**Notes on Selenium:**

-> System.setProperty("",""); ==> to set the driver with its path

webdriver.chrome.driver ==> for chrome

webdriver.gecko.driver ==> firefox

webdriver.ie.driver ==> Internet Explorer

-> obj.get(""); ==> Url to call the site

-> obj.getTitle(); ==> To get the title of the website

-> obj.getCurrentUrl(); ==> To get current url of the site which we have called

-> obj.getPageSource(); ==> To inspect the website

-> obj.navigate().back(); ==> to navigate to previous page

-> obj.close(); ==> Close the driver

-> obj.quit(); ==> close all the browser opened by selenium script

-> Locators supported by Selenium Webdrivers:-

ID

ClassName

Name

LinkText

Xpath

Css

-> obj.findElement(By.locator("")).sendKeys(""); ==> to find the element by using Locators

-> Every object may not have ID, ClassName or Name- Xpath and Css preffered

-> Alpha numeric id may vary on every refresh- check

-> Confirm the link object with anchor "a" tag

-> Multiple values -Selenium identifies the first one- Scans from top left

-> Double quotes inside double quoates are not accepted

-> obj.findElement(By.Xpath()).click(); ==> to click the button

-> Firepath depriciated from firefox.

-> //tagName[@attribute='value'] or //\*[@attribute='value'] ==> xpath syntax.

-> $x("") ==> tocheck xpath is true or not in console of browser

-> tagName[attribute='value'] or \*[attribute='value'] or tagName#id or tagName.classname ==> CSS syntax.

-> $("") ==> tocheck CSS is true or not in console of browser

-> firefox browser may give error on CSS path even if it is true.

-> //tagName[contains(@attribute,'value')] ==> xpath syntax for regular expression

-> //tagName[attribute\*='value'] ==> CSS syntax for regular expression

-----------------------------------------------

practice website:-

rahulshettyacademy.com

-----------------------------------------------

-> Parent-child relationship xpath ==> define xpath for parent/tagNames

-> Chropath ==> chrome extention which gives xpath & CSS directly

-----------------------------------------------

qaclickacademy.com

-----------------------------------------------

-> Relative xpath ==> Does not depend on parent

-> Absolute xpath ==> depend on parent

-> Traverse to sibling node using xpath ==> xpath/following-sibling::tagName[]

-> Traverse back to parent node using xpath ==> xpath/parent::tagName

-> Identifying objects with text using xpath locators ==> //\*[text()=' text '] ==> text may contain space ahead and at back.

-> CSS customized xpath

-> For static dropdown:-

Select s = new Select(driver.locator(""));

> s.selectByValue("") ==> use vale to select

> s.selectByIndex() ==> take index of the value (index starts wih 0)

> s.selectByVisibleText("") ==> use text to select the item

-> For current UI dropdown:-

use while loop for selecting multiple value at same time for example 4 adults

-> For dynamic dropdown:-

obj.findElement(By.locator("(xpath)[]")).event

also we can use parent-child relationship locator ==> parent xpath single space child xpath

-> for pressing enter using sendKeys ==> sendKeys(Keys.ENTER);

-> obj.findElement(By.locator()).isSelected() ==> For checkbox Returns true if selected else false

-> obj.findElement(By.locator()).Size() ==> For checkbox Returns total number of them present

-> Testng for assertion

-> Assert.assertFalse() ==> To check wether false or not

-> Assert.assertTrue() ==> to check wether true or not

-> Assert.assertEquals() ==> to check actual is same as expected

-> for alert ==> obj.switchTo().alert().accept(); ==> it accepts the alert and close it

-> obj.switchTo().alert().getText(); ==> to get alert text

-> obj.switchTo().alert().dismiss(); ==> it dismiss the alert and close it

-----------------------------------------

to reformat the code ==> ctr+shift+f

-----------------------------------------

-> class name should be starting with capital letter

-> variable name should start with small letter

-> naming convention should be in camel case

-> to execute script step by step right click on index and place toogle breakpoint

-> use stepover for further command

-> use stepinto to acess the method

-> for screenshot

public static void getScreenshot(WebDriver driver) throws IOException

{

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

File src = ((TakesScreenshot)driver).getScreenshotAs(OutputType.File);

