**Note: practise with code**

**what are limitations in selenium webdriver**

It does not support and non web-based applications, it only supports web based applications.

Its an open source tool so in case of any technical issues you need to rely on the selenium community forums to get your issue resolved.

You need to know at least one of the supported language very well in order to automate your application successfully.

No inbuilt reporting capability so you need plugins like JUnit and TestNG for test reports.

Lot of challenges with IE browser.

**installing/configure selenium**

1. Right-click on "newproject" and select Properties.
2. On the Properties dialog, click on "Java Build Path".
3. Click on the Libraries tab, and then.
4. Click on "Add External JARs.."

what are different ways of locating elements in selenium

Public class LocatingElements{

Public static void main(String []args){

System.*setProperty*("webdriver.gecko.driver", "Property path");

WebDriver driver=**new** FirefoxDriver ();

driver.get("https://facebook.com");

driver.findElement(By.*xpath*("Xpath”));

driver.findElement(By.id("element id”));

driver.findElement(By.*name*("element name”));

driver.findElement(By.*className*("element classname”));

driver.findElement(By.*cssSelector*("CSS Selector”); driver.findElement(By.*tagName*("element tagname”));

driver.findElement(By.*linkText*("linkText”));

driver.findElement(By.*partialLinkText*("PartialLinktext”));

}

}

**which is fastest way to identify elements in web page?**

The faster way is obvious by using By.id(), but you also have alternative using By.name() also, it also has same speed as of like By.id(). And cssSelector also uses the id, name so its equivalent to same as searching By.id() and By.name() . The main reason for using xpath is that, each web element has the unique path assigned to it, So when the same id, name and classname are shared by two elements, then xpath is the option, as a unique solution.

**what is absolute path and relative path in xpath**

import org.junit.Test;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class AbsoluteRelativePaths {

@Test

public void absolutePath() throws InterruptedException {

System.setProperty("webdriver.chrome.driver", "E:\\Selenium\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver = new ChromeDriver();

driver.get("E:\\Selenium\\absolute.html");

// **Absolute Path starts from root path**

driver.findElement(By.xpath("/html/body/li[@id='test']/a"));

// **Relative Path starts from current path**

driver.findElement(By.xpath(".//\*[@id='test']/a"));

}

}

different types of waits or synchronization in selenium webdriver

Synchronization/ Waits can be achieved in many different ways :

1. **Sleep method** of Thread class
2. **Page Load timeout**
3. **Script timeout**
4. **Implicit Wait**
5. **Explicit Wait**

**Sleep method:**

WebDriver driver = new FirefoxDriver();driver.get(“http://Facebook.com”);driver.manage().window().maximize();

driver.findElement(By.linkText(“Home”)).click();

try{

           Thread.sleep(5000);

} catch(Exception e){

            System.out.print(e);

}

**Page Load timeout:**

driver.manage().timeouts().pageLoadTimeout(20, TimeUnit.SECONDS);

**Script Timeout:**

driver.manage().timeouts().setScriptTimeout(20, TimeUnit.SECONDS);

**Implicity wait:**

driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);

**Explicit Wait:**

WebDriver driver = new FirefoxDriver();                          driver.get(“https://seleniumatfingertips.wordpress.com/”);                          driver.manage().window().maximize();

                          WebElement element = driver.findElement(By.linkText(“Home”));

**// waits until Home link is visible on the web page**

                          WebDriverWait wait = new WebDriverWait(driver, 20);                          wait.until(ExpectedConditions.visibilityOf(element));

                         element.click();

**// waits until About link is visible**                        wait.until(ExpectedConditions.visibilityOfElementLocated

                                           (By.linkText(“About”)));

**how to save screen shots using selenium webdriver**

import java.io.File;

import org.apache.commons.io.FileUtils;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.TakesScreenshot;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class ScreenShotsSelenium {

public static void main(String []args)

{

/\*System.setProperty("webdriver.gecko.driver", "C:\\Program Files\\geckodriver-v0.13.0-win64\\geckodriver.exe");

WebDriver obj=new FirefoxDriver (); \*/

System.setProperty("webdriver.chrome.driver", "C:\\Users\\sreekanth\\Downloads\\chromedriver\_win32\\chromedriver.exe");

