



KRN INFORMATIX

BRIDGE YOUR CAREER



SELENIUM

WEB DRIVER with PROJECT

- ◆ Java for Selenium
- ◆ Selenium IDE
- ◆ Selenium WebDriver
- ◆ TestNG Framework
- ◆ Framework
- ◆ ANT
- ◆ Maven
- ◆ DataBase Testing Using Selenium
- ◆ Grid
- ◆ POM (Page Object Model)
- ◆ Automating Window Components
- ◆ GIT
- ◆ Jenkins Integration
- ◆ Log4j
- ◆ CUCUMBER

Date:.....Timing:

Duration:.....Fee:.....

Faculty:.....

KRN INFORMATIX

#85/12, 3rd floor, Dental College Rd, Marathahalli - 37

+91 9700779923

JAVA FOR SELENIUM

- ◆ Selenium Overview
- ◆ Installing Java
- ◆ Installing Eclipse
- ◆ Features of Java
- ◆ Why Java for Selenium
- ◆ First Eclipse Project
- ◆ First Java program
- ◆ Concept of class file
- ◆ Platform Independence
- ◆ Datatypes in Java
- ◆ String class
- ◆ If statements
- ◆ Conditional and concatenation operators
- ◆ While Loop
- ◆ For Loops
- ◆ Practical Examples with loops
- ◆ Usage of loops in Selenium
- ◆ Single Dimensional Arrays
- ◆ Two Dimensional arrays
- ◆ Practical usage of arrays in Selenium
- ◆ Drawbacks of arrays
- ◆ What are Functions?
- ◆ Function Input Parameters
- ◆ Function Return Types
- ◆ Local Variables
- ◆ Global Variables
- ◆ Static and Non-Static Variables
- ◆ Static and Non-Static Functions
- ◆ Creating Objects in Java
- ◆ Meaning of static
- ◆ Why is main method static?
- ◆ Object and Object References
- ◆ Call by reference and Value
- ◆ Constructors
- ◆ Usage of Objects in Selenium
- ◆ Concept of Inheritance
- ◆ Interface
- ◆ Overloadings and Overriding Functions
- ◆ Example on inheritance
- ◆ Object Class
- ◆ Usage of Inheritance in Selenium
- ◆ Relevance of Packages
- ◆ Creating Packages
- ◆ Accessing Classes Across Packages
- ◆ Good Features of eclipse
- ◆ Accessing modifiers - Public, Private, Default, Protected
- ◆ Exception handling with try catch block
- ◆ Importance of exception handling
- ◆ Exception and Error
- ◆ Throwable Class
- ◆ Final and Finally
- ◆ Throw and Throws
- ◆ Different Types of Exceptions
- ◆ Need of exception handling in Selenium framework
- ◆ Introduction to Collections API

- ◆ ArrayList Class
- ◆ Hashtable Class
- ◆ String class and functions
- ◆ Reading/Writing Text Files
- ◆ Reading Properties File in Java
- ◆ Concept of jar file
- ◆ Reading/Writing Microsoft XLS Files

SELENIUM IDE

- ◆ Installing Selenium IDE
- ◆ Recording Script
- ◆ Running, Pausing and debugging Script
- ◆ Running a script line by line
- ◆ Inserting commands in between script
- ◆ XPATHS and installing firebug to get XPATHS
- ◆ Wait Commands
- ◆ Verification and Assertions
- ◆ Should I use verify or assertion
- ◆ JavaScript
- ◆ User-extension.js in Selenium IDE

SELENIUM WEBDRIVER

- ◆ Why WebDriver?
- ◆ Downloading WebDriver Jars and configuring in eclipse
- ◆ Architecture of selenium webdriver
- ◆ Drivers for Firefox, IE, chrome, Iphone, Android etc
- ◆ First Selenium Code
- ◆ Working with chrome and IE
- ◆ Selenium RC and WebDriver
- ◆ Concept of firefox profile
- ◆ What is Firefox profile
- ◆ Why we need firefox Profile
- ◆ Close and Quit -Difference
- ◆ Importing webdriver documentation in eclipse
- ◆ WebDriver DesiredCapabilities Class
- ◆ Proxy settings with webdriver/Working with proxy Servers
- ◆ HTMLUnit driver and desired capabilities
- ◆ Firepath and firebug Add-ons installation in Mozilla
- ◆ Inspecting elements in Mozilla, Chrome and IE
- ◆ HTML language tags and attributes
- ◆ Various locator strategies
- ◆ WebDriver Interface
- ◆ WebElement Interface
- ◆ Identifying WebElements using id, name, class
- ◆ Finding Xpaths to Identify
- ◆ Absolute and complete Xpaths
- ◆ Creating customized Xpaths without firebug
- ◆ Css Selectors
- ◆ Generating own CssSelectors
- ◆ Performance of CssSelectors as compared to Xpaths
- ◆ Finding xpath/cssselectors in different browsers - Mozilla, Chrome and IE
- ◆ Objects with same id/xpath/cssSelector
- ◆ What is class attribute?

