**Selenium Understanding Document**

**Selenium Sample Commands:**

**Command                                               Description**  
**driver.get("http://www.google.com");**  To open an application  
**driver.findElement(By.id("passwd-id"));**  Finding Element using Id  
**driver.findElement(By.name("passwd"));**  Finding Element using Name  
**driver.findElement(By.xpath("//input[@id=’passwd-id’]"));**  Finding Element using Xpath  
**element.sendKeys("some text");**  To type some data  
**element.clear();**  clear the contents of a text ﬁeld or textarea  
**driver.findElement(By.xpath("//select"));**  Selecting the value  
**select.findElements(By.tagName("option"));**  Selecting the value  
**select.deselectAll();**  This will deselect all OPTIONs from the ﬁrst SELECT on the page  
**select.selectByVisibleText("Edam");**  select the OPTION withthe displayed text of “Edam”  
**findElement(By.id("submit")).click();**  To click on Any button/Link  
**driver.switchTo().window("windowName");**  Moving from one window to another window  
**driver.switchTo().frame("frameName");**  swing from frame to frame (or into iframes)  
**driver.switchTo().frame("frameName.0.child");**  to access subframes by separating the path with a dot, and you can specify the frame by its index too.  
**driver.switchTo().alert();**  Handling Alerts  
**driver.navigate().to("http://www.example.com");**  To Navigate Particular URL  
**driver.navigate().forward();**  To Navigate Forward  
**driver.navigate().back();**  To Navigate Backword  
**driver.close()**  Closes the current window  
**driver.quit()**   Quits the driver and closes every associated window.  
**driver.switch\_to\_alert()**  Switches focus to an alert on the page.  
**driver.refresh()**  Refreshes the current page.  
**driver.implicitly\_wait(30)**  Amount of time to wait  
**driver.set\_script\_timeout(30)**  The amount of time to wait  
**driver.get\_screenshot\_as\_file('/Screenshots/foo.png')**  The full path you wish to save your screenshot to  
**driver.get\_screenshot\_as\_base64()**  Gets the screenshot of the current window as a base64 encoded string which is useful in embedded images in HTML

**Simple Launch Firefox browser:**

WebDriver webDriver = new FirefoxDriver();

// Setting the browser size

webDriver.manage().window().setSize(new Dimension(1024, 768));

// Go to wikipedia

webDriver.navigate().to("https://en.wikipedia.org/wiki/Main\_Page");

// Type in the search-field: "WebDriver"

webDriver.findElement(By.id("searchInput")).sendKeys("WebDriver");

// submitting the search query

webDriver.findElement(By.id("searchInput")).submit();

// Test if Wikipedia redirects to the correct article:// "Selenium (software)"

String textFound = webDriver.findElement(By.cssSelector("h1")).getText();

if (textFound.contains("Selenium (software)")) {

System.out.println("Test passes!");

} else {

System.out.println("Test fails!");

}

// Waiting a little bit before closing

Thread.sleep(7000);

// Closing the browser and webdriver

webDriver.close();

webDriver.quit();

}}

**NAVIGATION COMMANDS:**

1. **Browser Back and Forward (NAVIGATION)**

Steps to implement Browser back and forward through Selenium Web Driver

1. Create Driver for any Browser(Mozilla)

2. Go to the URL

3. Navigate to some page in website.

4. Use Selenium code to Navigate Back to Main Page.

**CODE:**

driver.navigate().back();

driver.navigate().forward();

**Example**

WebDriver driver =new FirefoxDriver();

driver.get("http://seleniumhq.org/");

driver.findElement(By.linkText("Download")).click();

Thread.sleep(3000);            //delay

driver.navigate().back();

driver.navigate().forward();

**2.Single selection dropdown list**

**WebElement element = driver.findElement(By.name("selectedCustomer"));  
Select dd= new Select(element);  
List allOptions= dd.getOptions();**  
//To go through the list, we can use an Iterator.   
//Iterator should be of the same type as the List  
//which is WebElement in this case.   
  
