2) Open http://www.wikipedia.org and write JUnit/WebDriver test for the following using Google Chrome: (12 points) Display on console the number of articles in English and click on “English” link (3 points) Search for Anna University. (3 points) Print the 'Motto in English' part in the console and ensure it has the word 'Knowledge' in it. (3 points) Ensure that “Shiv Nadar” is listed in notable people section (3 points)

package selenium;

import java.awt.Robot;

import java.awt.event.KeyEvent;

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.chrome.ChromeOptions;

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

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

public class Dummy {

public static WebDriver driver;

public static void main(String args[])

{

try {

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

ChromeOptions options = new ChromeOptions();

options.addArguments("start-maximized");

options.setExperimentalOption("useAutomationExtension", false);

options.setExperimentalOption("excludeSwitches", new String[]{"enable-automation"});

driver= new ChromeDriver(options);

driver.get("https://wikipedia.org");

Robot robot = new Robot();

Thread.sleep(3000);

driver.findElement(By.xpath("//strong[.='English']")).click();

driver.findElement(By.name("search")).sendKeys("Anna University");

Thread.sleep(1000);

robot.keyPress(KeyEvent.VK\_ENTER);

robot.keyRelease(KeyEvent.VK\_ENTER);

String motto = driver.findElement(By.xpath("//caption/following::td[2]")).getText();

System.out.println(motto);

// Thread.sleep(2000);

// Robot robot = new Robot();

// robot.keyPress(KeyEvent.VK\_TAB);

// robot.keyRelease(KeyEvent.VK\_TAB);

// Thread.sleep(1000);

// robot.keyPress(KeyEvent.VK\_TAB);

// robot.keyRelease(KeyEvent.VK\_TAB);

// Thread.sleep(1000);

// robot.keyPress(KeyEvent.VK\_TAB);

// robot.keyRelease(KeyEvent.VK\_TAB);

// Thread.sleep(1000);

// robot.keyPress(KeyEvent.VK\_SPACE);

// robot.keyRelease(KeyEvent.VK\_SPACE);

// Thread.sleep(3000);

// Process p = Runtime.getRuntime().exec("Upload\_file \_demoaut\_site.exe");

// Thread.sleep(5000);

//

// p.destroy();

Thread.sleep(1000);

driver.quit();

}

catch(Exception e)

{

e.printStackTrace();

}

}

}

3) Open http://maps.google.com and write JUnit / WebDriver test for the following using Google Chrome: (10 points) Search for your home address (2 points) Print the address text that is displayed in the left frame (2 points) Take screenshot of the page at this instant and save the image (2 points) Click on direction and get direction from your office address to your home (2 points) Display the first option distance and time suggested in eclipse console (2 points)

**package** selenium;

**import** java.awt.Robot;

**import** java.awt.event.KeyEvent;

**import** java.io.File;

**import** java.text.DateFormat;

**import** java.text.SimpleDateFormat;

**import** java.util.Calendar;

**import** org.apache.commons.io.FileUtils;

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

**import** org.openqa.selenium.OutputType;

**import** org.openqa.selenium.TakesScreenshot;

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

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

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

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

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

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

**public** **class** Dummy {

**public** **static** WebDriver *driver*;

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

{

**try** {

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

ChromeOptions options = **new** ChromeOptions();

options.addArguments("start-maximized");

options.setExperimentalOption("useAutomationExtension", **false**);

options.setExperimentalOption("excludeSwitches", **new** String[]{"enable-automation"});

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

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

Robot robot = **new** Robot();

Thread.*sleep*(3000);

*driver*.findElement(By.*id*("searchboxinput")).sendKeys("alwarthirunagar annexe");

Thread.*sleep*(3000);

robot.keyPress(KeyEvent.***VK\_ENTER***);

robot.keyRelease(KeyEvent.***VK\_ENTER***);

Thread.*sleep*(8000);

*takeSnapshot*(*driver*,"home");

*driver*.findElement(By.*xpath*("//div[@class='iRxY3GoUYUY\_\_taparea' and @data-value='Directions']")).click();

Thread.*sleep*(4000);

*driver*.findElement(By.*xpath*("//input[@placeholder='Choose starting point, or click on the map...']")).sendKeys("chennai one thoraipakkam");

Thread.*sleep*(3000);

robot.keyPress(KeyEvent.***VK\_ENTER***);

robot.keyRelease(KeyEvent.***VK\_ENTER***);

