 Create a basic automation framework which supports all browsers

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

import org.testng.annotations.Parameters;

import org.testng.annotations.Test;

public class TestCase1 {

@Test

// Here this parameters we will take from testng.xml

@Parameters("Browser")

public void test1(String browser) {

if(browser.equalsIgnoreCase("FF")){

WebDriver driver=new FirefoxDriver();

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

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

driver.quit();

}

else if(browser.equalsIgnoreCase("IE")){

System.setProperty("webdriver.ie.driver", "./server/IEDriverServer.exe");

WebDriver driver=new InternetExplorerDriver();

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

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

driver.quit();

}

}

}

Now, create]ing a TestNG.xml file to pass the parameters.

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">

Here parallel is an attribute which specify the mode of execution and thread-count specify how many browser should open

<suite name="Suite" parallel="tests" thread-count="2">

<test name="Test">

<parameter name="Browser" value="FF" />

<classes>

<class name="SampleTestcases.TestCase1"/>

</classes>

</test>

<test name="Test1">

<parameter name="Browser" value="IE" />

<classes>

<class name="SampleTestcases.TestCase1"/>

</classes>

</test>

</suite>

2)Automate Flipkart web application using the Selenium tool which should support in any Browser

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.openqa.selenium.ie.InternetExplorerDriver;

import org.testng.annotations.Parameters;

import org.testng.annotations.Test;

public class TestCase1 {

@Test

// Here this parameters we will take from testng.xml

@Parameters("Browser")

public void test1(String browser) {

if(browser.equalsIgnoreCase("FF")){

WebDriver driver=new FirefoxDriver();

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

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

driver.quit();

}

else if(browser.equalsIgnoreCase("IE")){

System.setProperty("webdriver.ie.driver", "./server/IEDriverServer.exe");

WebDriver driver=new InternetExplorerDriver();

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

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

driver.quit();

}

}

}

Now, creating a TestNG.xml file to pass the parameters.

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">

Here parallel is an attribute which specify the mode of execution and thread-count specify how many browser should open

<suite name="Suite" parallel="tests" thread-count="2">

<test name="Test">

<parameter name="Browser" value="FF" />

<classes>

<class name="SampleTestcases.TestCase1"/>

</classes>

</test>

<test name="Test1">

<parameter name="Browser" value="IE" />

<classes>

<class name="SampleTestcases.TestCase1"/>

</classes>

</test>

</suite>

3)Open the Flipkart portal from the link [https://www.flipkartcom](https://www.flipkartcom/)

@BeforeClass

public void launchBrowser()

{

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

driver = new ChromeDriver();

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

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

}

@AfterClass

public void closeBrowser()

{

driver.close();

driver.quit();

}

4) Use Grid, Javascript and different kinds of gestures like click, scroll, swipe other api’s available in Selenium

### Scenario 1: To scroll down the web page by pixel.

### import org.openqa.selenium.JavascriptExecutor;

### import org.openqa.selenium.WebDriver;

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

### import org.testng.annotations.Test;

### public class ScrollByPixel {

### WebDriver driver;

### @Test

### public void ByPixel() {

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

### driver = new ChromeDriver();

### JavascriptExecutor js = (JavascriptExecutor) driver;

### // Launch the application

### driver.get("http://demo.guru99.com/test/guru99home/");

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

### // This will scroll down the page by 1000 pixel vertical

### js.executeScript("window.scrollBy(0,1000)");

### }

### }

### Scenario 2: To scroll down the web page at the bottom of the page.

### import org.openqa.selenium.JavascriptExecutor;

### import org.openqa.selenium.WebDriver;

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

### import org.testng.annotations.Test;

### public class ScrollByPage {

### WebDriver driver;

### @Test

### public void ByPage() {

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

### driver = new ChromeDriver();

### JavascriptExecutor js = (JavascriptExecutor) driver;

### // Launch the application

### driver.get("http://demo.guru99.com/test/guru99home/");

### //This will scroll the web page till end.

### js.executeScript("window.scrollTo(0, document.body.scrollHeight)");

### }

### }

### **Test Cases On Selenium Grid:**

### **//DesiredCapabilites**is used to set the type of **browser** and **OS**that we will automate

* **RemoteWebDriver**is used to set which node (or machine) that our test will run against.