**Iterator it = allOptions.iterator();**  
//Using while loop, we can iterate till the List has   
//a next WebElement [hasNext() is true]  
//number of items in the list  
**System.out.println(allOptions.size());**  
  
**while(it.hasNext()){**//When you say it.next(), it points to a particular  
//WebElement in the List.  
**WebElement el = it.next();** //Check for the required element by Text and click it  
**if(el.getText().equals("mango")){  
  System.out.println(el.getAttribute("value"));  
   el.click();   
 }**}

**WebElement customerdd =driver.findElement(By.name("customerProject.shownCustomer"));**

**//convert the element to select object**

Select cust = new Select(**customerdd**);

**cust.selectByIndex(1);**                                       //**Select by** **Index**

Thread.sleep(3000);

**cust.selectByValue("2");                                   //Select by Value**

Thread.sleep(3000);

**cust.selectByVisibleText("mango");                //Select by Visible Text**

**Multiple select**

WebElement userdd = driver.findElement(By.name("users"));  
**Select usr = new Select(userdd);  
usr.selectByIndex(0);                     //Select by Index(From Start location)  
usr.selectByIndex(2);                     //Select by index(To End Location)**

**Deselect All:**

//You can deselect the options  
usr.**deselectAll**();                                          //Deselect ALL selected elements  
usr.deselectByIndex(0);                              //Deselect By using Index  
usr.deselectByValue(value);                       //Deselect By using Value  
usr.deselectByVisibleText(text);                 //Deselect By using Text  
-----------------------------------------------------------------------------------------------------------------------------

**IFRAMES - How to handle Frames in Web Driver**

|  |
| --- |
| ChromeOptions options = new ChromeOptions();  options.addArguments("test-type");  options.addArguments("start-maximized");  options.addArguments("--js-flags=--expose-gc");  options.addArguments("--enable-precise-memory-info");  options.addArguments("--disable-popup-blocking");  options.addArguments("--disable-default-apps");  options.addArguments("test-type=browser");  options.addArguments("disable-infobars");  driver = new ChromeDriver(options);  driver.manage().window().maximize();  driver.get("http://timesofindia.indiatimes.com/");  driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);  List forms= driver.findElements(By.tagName("iframe")); //Frame List  System.out.println(forms.size());  for(int i=0;i<forms.size();i++)  {  System.out.println(forms.get(i).equals(driver.getPageSource().contains("src")));  } |

**To Perform actions in frames:**

WebElement ifr = driver.findElement(By.xpath("//iframe[@src='/poll.cms']"));

**driver.switchTo().frame(ifr);**                             **//Switch to iFrame**driver.findElement(By.id("test")).sendKeys("8");  //Perform Action in iFrame

**Steps to switch to particular iFrame by index through Selenium Web Driver.**

driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);  
**List forms= driver.findElements(By.tagName("iframe"));**System.out.println(forms.size());  
**driver.switchTo().frame(0);**driver.findElement(By.id("clicktripad")).click();

**Steps to iterate through the Window Handles**

1. Create Driver for any Browser(Mozilla)

2. Go to the URL  
3. Collect Window Handles through Set  
4. Create an iterator to iterate through Window Handles.  
5. At First iterator will not be pointing to any Window Handle, only First increment Points to First Window Handle, Second increment Points to second iterator.  
Set windowHandles = driver.getWindowHandles();  
Iterator it = windowHandles.iterator();  
while(it.hasNext())  
{  
 System.out.println(it.next());

}

**When two browsers are opened and Web Driver need to shift the control from Parent Window to Child Window.**

1. Create Driver for any Browser (Mozilla)

2. Go to the URL  
3. Collect Window Handles through Set  
4. Create an iterator to iterate through Window Handles.  
5. Increment the iterator and store the Window Handle as Parent.  
6. Increment the iterator and store next Window Handle as Child.  
7. Switch to Child Browser using Child Window Handle.  
Set windowHandles = driver.getWindowHandles();  
Iterator it = windowHandles.iterator();  
  
String parentBrowser= it.next();  
String childBrowser = it.next();  
driver.switchTo().window(childBrowser);

**Close the pop up window and switch to parent window**

Set windowHandles = driver.getWindowHandles();  
Iterator it = windowHandles.iterator();  
String parentBrowser= it.next();  
String childBrowser = it.next();  
driver.switchTo().window(childBrowser);   
Thread.sleep(3000);  
driver.close(); //close the current window(Child Browser)  
driver.switchTo().window(parentBrowser); //Switch to Parent Browser

**Calendar PopUp - 1**

Normal Calendar (current month) Popup can be handled in the following way.