WebDriver obj = new ChromeDriver ();

// Maximize the window

obj.manage().window().maximize();

obj.get("http://facebook.com");

File src= ((TakesScreenshot)obj).getScreenshotAs(OutputType.FILE);

try {

// now copy the screenshot to desired location using copyFile //method

//FileUtils.copyFile(src, new File("D:/selenium/error.png"));

FileUtils.copyFile(src, new File("D:/selenium/"+System.currentTimeMillis()+".png"));

}

catch (Exception e)

{

System.out.println(e.getMessage());

}

}

}

**how to handle multiple windows in selenium webdriver**

import java.util.\*;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class WindowsExcersise {

public static void main(String []args)

{

System.setProperty("webdriver.gecko.driver", "C:\\Program Files\\geckodriver-v0.13.0-win64\\geckodriver.exe");

WebDriver driver=new FirefoxDriver ();

driver.get("https://accounts.google.com/SignUp");

driver.findElement(By.xpath(".//\*[@id='wrapper']/div[2]/div/div[1]/p/a")).click();

System.out.println(driver.getTitle());

Set<String>ids = driver.getWindowHandles();

Iterator<String> it=ids.iterator();

String parentid=it.next();

String childid=it.next();

driver.switchTo().window(childid);

System.out.println(driver.getTitle());

driver.switchTo().window(parentid);

}

}

**how to lanuch webpage using chrome driver**

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

public class ChromeTesting {

public static void main(String []args)

{

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\sreekanth\\Downloads\\chromedriver\_win32\\chromedriver.exe");

WebDriver driver = **new** ChromeDriver ();

driver.get("http://google.com");

}}

what is desired capabilities in selenium webdriver

Ex: write code

**how to set language while opening website**

**Using Firefox Browser :**

FirefoxProfile profile = new FirefoxProfile();  
//setting the locale french : ‘fr’  
profile.setPreference(“intl.accept\_languages”,”fr”);  
WebDriver driver = new FirefoxDriver(profile);  
driver.get(“[http://google.co.in&#8221](#8221););

**Using Chrome Browser :**

System.setProperty(“webdriver.chrome.driver”,”D:/DollarArchive/chromedriver.exe”);  
ChromeOptions options = new ChromeOptions();  
options.addArguments(“–lang= sl”);  
WebDriver driver = new ChromeDriver(options);  
driver.get(“[http://google.co.in&#8221](#8221););

Unfortunately it wont work for IE browser, We need to change it manually.

**how to handle windows based popups (upload and dropdown)**

**import** org.openqa.selenium.By;

**import** org.openqa.selenium.WebDriver;

**import** org.openqa.selenium.chrome.ChromeDriver;

**public** **class** HandlingJavaPp {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\sreekanth\\Downloads\\chromedriver\_win32\\chromedriver.exe");

WebDriver obj = **new** ChromeDriver ();

obj.get("http://www.tizag.com/javascriptT/javascriptheadnbody.php");

obj.findElement(By.*xpath*("//input[@value='popup']")).click();

System.***out***.println(obj.switchTo().alert().getText());//to get text on popup

//we can't test popups which is built in java

//in that case we can use below statement

//we are swiching webdriver to java

obj.switchTo().alert().accept(); //we use accept for ok or yes or any possitive

} // we can use for cancel or negitives " Dismis"

}

**write code to verify any application login page is working or not**

**(u should write code to use textbox, button click events)**

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class FbLogin {

public static void main(String []args)

{

System.setProperty("webdriver.gecko.driver", "C:\\Program Files\\geckodriver-v0.13.0-win64\\geckodriver.exe");

WebDriver obj=new FirefoxDriver ();

obj.get("http://facebook.com");

obj.findElement(By.id("email")).sendKeys(“srisrikanth@gmail.com");

obj.findElement(By.name("pass")).sendKeys("srikanth");

obj.findElement(By.xpath(".//\*[@id='loginbutton']")).click();

}}

how to select items from dropdown/select box

Ex: write code

**how to know if checkbox is checked or not in webpage**

**public** **class** HandlingDynamicDropDown {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

System.*setProperty*("webdriver.chrome.driver", "C:\\Users\\sreekanth\\Downloads\\chromedriver\_win32\\chromedriver.exe");