### import org.openqa.selenium.\*;

### import org.openqa.selenium.remote.DesiredCapabilities;

### import java.net.MalformedURLException;

### import java.net.URL;

### import org.openqa.selenium.remote.RemoteWebDriver;

### import org.testng.Assert;

### import org.testng.annotations.\*;

### public class SeleniumGrid {

### WebDriver driver;

### String baseURL, nodeURL;

### @BeforeTest

### public void setUp() throws MalformedURLException {

### baseURL = "http://demo.guru99.com/test/guru99home/";

### nodeURL = "http://192.168.43.223:4444/wd/hub";

### DesiredCapabilities capability = DesiredCapabilities.chrome();

### capability.setBrowserName("chrome");

### capability.setPlatform(Platform.WIN10);

### driver = new RemoteWebDriver(new URL(nodeURL), capability);

### }

### @AfterTest

### public void afterTest() {

### driver.quit();

### }

### @Test

### public void sampleTest() {

### driver.get(baseURL);

### 

### if (driver.getPageSource().contains("MOBILE TESTING")) {

### Assert.assertTrue(true, "Mobile Testing Link Found");

### } else {

### Assert.assertTrue(false, "Failed: Link not found");

### }

### }

### }

### **Double click scenerio**

### **package test;**

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

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

### **import org.openqa.selenium.WebElement;**

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

### **import org.openqa.selenium.interactions.Actions;**

### **import org.openqa.selenium.Alert;**

### **public class DobuleClickDemo {**

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

### **WebDriver driver;**

### **System.setProperty("webdriver.chrome.driver","X://chromedriver.exe");**

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

### **//Launch the Application Under Test (AUT)**

### **driver.get("http://demo.guru99.com/test/simple\_context\_menu.html");**

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

### **driver.get("http://demo.guru99.com/test/simple\_context\_menu.html");**

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

### **//Double click the button to launch an alertbox**

### **Actions action = new Actions(driver);**

### **WebElement link =driver.findElement(By.xpath("//button[text()='Double-Click Me To See Alert']"));**

### **action.doubleClick(link).perform();**

### **//Switch to the alert box and click on OK button**

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

### **System.out.println("Alert Text\n" +alert.getText());**

### **alert.accept();**

### **//Closing the driver instance**

### **//driver.quit();**

### **}**

### **}**

**5)Automation the scenario “Login to Flipkart portal and search for an item and add to cart and purchase it and logout”**