/\*IRCTC calendar\*/  
driver.findElement(By.id("calendar\_icon1")).click();   
driver.findElement(By.xpath("//div[@id='CalendarControl']/table[tbody[tr[td[text()='October 2012']]]]/descendant::a[text()='5']")).click();

**/\*makemytrip calendar\*/**driver.get("http://www.makemytrip.com/");  
driver.manage().timeouts().implicitlyWait(30, TimeUnit.SECONDS);  
driver.findElement(By.id("deptDateRtripimgExact")).click(); //find Calendar  
driver.manage().timeouts().implicitlyWait(0, TimeUnit.SECONDS);  
boolean flag=true;  
while(flag){  
  try {  
 WebElement el = driver.findElement(By.xpath("//div[contains(@class,'ui-datepicker-group') and descendant::span[text()='March']]/descendant::a[text()='5']")); // Required future date  
 if(el !=null)   //Check if the required date element is found or not  
       {  
 el.click(); // if required Date is found, then click  the date  
 flag=false;  
 }  
    }   
catch (Exception e) { //Catches exception if no element found  
try {  
 Thread.sleep(500);  
 driver.findElement(By.xpath("//a[@title='Next']")).click(); //Click on next month  
 }  
catch (InterruptedException e1)   
      {  
 // TODO Auto-generated catch block  
  e1.printStackTrace();  
      }  
**Drop Down Menu:**

WebElement parentMenu = driver.findElement(By.linkText("Tourist Trains"));

Actions act = new Actions(driver); // Create an Action object  
//move to the parent menu item

act.moveToElement(parentMenu).build().perform();

Thread.sleep(3000);   //wait till the child items are displayed

driver.findElement(By.linkText("Bharat Tirth")).click();

### Context Click (Right Click)

WebElement parentMenu = driver.findElement(By.linkText("Tourist Trains"));  
Actions act = new Actions(driver); //Create Action object for Driver  
act.contextClick(parentMenu).build().perform(); //Context Click  
act.sendKeys(Keys.ARROW\_RIGHT).build().perform();  
Thread.sleep(1000);  
act.sendKeys(Keys.ARROW\_DOWN).build().perform();  
Thread.sleep(1000);  
act.sendKeys(Keys.ENTER).build().perform();

**Use JavaScript to perform some actions**

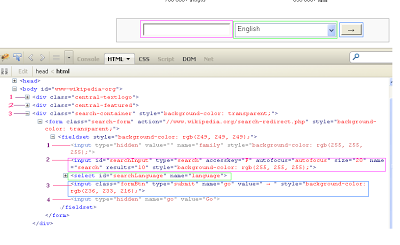
JavascriptExecutor js = (JavascriptExecutor) driver;  
String jsCmd = "document.getElementsByName('city')[0].value='ban'";  
js.executeScript(jsCmd);

**Use Alerts**

WebDriver driver = new FirefoxDriver();  
driver.get("http://www.2shared.com/");  
driver.manage().timeouts().implicitlyWait(3,TimeUnit.MINUTES);  
  
Alert alert = driver.switchTo().alert();  
alert.accept();  
//or  
alert.dismiss();

**Xpath**

Xpath in XML document shows the direction of element location through nodes and attributes. Let we try to understand how to identify Xpath of element with examples.