- ◆ Handling Dynamic objects/ids on the page
- ◆ Working with different browsers without changing code
- ◆ Managing Input fields, Buttons and creating custom xpaths
- ◆ Managing/Identifying Links with xpaths/css selectors
- ◆ Extracting More than one object from a page
- ◆ Extracting all links of a page/Bulk extraction of objects
- ◆ Extracting Objects from a specific area of a web page
- ◆ Various strategies to test Links on a page by clicking on them one by one
- ◆ Finding whether object is present on page or not
- ◆ Handling drop down list
- ◆ Select Class in Selenium API
- ◆ Managing radio buttons and Checkboxes
- ◆ Hidden components
- ◆ isDisplayed function
- ◆ Taking Screenshots of the web page
- ◆ Implicit and Explicit waits
- ◆ PageLoadTimeout Property
- ◆ WebDriverWait Class
- ◆ WebDriver.Timeout Interface
- ◆ ExpectedCondition interface and ExpectedConditions class
- ◆ WaitUntil Condition
- ◆ Fluent Wait
- ◆ Managing Ajax based components
- ◆ Concepts of Set Interface in Java
- ◆ Window Handles
- ◆ Managing tabbed windows in IE, Chrome and Mozilla
- ◆ Managing popups in IE, Chrome and Mozilla
- ◆ Closing windows
- ◆ Default Popups
- ◆ Extracting Data From WebTable
- ◆ Dynamic WebTable Handling
- ◆ Attaching files with Selenium
- ◆ Changing your facebook profile picture by attaching new picture
- ◆ Mouse movement with Selenium- Mouse Interface
- ◆ Handling Ajax Autosuggests
- ◆ Handling Google Ajax Autosuggests
- ◆ Handling Frames in Web Page
- ◆ Handling cookies
- ◆ More Examples on Webtables
- ◆ Webtables and css Selectors
- ◆ Building custom functions for Webtables
- ◆ Managing Javascript alerts
- ◆ Simulating front and back button click on Browser using selenium
- ◆ Assigning Firefox profile parameters
- ◆ Downloading files using selenium
- ◆ Selenium Javadocs
- ◆ Moving a mouse on a Object and right clicking on it
- ◆ Finding Coordinates of a Web Object
- ◆ Actions class In Webdriver-
- ◆ Handling CSS menu with Action class-
- ◆ Handling CSS menu with JavaScriptExecutor-
- ◆ JavaScriptExecutor example-
- ◆ Drag, drop, native events

TestNG FRAMEWORK

- ◆ Testng Framework
- ◆ TestNG Framework
- ◆ What is TestNg
- ◆ Installing TestNg in Eclipse
- ◆ TestNg annotations
- ◆ Understanding usage of annotations
- ◆ Running a Test in TestNg
- ◆ Batch Running of tests in TestNg
- ◆ Skipping Tests
- ◆ Parameterizing Tests - DataProvider
- ◆ Assertions/Reporting Errors
- ◆ TestNg Reports
- ◆ Advantages over Junit
- ◆ Using TestNg in Selenium

FRAMEWORKS

- ◆ Modular Driven Framework
- ◆ Data driven Framework
- ◆ Keyword driven Framework
- ◆ Hybrid Framework

ANT

- ◆ Downloading and configuring Ant
- ◆ Build.xml configuration
- ◆ HTML Report generation using Ant
- ◆ What is Ant
- ◆ XSLT report generation generation using TestNg and Ant
- ◆ Building a BAT file to run tests using ANT
- ◆ Overview of Data driven framework
- ◆ Building the Test Base Class
- ◆ Reading XPATHS, Configuration from properties file
- ◆ Initialize the Webdriver
- ◆ Repeating a test with different Data
- ◆ Building utility functions
- ◆ Parameterizing tests using XL Files
- ◆ Controlling Execution order from XL Files
- ◆ Assertions and Reporting Errors / ErrorCollector
- ◆ Storing Screenshots of errors
- ◆ Generating the HTML reports
- ◆ Emailing test reports
- ◆ Creating a BAT file for project execution