**package com.techbeamers.exercises;**

**import java.util.concurrent.TimeUnit;**

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

**import org.openqa.selenium.NoSuchElementException;**

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

**import org.openqa.selenium.WebElement;**

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

**import org.openqa.selenium.support.ui.ExpectedConditions;**

**import org.openqa.selenium.support.ui.FluentWait;**

**import org.openqa.selenium.support.ui.WebDriverWait;**

**import com.google.common.base.Function;**

**public class automateFlipkart {**

**public static WebDriver driver;**

**public static String driverPath = "C:\workspace\tools\selenium\";**

**public static String sBookKey = "Selenium";**

**private static final String sSearchBox = "fk-top-search-box";**

**private static final String sSearchResult = "//li[contains(text(),'in')]//span";**

**private static final String sBookName = "(//a[contains(.,'" + sBookKey + "')])[last()]";**

**private static final String sAddToCart = "(//input[contains(@value,'Add to Cart')])[1]";**

**private static final String sViewCartXPath = "(//a[contains(.,'view cart')])[1]";**

**public static void initWebDriver(String URL) throws InterruptedException {**

**// Setting up Chrome driver path.**

**System.setProperty("webdriver.chrome.driver", driverPath + "chromedriver.exe");**

**// Launching Chrome browser.**

**driver = new ChromeDriver();**

**driver.get(URL);**

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

**}**

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

**initWebDriver("http://www.flipkart.com");**

**flipkartLogin();**

**driver.findElement(By.id(sSearchBox)).sendKeys(sBookKey);**

**WebElement searchResult = getElement(By.xpath(sSearchResult));**

**searchResult.click();**

**WebDriverWait wait = new WebDriverWait(driver, 30);**

**wait.until(ExpectedConditions.elementToBeClickable(By.xpath(sBookName))).click();**

**wait.until(ExpectedConditions.elementToBeClickable(By.xpath(sAddToCart))).click();**

**getElement(By.xpath(sViewCartXPath)).click();**

**getElement(By.cssSelector("form[id='view-cart-form'] button")).click();**

**getElement(By.xpath("//input[@id='email' and @name='email']")).sendKeys("test@testmail.com");**

**// pause for a second and close the browser.**

**Thread.sleep(1000);**

**endSession();**

**}**

**public static WebElement getElement(final By locator) {**

**FluentWait<WebDriver> wait = new FluentWait<WebDriver>(driver).withTimeout(30, TimeUnit.SECONDS)**

**.pollingEvery(5, TimeUnit.SECONDS).ignoring(NoSuchElementException.class);**

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

**@Override**

**public WebElement apply(WebDriver arg0) {**

**return arg0.findElement(locator);**

**}**

**});**

**return element;**

**}**

**public static void flipkartLogin() {**

**driver.findElement(By.linkText("Log In")).click();**

**// TBD: Fill your username/password of flipkart.**

**getElement(By.cssSelector("input[placeholder='Enter email/mobile']")).sendKeys("");**

**getElement(By.cssSelector("input[placeholder='Enter password']")).sendKeys("");**

**getElement(By.cssSelector("input[value='Login'][class='submit-btn login-btn btn']")).click();**

**try {**

**Thread.sleep(1000);**

**} catch (InterruptedException e) {**

**// TBD: Auto-generated catch block.**

**e.printStackTrace();**

**}**

**}**

**public static void endSession() {**

**driver.close();**

**driver.quit();**

**}**

**}**

** Verify information (name/price/description...) on product search screen, comparing those info between product search screen vs checkout screen**

@Test

public void flipkartScript() throws InterruptedException {

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

driver.findElement(By.id("fk-top-search-box")).sendKeys("Asus Zenfone 5");

driver.findElement(By.xpath("//input[@value='Search']")).click();

// Selecting first product

driver.findElement(By

.xpath(".//\*[@id='products']/div/div[1]/div[1]/div/div[1]/a[1]"));

String name = driver.findElement(By.className("title")).getText();

String price = driver.findElement(By

.xpath("//span[@class='selling-price omniture-field']")).getText();

System.out.println("ProductName: "+name + " ProductPrice: "+price);

//Adding selected product to cart

driver.findElement(By.xpath("//input[@value='Add to Cart']")).click();

Thread.sleep(3000);

//Checking product is available in cart or not

driver.findElement(By.xpath("//span[@class='cart-label']")).click();

driver.findElement(By.xpath("//\*[@id='products']/div/div[1]/div[1]/div"))

.getAttribute("data-pid") //Id from first product in search result

driver.findElement(By.xpath("//\*[@class='compare-items']/div[1]"))

.getAttribute("item\_id")) //Id from product added to cart

//product detail page

String detail\_page=driver.findElement(By.xpath("//span[@class='subtitle']")).getText();

//cart page

String cart\_page=driver.findElement(By.xpath("//p[contains(@class,'fk-font-11')]")).getText();

Assert.assertEquals(detail\_page, cart\_page);

}



6)Read the data from external source

read an excel file using the POI API.

package excelExportAndFileIO;

import java.io.File;

import java.io.FileInputStream;

import java.io.IOException;

import org.apache.poi.hssf.usermodel.HSSFWorkbook;

import org.apache.poi.ss.usermodel.Row;

import org.apache.poi.ss.usermodel.Sheet;

import org.apache.poi.ss.usermodel.Workbook;

import org.apache.poi.xssf.usermodel.XSSFWorkbook;

public class ReadExcelFile {

public void readExcel(String filePath,String fileName,String sheetName) throws IOException{

//Create an object of File class to open xlsx file

File file = new File(filePath+"\\"+fileName);

//Create an object of FileInputStream class to read excel file

FileInputStream inputStream = new FileInputStream(file);

Workbook Workbook = null;

//Find the file extension by splitting file name in substring and getting only extension name

String fileExtensionName = fileName.substring(fileName.indexOf("."));

//Check condition if the file is xlsx file

if(fileExtensionName.equals(".xlsx")){

//If it is xlsx file then create object of XSSFWorkbook class

Workbook = new XSSFWorkbook(inputStream);

}

//Check condition if the file is xls file

else if(fileExtensionName.equals(".xls")){

//If it is xls file then create object of XSSFWorkbook class

Workbook = new HSSFWorkbook(inputStream);

}

//Read sheet inside the workbook by its name

Sheet Sheet = Workbook.getSheet(sheetName);

//Find number of rows in excel file

int rowCount = Sheet.getLastRowNum()-Sheet.getFirstRowNum();

//Create a loop over all the rows of excel file to read it

for (int i = 0; i < rowCount+1; i++) {

Row row = Sheet.getRow(i);

//Create a loop to print cell values in a row

for (int j = 0; j < row.getLastCellNum(); j++) {

//Print Excel data in console

System.out.print(row.getCell(j).getStringCellValue()+"|| ");