[](http://2.bp.blogspot.com/-9T8oQ-2SPbA/UbfoKtQdYFI/AAAAAAAAAGo/fsixYo0R5ss/s1600/Xpath.PNG)

Above given image is taken from http://www.wikipedia.org/. Look into the image there are three fields 1. Input text box 2. select drop down and 3. input button. And bellow of those fields there is expansion of relative XML nodes through firebug. As you see in image, you can use "id=searchInput" or "name=search" to identify input text box to type something in to it as bellow given example.

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| type | id=searchInput | ID Example |

or

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| type | name=search | Name Example |

**Xpath Tutorials**

Now if you want to identify same element (input textbox) with xpath then you can use any of the bellow given syntax in to the target column with type command in above example.

**Locating element using Xpath with Examples for input text box**

**1. Identifying Xpath using full path of XML**

**xpath=//body/div[3]/form/fieldset/input[2]**  //// Here //body is the main root node, /div[3] describes the 3rd div child node of parent node body, /form describes the child node form of parent node div[3], /fieldset describes the child node fieldset of parent node form, /input[2] describes the 2nd input child node of parent node fieldset.

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| type | xpath=//body/div[3]/form/fieldset/input[2] | Xpath Example1 |

**2. Writting Xpath using last()**

**xpath=//body/div[3]/form/fieldset/input[last()-2]**  //// Here /input[last()-2] describes the 3rd upper input node(input[2]) from last input node.  
**xpath=//body/div[3]/form/fieldset/\*[last()-3]**//// Here /\*[last()-3] describes the 4th upper  node(input[2]) from last node.

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| type | xpath=//body/div[3]/form/fieldset/input[last()-2] | Xpath Example2 |

**3. Xpath locator using @ and attribute**

**xpath=//body/div[3]/form/fieldset/input[@type='search']**   //// Here /input[@type='search'] describes the input node having attribute type='search'.

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| type | xpath=//body/div[3]/form/fieldset/input[@type='search'] | Xpath Example3 |

**4. Xpath expression using @ and attribute**

**xpath=//body/div[3]/form/fieldset/input[@accesskey='F']**   //// Here /input[@accesskey='F'] describes the input node having attribute @accesskey='F'. Another way of same is as bellow.

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| type | xpath=//body/div[3]/form/fieldset/input[@accesskey='F'] | Xpath Example4 |

**5. Xpath syntax using @ and attribute**

**xpath=//input[@accesskey='F']**  //// Here //input[@accesskey='F'] describes the input node having attribute @accesskey='F'. Try it by using it in above example.

**6. Xpath example using @ and attribute**

**xpath=//input[@type='search']**   //// Here /input[@type='search'] describes the input node having attribute type='search'. Try it by using it in above example.

**7. XML Xpath using /descendant:: keyword**

**xpath=//div[@class='search-container']/descendant::input[@accesskey='F']**   //// Here i have used descendant in between. In this case i have described only starting node div with attribute class='search-container' and final node input with accesskey='F' attribute. So not need to describe in between nodes. Try it by using it in above example.

**8. Xpath query example using contains keyword**

**xpath=//input[contains(@id, "searchInput")]**   ////Here i have used contains keyword to identify id attribute with text "searchInput". Try it by using it in above example.

**9. xpath using and with attributes**

**xpath=//input[contains(@id, "searchInput") and contains(@accesskey,"F")]**   ////In this example, It will look at two attributes in input node. Try it by using it in above example.

**10. XML xpath value value using position()**

**xpath=//div[@class='search-container']/descendant::input[position()=2]**  ////This xpath will select input node which is on number 2 position and it is for input text box as shown in image. Try it by using it in above example.

**11. Using starts-with keyword**  
**xpath=//input[starts-with(@type, "s")]    ////**In this example, It will find input node with attribute is 'type' and its value is starting with 's' (here it will get type = 'search').

**12. Using OR (|) condition with xpath**  
**xpath=//input[@accesskey='F'] | //input[@id='searchInput']**  
**xpath=//input[@accesskey='F' or @id='searchInput']**  //// In both these example, it will find input text box with accesskey='F' or @id='searchInput'. If any one found then it will locate it. Very useful when elements appears alternatively.  
  