MAVEN

- ◆ What is Maven and Why Maven?
- ◆ Installing/Configuring Maven
- ◆ Archetypes in Maven
- ◆ Creating maven project through command line
- ◆ POM.xml
- ◆ Importing Archetypes
- ◆ Maven Repositories
- ◆ Building POM.xml through command line to configure
- ◆ Selenium and Junit
- ◆ Importing the maven project into eclipse
- ◆ Building a selenium project and running it through Maven
- ◆ Maven Antrun Plugin configuration in POM.xml
- ◆ Eclipse plugin for maven
- ◆ Using the maven plugin to execute phases - compile, clean, package etc

DATABASE TESTING USING SELENIUM

- ◆ Installing MySQL Database and making some tables in it
- ◆ java.sql package
- ◆ JDBC Drivers
- ◆ Connection Interface
- ◆ Statement and PreparedStatement Interfaces
- ◆ ResultSet Interface
- ◆ Firing select, insert, update and delete queries with database using Java JDBC
- ◆ Looping the ResultSet
- ◆ Using Annotations of Junit/TestNg to establish connection with database when Using Selenium

GRID

- ◆ Introduction to Grid 2
- ◆ Creating Nodes and Hubs
- ◆ Deciding number of browsers on a Node
- ◆ Deciding type of browsers on Node
- ◆ Limiting number of concurrent browsers on node
- ◆ Difference between maxSession and maxInstance
- ◆ Configuring chromedriver and IEdriver exe files on grid
- ◆ Testng configurations
- ◆ RemoteWebDriver and DesiredCapabilities
- ◆ Configuring JSON file format to initialize/configure hub and nodes
- ◆ Running single test on single node - serially In multiple browsers
- ◆ Running single test on single node – parallelly on multiple browsers
- ◆ Running single test on multiple nodes – each node having different browser
- ◆ Can we decide node to run test?
- ◆ Running multiple tests spread across 3 nodes – one for ie, one for firefox and one for chrome
- ◆ Node Timeout
- ◆ Prioritizing the test cases
- ◆ Prioritizer Interface and CustomPrioritizer
- ◆ Sharing same webdriver instance among multiple tests after prioritizing them
- ◆ Grid Coding in eclipse
- ◆ Creating a Grid sample test case
- ◆ End to end scenario building and execution on Grid - demonstration on one node
- ◆ End to end scenario building and execution on Grid - demonstration on multiple nodes on virtual machine
- ◆ End to end scenario building and execution on Grid - demonstration on multiple nodes on virtual machine and on multiple browsers

POM (PAGE OBJECT MODEL)

- ◆ What is POM
- ◆ When to use POM
- ◆ Inheritance and Encapsulation In POM
- ◆ Designing POM classes
- ◆ Putting up test configurations in a properties file
- ◆ Building Page class
- ◆ Building general utility functions
- ◆ Batch running the test cases

- ◆ Using junit/testing to execute test cases
- ◆ Parameterizing tests using XL Files
- ◆ Taking screenshots
- ◆ Generating reports and reporting errors
- ◆ Mailing the test Reports
- ◆ Creating a Bat file for project executing

AUTOMATING WINDOW COMPONENTS

- ◆ Autoit
- ◆ Robot

GIT

- ◆ Downloading and installing GIT
- ◆ Installing GIT and GITHUB plugins for Jenkins
- ◆ Configuring ssh host keys for GIT and Jenkins
- ◆ Git Bash commands and operations
- ◆ Uploading project to GIT using GIT Bash

JENKINS

- ◆ What is continuous integration
- ◆ How Jenkins helps in continuous integration
- ◆ Downloading and Installing Jenkins
- ◆ Executing simple batch commands
- ◆ Build project at will and build at intervals/ Scheduling builds
- ◆ Build Triggers
- ◆ Mailing if build fails
- ◆ Configuring/Scheduling Maven Project In Jenkins
- ◆ Configuring/Scheduling remote Git maven project in Jenkins

LOG4J

- ◆ Introduction about logging
- ◆ Introduction about Log4J framework
- ◆ Logging Problem without log4J
- ◆ How to solve logging problem with log4J
- ◆ What is layout
- ◆ Different types of layouts In log4J
- ◆ What is Appender
- ◆ Different types of Appenders in log4J

CUCUMBER

- ◆ What Is BDD
- ◆ Downloading Cucumber Jars
- ◆ Junit & Cucumber
- ◆ Preparing Feature File
- ◆ Step definition
- ◆ Runner Class
- ◆ Given, when, Then, And, But annotations
- ◆ Passing Parameters to Step function