}

System.out.println();

}

}

//Main function is calling readExcel function to read data from excel file

public static void main(String...strings) throws IOException{

//Create an object of ReadExcelFile class

ReadExcelFile objExcelFile = new ReadExcelFile();

//Prepare the path of excel file

String filePath = System.getProperty("user.dir")+"\\src\\excelExportAndFileIO";

//Call read file method of the class to read data

objExcelFile.readExcel(filePath,"ExportExcel.xlsx","ExcelDemo");

}

}

7) Handle screen rotation efficiently.

import io.appium.java\_client.AppiumDriver;

import io.appium.java\_client.android.AndroidDriver;

import org.junit.After;

import org.junit.Before;

import org.junit.Test;

import org.openqa.selenium.ScreenOrientation;

import org.openqa.selenium.remote.DesiredCapabilities;

import org.openqa.selenium.support.ui.WebDriverWait;

import java.io.File;

import java.net.URL;

public class OCRNekoAtsumeTestSuite {

private AppiumDriver driver;

private WebDriverWait wait;

private OCR OCR;

private File imgDir;

@Before

public void setUp() throws Exception {

//Appium setup for the app

//needs to be installed on target device before the test

DesiredCapabilities capabilities = new DesiredCapabilities();

capabilities.setCapability("appPackage", "jp.co.hit\_point.nekoatsume");

capabilities.setCapability("appActivity", "jp.co.hit\_point.nekoatsume.GActivity");

capabilities.setCapability("deviceName", "Android Emulator");

capabilities.setCapability("platformVersion", "5.0.1");

driver = new AndroidDriver(new URL("http://0.0.0.0:4723/wd/hub"), capabilities);

wait = new WebDriverWait(driver, 10);

//Sikuli settings

OCR = new OCR(driver);

//location of screenshots

File classpathRoot = new File(System.getProperty("user.dir"));

imgDir = new File(classpathRoot, "src/main/resources");

//switch to native app + portrait mode

driver.context("NATIVE\_APP");

driver.rotate(ScreenOrientation.PORTRAIT);

}

Handling Browser Complatibility

import java.util.concurrent.TimeUnit;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.edge.EdgeDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Parameters;

import org.testng.annotations.Test;

public class CrossBrowserScript {

WebDriver driver;

/\*\*

\* This function will execute before each Test tag in testng.xml

\* @param browser

\* @throws Exception

\*/

@BeforeTest

@Parameters("browser")

public void setup(String browser) throws Exception{

//Check if parameter passed from TestNG is 'firefox'

if(browser.equalsIgnoreCase("firefox")){

//create firefox instance

System.setProperty("webdriver.firefox.marionette", ".\\geckodriver.exe");

driver = new FirefoxDriver();

}

//Check if parameter passed as 'chrome'

else if(browser.equalsIgnoreCase("chrome")){

//set path to chromedriver.exe

System.setProperty("webdriver.chrome.driver",".\\chromedriver.exe");

//create chrome instance

driver = new ChromeDriver();

}

//Check if parameter passed as 'Edge'

else if(browser.equalsIgnoreCase("Edge")){

//set path to Edge.exe

System.setProperty("webdriver.edge.driver",".\\MicrosoftWebDriver.exe");

//create Edge instance

driver = new EdgeDriver();

}

else{

//If no browser passed throw exception

throw new Exception("Browser is not correct");

}

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

}

@Test

public void testParameterWithXML() throws InterruptedException{

driver.get("http://demo.guru99.com/V4/");

//Find user name

WebElement userName = driver.findElement(By.name("uid"));

//Fill user name

userName.sendKeys("guru99");

//Find password

WebElement password = driver.findElement(By.name("password"));

//Fill password

password.sendKeys("guru99");

}

}

testing.xml

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE suite SYSTEM "http://testng.org/testng-1.0.dtd">

<suite name="TestSuite" thread-count="2" parallel="tests" >

<test name="ChromeTest">

<parameter name="browser" value="Chrome" />

<classes>

<class name="parallelTest.CrossBrowserScript">

</class>

</classes>

</test>

<test name="FirefoxTest">

<parameter name="browser" value="Firefox" />

<classes>

<class name="parallelTest.CrossBrowserScript">

</class>

</classes>

</test>

<test name="EdgeTest">

<parameter name="browser" value="Edge" />

<classes>

<class name="parallelTest.CrossBrowserScript">

</class>

</classes>

</test>

</suite>