**13. Using wildcard \* with to finding element xpath**  
**xpath=//\*[@accesskey='F']**  
  
**14. Finding nth child element of parent**  
**xpath=//body/\*[3]/form/fieldset/\*[2]**    ////This xpath is for search text box. Here, /\*[3] describes the 3rd child element of body which is div[3]. Same way \*[2] describes the 2nd child element of fieldset which is input[2]  
  
All above examples are for input text box. Now let me write Xpath for drop down.

**Xpath Examples for drop down**

**1. xpath=//body/div[3]/form/fieldset/select**

**2. xpath=//body/div[3]/form/fieldset/select[last()]**

**3. xpath=//body/div[3]/form/fieldset/select[@id='searchLanguage']**

**4. xpath=//body/div[3]/form/fieldset/select[@name='language']**

**5. xpath=//div[@class='search-container']/descendant::select[@name='language']**

**6. xpath=//select[contains(@id, "searchLanguage")]**

**7. xpath=//div[@class='search-container']/descendant::select[position()=1]**  
**8. xpath=//body/div[3]/form/fieldset/select[count(\*)>1]**

| New Test | | |
| --- | --- | --- |
| **Command** | **Target** | **Value** |
| open | http://www.wikipedia.org/ |  |
| select | xpath=//div[@class='search-container']/descendant::select[position()=1] | label=English |

**Other Xpath Example**

**1. Finding xpath for target link 'url'**  
**//a[@href='//meta.wikimedia.org/wiki/List\_of\_Wikipedias']**////This xpath example will find link with given URL (//meta.wikimedia.org/wiki/List\_of\_Wikipedias) on the page.  
  
**2. Finding xpath of element with no child**  
**xpath=//img[count(\*)=0]**////This xpath is for wikipedia text logo which is display on top of the page. This xpath will find that image element which have not any child element. Here image node is last and it has not any child element.  
  
**xpath=//div[2]/descendant::img[count(\*)=0]   ////**This xpath is for wikipedia logo image which is display under logo text.

**We can count the number of links present in the page. We can also print the link text of each Web link.**

List allLinks= driver.findElements(By.xpath("//a"));  
//display the count of links in the page  
System.out.println(allLinks.size());  
//display the text for each link on the page  
for(int i=0;i< allLinks.size();i++)  
{ //display href for each link  
 System.out.println(allLinks.get(i).getAttribute("href"));  
 //display text for each link  
 System.out.println(allLinks.get(i).getText());  
 allLinks.get(i).click();  
}

**Proxy Settings:**

1. Import Selenium.Proxy

2. Create a Profile object for Firefox

3. Create a string variable with value.

4. Create a Proxy object.

5. Set the values through proxy.

6. Set the proxy preference to proxy object using profile object.

7. Pass the profile object to Firefox Driver.

import org.openqa.Selenium.Proxy  
FirefoxProfile profile = new FirefoxProfile();  
String PROXY = "xx.xx.xx.xx:xx";  
Proxy proxy = new Proxy();  
proxy.HttpProxy=PROXY;  
proxy.FtpProxy=PROXY;  
proxy.SslProxy=PROXY;  
profile.SetProxyPreferences(proxy);  
FirefoxDriver driver = new FirefoxDriver(profile);

**File Upload**

WebDriver driver = new FirefoxDriver();  
driver.get("http://www.2shared.com/");  
String FilePath = "C:\\Users\\abc\\Desktop\\test.xml";  
driver.findElement(By.id("upField")).sendKeys(FilePath);  
driver.findElement(By.xpath("//input[@type='image']")).click();

**File Download**

FirefoxProfile Prof = new FirefoxProfile();  
Prof.setPreference("browser.download.dir", "D:\\java prj");  
Prof.setPreference("browser.download.folderList", 2);  
Prof.setPreference("browser.helperApps.neverAsk.saveToDisk","application/zip");  
   
