Time picker

### Handling Calendar pick list and populate Date Time Picker

## 30.0 Method Overloading Test Scenario

## Method Overloading Test Scenario

## 31.0 Method overriding Test Scenario

## Method overriding Test Scenario

## 32.0 Smart Search Test Scenario Automation

## Find the Keywords in Smart Search Test Scenario

## 33.0 Cookies

### driver.manage().getCookies();

### driver.manage().getCookieNamed(arg0);

### driver.manage().addCookie(arg0);

### driver.manage().deleteCookie(arg0);

### driver.manage().deleteCookieNamed(arg0);

### driver.manage().deleteAllCookies();

## 34.0 Exception Handling

### Try-catch-Finally block

### TC001\_NoSuchElementException

### TC002\_StaleElementException

### TC003\_ElementNotVisibleException

### TC004\_NoSuchWindowException

### TC005\_NoSuchFrameException

### TC006\_NoAlertPresentException

### TC007\_InvalidSelectorException

### TC008\_ElementNotSelectableException

### TC009\_TimeoutException

### TC010\_NoSuchSessionException

## 35.0 Selenium Grid

### Execute Test cases in Remote system

### Hub & Note concepts

### Remote Web Driver

### Desired Capabilities

## 36.0 Create Maven Project in Eclipse

### How to Import an Existing Maven Project into eclipse

### How to create a new maven Project in eclipse

### How to update Maven Dependencies in POM.xml

### How to add Plug-ins of TestNG entry in POM.xml

## 37.0 Page Object Model

### Test Script automation by using Page Object Model

## 38.0 Page Factory Model

### Test Script automation by using Page Factory Model

## 39.0 Key Word Driven Framework Approach

### Driver Script

### Actions

### Keywords

### Automate Test Case by using Key word driven framework approach

## 40.0 Hybrid Framework Approach

### Integrating Data driven Framework + Page Object Model

### One Live Project Implementation by using Hybrid Framework

## 41.0 Selenium with Cucumber (BDD Framework)

### Why use Cucumber with Selenium

### Prerequisite for using Cucumber with Selenium

### Jar files For Cucumber

### Automation Testing Using Cucumber with Selenium

### @Given, @When , @Then

### Test Execution & Results

## 42.0 Git HUB

### How to create Local Repository

### How to create Remote Repository

### Push & Commit changes from Local Repository to Remote Repository

### How to download Remote Repository

## 43.0 Jenkins

### maven-compiler-plugin

### maven-surefire-plugin

### testng.xml

### Install Jenkins and configure it to Run Maven with TestNg

### Scheduling Jenkins for automatic execution.

## 44.0 Selenium API Automation Testing

### JSON – Request & Response

### SOAP UI – XML Request & Response

### GET / POST / PUT / PATCH / DELETE  requests.

## 45.0 JDBC – Database Testing

### JDBC Driver.

### Connection.

### Statement.

### PreparedStatement.

### CallableStatement.

### ResultSet.

### Batch Updates.

### Transactions.

## 46.0 Career Guidance

### Sample Selenium Profile Resume Assistance

### Personal Lab Assistance will be taken care

### We will ensure that you are comfortable in all 40 labs

### 2 Live Projects Automation

### Selenium Interview FAQ’s

### 2 rounds of Mock Interview Preparation & Improvements

### Job Assistance

### Technical Skills : Java, Selenium 3.141.59 , Eclipse, Git Hub, Jenkins,HP ALM 12.53 ,TestNG 6.14.3

### Framework Expertized in : Page Object Model Design pattern, Page Factory Design Pattern, Data driven frame work , Key word driven frame work , BDD Cucumber frame work, Hybrid Framework

### After completion of course you can able to automate any web sites independently