WebDriver obj = **new** ChromeDriver ();

obj.get("http://spicejet.com/");

// Handling Checkboxes

// isSelected() method is have boolean expression

// it will print True if the check box is checked else false

obj.findElement(By.*xpath*(".//\*[@id='ctl00\_mainContent\_chk\_IndArm']")).click();

System.***out***.println(obj.findElement(By.*xpath*(".//\*[@id='ctl00\_mainContent\_chk\_IndArm']")).isSelected());

}

}

**write code to find out if all links are working or not**

String allLinks[]=selenium.getAllLinks();  
for(int i=0;i<allLinks.length;i++){  
  
  selenium.click(allLinks[i]);  
  Thread.sleep(3000);  
     
  }  
---------------------------------

import java.util.ArrayList;

import java.util.List;

import [org.openqa.selenium.By](http://org.openqa.selenium.by/" \t "_blank);

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.firefox.FirefoxDriver;

public class FindBrokenLinks {

 public static void main(String[] args) throws InterruptedException {

  WebDriver driver = new FirefoxDriver();

  driver.get("<https://www.xyz.com/>");

  Thread.sleep(5000L);

  int size = driver.findElements(By.tagName("a")).size();

  System.out.println(size);

  List<String> Linkarray = new ArrayList<String>();

      List<WebElement> Linklist = driver.findElements(By.tagName("a"));

      for (WebElement link : Linklist) {

       String links = link.getText();

       Linkarray.add(links );

      }

      for (String linkToTest : Linkarray){

       driver.findElement(By.linkText(linkToTest)).click();

       Thread.sleep(15000L);

    if(driver.getTitle().contains("Problem")) {

     System.out.println("Fail");

    }

    else

    {

     System.out.println("pass");

    }

    driver.navigate().back();

    Thread.sleep(5000L);

   }

  driver.close();

}

**write code on how to use javascript executor?**

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class JavaScriptExecuter {

public static void main(String[] args) {

WebDriver driver = new FirefoxDriver();

//Launching the browser application

driver.get("http://www.uftHelp.com");

//Adding wait

driver.manage().timeouts().implicitlyWait(20, TimeUnit.SECONDS);

//Maximize window

driver.manage().window().maximize();

//Creating the Javascriptexecutor interface object by Type casting

JavascriptExecutor js = (JavascriptExecutor)driver;

//Fetching the Domain Name

String sDomain = js.executeScript("return document.domain;").toString();

System.out.println("Domain = "+sDomain);

//Fetching the URL

String sURL = js.executeScript("return document.URL;").toString();

System.out.println("URL = "+sURL);

//Fetching the Title

String sTitle = js.executeScript("return document.title;").toString();

System.out.println("Title = "+sTitle);

//Vertical scroll - down by 200 pixels

js.executeScript("window.scrollBy(0,200)");

System.out.println("Successfully did the vertical scroll by 200px");

}

}

difference between assert and verify?

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import junit.framework.Assert;

public class AssertsDemo {

public static void main(String[] args)

{

// TODO Auto-generated method stub

System.setProperty("webdriver.gecko.driver", "c://geckodriver.exe");

WebDriver driver=new FirefoxDriver();

driver.get("https://www.makemytrip.com/flights");

System.out.println("before clicking");

Assert.assertTrue(driver.findElement(By.xpath(".//\*[@id='return\_date\_sec']")).isDisplayed());

driver.findElement(By.xpath(".//\*[@id='multi\_city\_button']/span")).click();

System.out.println("after clicking");

Assert.assertTrue(driver.findElement(By.xpath(".//\*[@id='return\_date\_sec']")).isDisplayed());

}

}

verify

package basics;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import junit.framework.Assert;

public class VerifyDemo {

public static void main(String[] args)

{

// TODO Auto-generated method stub

System.setProperty("webdriver.gecko.driver", "c://geckodriver.exe");

WebDriver driver=new FirefoxDriver();

driver.get("https://www.makemytrip.com/flights");

if(driver.getPageSource().contains("Text –Need Hotels"))

{

System.out.println("Text is Present");

}

else

{

System.out.println("Text is not Present");

}

}}

difference between driver.close and driver.quit methods?