WebDriver driver = new FirefoxDriver(Prof);  
driver.get("http://seleniumhq.org/download/");  
driver.manage().timeouts().implicitlyWait(3,TimeUnit.MINUTES);  
driver.findElement(By.xpath("//a[@name='client-drivers']/table/tbody/tr[1]/td[4]/a")).click();

**Web Driver with Excel**import java.io.FileInputStream;  
import jxl.Sheet;  
import jxl.Workbook;  
import org.openqa.selenium.By;  
import org.openqa.selenium.WebDriver;  
import org.openqa.selenium.firefox.FirefoxDriver;  
import org.testng.annotations.AfterClass;  
import org.testng.annotations.BeforeClass;  
import org.testng.annotations.Test;  
public class WebDriverExcel {  
private WebDriver driver;  
@BeforeClass  
public void Startup(){  
driver = new FirefoxDriver();  
}  
@Test (description="login")  
public void Login() throws Exception{  
FileInputStream fi=new FileInputStream("D:\\Selenium\\Test.xls");  
Workbook w=Workbook.getWorkbook(fi);  
Sheet s=w.getSheet(0);  
driver.get("http://127.0.0.1/login");  
try  
{  
for (int i = 1; i < s.getRows(); i++)  
{  
//Read data from excel sheet  
String s1 = s.getCell(0,i).getContents();  
String s2 = s.getCell(1,i).getContents();  
driver.findElement(By.name("txtUserName")).sendKeys(s1);  
driver.findElement(By.name("txtPassword")).sendKeys(s2);  
driver.findElement(By.name("Submit")).click();  
Thread.sleep(3000);  
if(driver.getTitle().equals("OrangeHRM"))  
{  
System.out.println("admin page Displayed successfully");  
}  
else  
{  
System.out.println("Do not Displayed");  
}  
driver.findElement(By.linkText("Logout"));  
}  
}  
catch(Exception e)  
{  
System.out.println(e);  
}  
}  
@AfterClass  
public void teardown(){  
driver.quit();  
}  
}

**Sample DB connection Snippet:**

import java.sql.\*;

import javax.sql.\*;

public class dbconnection

{

public static void main(String args[])

{

String email;

String dbUrl = "jdbc:mysql://localhost:3306/test";  //This URL is based on your IP address

String username="username"; //Default username is root

String password="password"; //Default password is root

String dbClass = "com.mysql.jdbc.Driver";

String query = "Select email from users where user\_id = 1;";

try

{

Class.forName(dbClass);

Connection con = DriverManager.getConnection (dbUrl,username,password);

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery(query);

while (rs.next()) {

dbtime = rs.getString(1);

System.out.println(email);

} //end while

con.close();

} //end try

catch(ClassNotFoundException e) {

e.printStackTrace();

}

catch(SQLException e) {

e.printStackTrace();

}

}  //end main

}  //end class

**Sample Practices 🡪Gmail automation**

|  |
| --- |
| System.*setProperty*("webdriver.chrome.driver","C://Users/sangee/Downloads/chromedriver\_win32 (1)/chromedriver.exe");  ChromeOptions options = **new** ChromeOptions();  options.addArguments("test-type");  options.addArguments("start-maximized");  options.addArguments("--js-flags=--expose-gc");  options.addArguments("--enable-precise-memory-info");  options.addArguments("--disable-popup-blocking");  options.addArguments("--disable-default-apps");  options.addArguments("test-type=browser");  options.addArguments("disable-infobars");  *driver* = **new** ChromeDriver(options);  *driver*.manage().window().maximize();  *driver*.get("https://accounts.google.com/ServiceLogin?");  // gmail login  *driver*.findElement(By.*id*("identifierId")).sendKeys("\*\*\*\*\*\*\*username\*\*\*\*");  *driver*.findElement(By.*xpath*("//\*[@id='identifierNext']/content/span")).click();  Thread.*sleep*(1000);  *driver*.findElement(By.*name*("password")).sendKeys("\*\*\*\*\*\*\*\*\*\*\*");  *driver*.findElement(By.*xpath*("//\*[@id='passwordNext']/content/span")).click();  System.***out***.println("Logged in successfully"); |