FileUtils.copyFile(src,new File("location\\screenshot.png"));

}

-> use .split(""); to split the line into two parts using some letter or sp.char in the line

-> use .trim(); to remove spaces

synchronization in selenium

- implicit wait --> gloabally

- explicit wait --> specific

- Thread.Sleep

- Fluent wait

Implicit wait

driver.manage().timeouts().implicitlyWait(30,TimeUnit.SECONDS);

Explixit wait

driver.getElement(By.xpath("//\*[text()='Click to load get data via Ajax!']")).click;

WebDriverWait w =new WebDriverWait(driver,5);

w.until(ExpectedConditions.visibilityOfElementLocated(By.xpath("//\*[@id="results"]")));

Fluent wait

Wait<WebDriver> wait=new FluentWait<WebDriver>(driver).withTimeout(Duration.ofSeconds(30)).pollingEvery(Duration.ofSeconds(3)).ignoring(NoSuchElementException.class);

WebElement foo = wait.until(new Function<WebDriver, WebElement>(){

public WebElement apply(WebDriver driver){

if(driver.findElement(By.locator()).isDisplayed()){

return driver.findElement(By.locator());

}

else

return null;

}

});

Actions:--

Actions a = new Actions(diver);

WebElement move = driver.findElement(By.locator());

a.moveToElement(driver.findElement(By.locator)).build().perform();

Actions a = new Actions(diver);

WebElement move = driver.findElement(By.locator());

a.moveToElement(driver.findElement(By.locator)).click().keyDown(Keys.SHIFT).sendKeys("hello").doubleClick().build().perform();

a.moveToElement(move).build.perform();

use contextClick() as right click

switch betwen two screens/tabs/windows:--

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

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

String parentid = it.next();

String childid = it.next();

driver.switchTo().window(childid);

switch between frames:--

driver.switchTo().frame(driver.findElement(By.locator()));

drag and drop action:--

Actions a = new Actions(driver);

a.dragAndDrop(source,target).build().perform();

to press ctrl and tap on link

String clicke = Keys.chord(Keys.CONTROL,Keys.ENTER);

driver.findElements(By.locator()).sendKeys(click);

Javascript Hidden:-

JavascriptExecutor js = (JavascriptExecutor)driver;

String script = "return document.getElementById(\"autocomplete\").value;";

String text=(String) js.executeScript(script);

handling HTTP cerifications:-

DesiredCapabilities ch = DesiredCapabilitirs.chrome();

ch.acceptInsecureCerts();

ChromeOptions c = new ChromeOprions();

c.merge(ch);

WebDriver driver=newChromeDriver(c);

deleting the cookie:-

driver.manage().deleteAllCookies();

css to select particular column:-

tr td nth-child()

**Some basic Codes in JAVA:**

1. **Nested Frames:**

|  |
| --- |
| package xyz;  import java.util.concurrent.TimeUnit;  import org.openqa.selenium.By;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.chrome.ChromeDriver;  public class nestedframes {  public static void main(String[] args) {  System.setProperty("webdriver.chrome.driver", "C:\\Users\\chromedriver.exe");  WebDriver driver = new ChromeDriver();  driver.manage().timeouts().implicitlyWait(5,TimeUnit.SECONDS);  driver.get("https://the-internet.herokuapp.com/");  driver.findElement(By.xpath("//\*[text()='Nested Frames']")).click();  driver.switchTo().frame("frame-top");  driver.switchTo().frame("frame-middle");  System.out.println(driver.findElement(By.id("content")).getText());  }  } |