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class Locator1

{

public static void main(String[] args)

{

System.setProperty("webdriver.gecko.driver", "c://geckodriver.exe");

WebDriver driver=new FirefoxDriver();

driver.get("https://www.facebook.com");

driver.findElement(By.xpath(".//\*[@id='wrapper']/div[2]/div/div[1]/p/a")).click();

driver.quit();

}

}

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class Locator1

{

public static void main(String[] args)

{

System.setProperty("webdriver.gecko.driver", "c://geckodriver.exe");

WebDriver driver=new FirefoxDriver();

driver.get("https://www.facebook.com");

driver.findElement(By.xpath(".//\*[@id='wrapper']/div[2]/div/div[1]/p/a")).click();

driver.close();

}

}

**common exceptions in selenium?**

public void findSearchTextBox()   
{   
 By incorrectSearchTextBoxLocator = By.xpath("//input[@id='globalQuery1']");  
 driver.get("http://www.vpl.ca");   
 try  
 {   
  WebElement searchField = driver.findElement(incorrectSearchTextBoxLocator);   
 }   
 catch(NoSuchElementException ex)   
 {   
   fail("search text box not found");   
 }   
}

[@Test](https://huddle.eurostarsoftwaretesting.com/members/test/)  
public void findSearchTextBox() throws IOException, ElementNotFoundException   
{   
 By incorrectSearchTextBoxLocator = By.xpath("//input[@id='globalQuery1']");  
  
 driver.get("http://www.vpl.ca");   
 try  
 {   
   WebElement searchField = driver.findElement(incorrectSearchTextBoxLocator);   
 }   
 catch(NoSuchElementException exception)   
 {   
   openLog();   
   try   
   {   
     addToLog(exception.getMessage());   
   }   
   catch (IOException ex)   
   {   
     throw new ElementNotFoundException("search text box not found", exception);   
   }   
 }   
 finally   
 {   
   closeLog();   
 }   
}

how to handle Ajax calls in selenium?

public void waitForAjaxControls(int timeoutInSeconds) {

System.out

.println("Querying active AJAX controls by calling jquery.active");

try {

if (browser instanceof JavascriptExecutor) {

JavascriptExecutor jsDriver = (JavascriptExecutor) browser;

for (int i = 0; i < timeoutInSeconds; i++) {

Object numberOfAjaxConnections = jsDriver

.executeScript("return jQuery.active");

// return should be a number

if (numberOfAjaxConnections instanceof Long) {

Long n = (Long) numberOfAjaxConnections;

System.out

.println("Number of active jquery AJAX controls: "

+ n);

if (n.longValue() == 0L)

break;

}

Thread.sleep(1000);

}

} else {

System.out.println("Web driver: " + browser

+ " can't run javascript.");

}

} catch (InterruptedException e) {

System.out.println(e);

}

}

How to assign the value to textbox other than sendkeys method?

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class Locator1

{

public static void main(String[] args)

{

System.setProperty("webdriver.gecko.driver", "c://geckodriver.exe");

WebDriver driver=new FirefoxDriver();

driver.get("http://www.qajudge.com/");

WebElement cssValue= driver.findElement(By.xpath(".//\*[@id='s']"));

JavascriptExecutor jse = (JavascriptExecutor) driver;

jse.executeScript("document.getElementById('s').value='Virender Testing

sending'");

**Selenium grid, how to execute scripts on multiple browser**

public enum Environments {

FF\_18\_WIN7("firefox", "18", Platform.WINDOWS),

CHR\_24\_WIN7("chrome", "24", Platform.WINDOWS),

IE\_9\_WIN7("internet explorer", "9", Platform.WINDOWS)

;

private final DesiredCapabilities capabilities;

private final String browserName;

private final String version;

private final Platform platform;

Environments(final String browserName, final String version, final Platform platform) {

this.browserName = browserName;

this.version = version;

this.platform = platform;

capabilities = new DesiredCapabilities();

}

public DesiredCapabilities capabilities() {

capabilities.setBrowserName(browserName);

capabilities.setVersion(version);

capabilities.setPlatform(platform);

return this.capabilities;

}

public String browserName() {

return browserName;

}

}