1. **For controlling Alert box:**

|  |
| --- |
| package xyz;  import java.util.concurrent.TimeUnit;  import org.openqa.selenium.By;  import org.openqa.selenium.Keys;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.WebElement;  import org.openqa.selenium.chrome.ChromeDriver;  import org.openqa.selenium.support.ui.Select;  //import com.sun.org.apache.bcel.internal.generic.Select;  public class Assignment6 {  public static void main(String[] args) throws InterruptedException {  System.setProperty("webdriver.chrome.driver", "C:\\Users\\chromedriver.exe");  WebDriver driver = new ChromeDriver();  driver.get("https://rahulshettyacademy.com/AutomationPractice/");  Thread.sleep(5000);  WebElement checkbox = driver.findElement(By.xpath("//\*[@id='checkbox-example']/fieldset/label[3]"));  driver.findElement(By.id("checkBoxOption3")).click();  Thread.sleep(2000);  System.out.println(checkbox.isSelected());  System.out.println(checkbox.getText());  String val = checkbox.getText();  System.out.println(val);  Select s = new Select(driver.findElement(By.id("dropdown-class-example")));  s.selectByVisibleText(val);  driver.findElement(By.name("enter-name")).sendKeys(val);  driver.findElement(By.id("alertbtn")).click();  System.out.println(driver.switchTo().alert().getText());  }  } |

1. **For collecting table data:**

|  |
| --- |
| package xyz;  import org.openqa.selenium.By;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.WebElement;  import org.openqa.selenium.chrome.ChromeDriver;  public class assignment7 {  public static void main(String[] args) {  System.setProperty("webdriver.chrome.driver", "C:\\Users\\chromedriver.exe");  WebDriver driver = new ChromeDriver();  driver.get("https://rahulshettyacademy.com/AutomationPractice/");  WebElement table = driver.findElement(By.xpath("//\*[@id=\'product\']"));  int rowCount = table.findElements(By.xpath("//\*[@id=\'product\']/tbody/tr")).size();  System.out.println("Total Rows:- "+rowCount);  int colCount = table.findElements(By.xpath("//\*[@id=\'product\']/tbody/tr/th")).size();  System.out.println("Total Cols:- "+colCount);  System.out.println("Second Row data:- "+driver.findElement(By.xpath("//\*[@id=\'product\']/tbody/tr[3]")).getText());  }  } |

1. **For dropdown (JavaScript was used):**

|  |
| --- |
| package xyz;  import org.openqa.selenium.By;  import org.openqa.selenium.JavascriptExecutor;  import org.openqa.selenium.Keys;  import org.openqa.selenium.WebDriver;  import org.openqa.selenium.chrome.ChromeDriver;  public class assignment8 {  public static void main(String[] args) {  System.setProperty("webdriver.chrome.driver", "C:\\Users\\chromedriver.exe");  WebDriver driver = new ChromeDriver();  driver.get("https://rahulshettyacademy.com/AutomationPractice/");  driver.findElement(By.id("autocomplete")).click();  driver.findElement(By.id("autocomplete")).sendKeys("ind");  JavascriptExecutor js = (JavascriptExecutor) driver;  String script = "return document.getElementById(\"autocomplete\").value;";  String text = (String) js.executeScript(script);  System.out.println(text);  while (!text.equalsIgnoreCase("INDIA")) {  driver.findElement(By.id("autocomplete")).sendKeys(Keys.ARROW\_DOWN);  text = (String) js.executeScript(script);  System.out.println(text);  }  }  } |

1. **For connecting Appium (used for mobile testing):**

|  |
| --- |
| package xyz;  import java.io.File;  import java.net.MalformedURLException;  import java.net.URL;  import org.openqa.selenium.remote.DesiredCapabilities;  import io.appium.java\_client.android.AndroidDriver;  import io.appium.java\_client.android.AndroidElement;  import io.appium.java\_client.remote.MobileCapabilityType;  public class Appium1 {  public static void main(String[] args) throws MalformedURLException {  File f = new File("src");  File fs = new File(f,"app-debug.apk");  DesiredCapabilities cap = new DesiredCapabilities();  cap.setCapability(MobileCapabilityType.DEVICE\_NAME, "Pixel\_3\_XL\_Edited\_1\_API\_29");  cap.setCapability(MobileCapabilityType.APP,fs.getAbsolutePath());  AndroidDriver<AndroidElement> driver = new AndroidDriver<AndroidElement>(new URL("http://127.0.0.1:4723/wd/hub"),cap);  }  } |

**Gathered much info from Udemy course :“**<https://www.udemy.com/course/selenium-real-time-examplesinterview-questions/>” with mentor Rahul Shetty.

**I have published an article on the same: “**<https://medium.com/@nevilparikh1410/basics-of-selenium-28f67220e256>”