Thread.*sleep*(4000);

String time = *driver*.findElement(By.*xpath*("//img[@id='section-directions-trip-travel-mode-0']/following::span[1]")).getText();

String distance = *driver*.findElement(By.*xpath*("//img[@id='section-directions-trip-travel-mode-0']/following::div[6]")).getText();

System.***out***.println("The first suggested route details as following");

System.***out***.println("Distance is : "+distance);

System.***out***.println("Time to reach is : "+time);

Thread.*sleep*(1000);

*driver*.quit();

}

**catch**(Exception e)

{

e.printStackTrace();

}

}

**public** **static** **void** takeSnapshot(WebDriver driver, String fileSuffix) {

String currdatentimestamp = **null**;

**try** {

DateFormat dateFormat = **new** SimpleDateFormat("HH\_MM\_SS");

Calendar cal = Calendar.*getInstance*();

currdatentimestamp = dateFormat.format(cal.getTime());

TakesScreenshot scrshot = ((TakesScreenshot)driver);

String filepath = ".//snapshots//"+fileSuffix;

File srcFile = scrshot.getScreenshotAs(OutputType.***FILE***);

String filename = filepath+"\_"+currdatentimestamp+".png";

System.***out***.println("Screenshot destination: "+filename);

File destFile = **new** File(filename);

FileUtils.*copyFile*(srcFile, destFile);

System.***out***.println("Screenshot captured");

} **catch** (Exception e) {

e.printStackTrace();

}

}

}

Output:

Starting ChromeDriver 83.0.4103.39 (ccbf011cb2d2b19b506d844400483861342c20cd-refs/branch-heads/4103@{#416}) on port 28644

Only local connections are allowed.

Please see https://chromedriver.chromium.org/security-considerations for suggestions on keeping ChromeDriver safe.

ChromeDriver was started successfully.

[1598533278.081][WARNING]: This version of ChromeDriver has not been tested with Chrome version 84.

Aug 27, 2020 6:31:21 PM org.openqa.selenium.remote.ProtocolHandshake createSession

INFO: Detected dialect: W3C

Screenshot destination: .//snapshots//home\_18\_08\_470.png

Screenshot captured

The first suggested route details as following

Distance is : 17.6 km

Time to reach is : 43 min

package selenium;

import java.awt.Robot;

import java.awt.event.KeyEvent;

import java.io.File;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Set;

import org.apache.commons.io.FileUtils;

import org.openqa.selenium.By;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.TakesScreenshot;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.chrome.ChromeOptions;

import org.openqa.selenium.interactions.Actions;

public class Dummy {

public static WebDriver driver;

public static void main(String args[])

{

try {

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

ChromeOptions options = new ChromeOptions();

options.addArguments("start-maximized");

options.setExperimentalOption("useAutomationExtension", false);

options.setExperimentalOption("excludeSwitches", new String[]{"enable-automation"});

driver = new ChromeDriver(options);

driver.get("https://mu.ac.in/portal");

WebElement Academics = driver.findElement(By.xpath("//li[@id='menu-item-2100']/following::a[.='ACADEMICS']"));

WebElement Faculty = driver.findElement(By.xpath("//li[@id='menu-item-2100']/following::a[.='ACADEMICS']/following::a[.='FACULTY']"));

WebElement Science = driver.findElement(By.xpath("//li[@id='menu-item-2100']/following::a[.='ACADEMICS']/following::a[.='FACULTY']/following::a[.='SCIENCE & TECHNOLOGY']"));

Actions actions = new Actions(driver);

actions.moveToElement(Academics);

Thread.sleep(2000);

actions.moveToElement(Faculty).perform();

Thread.sleep(5000);

actions.moveToElement(Science).click().perform();

Thread.sleep(5000);

driver.findElement(By.xpath("//a[.='DEPARTMENT OF INFORMATION TECHNOLOGY']")).click();

Thread.sleep(2000);;

String ParentWindow = driver.getWindowHandle();

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

for(String currwin : allwindow) {

driver.switchTo().window(currwin);

}

System.out.println("Parent window : "+ParentWindow);

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

takeSnapshot(driver,"home");

Thread.sleep(1000);

driver.quit();

}

catch(Exception e)

{

e.printStackTrace();

}

}

public static void takeSnapshot(WebDriver driver, String fileSuffix) {

String currdatentimestamp = null;

try {

DateFormat dateFormat = new SimpleDateFormat("HH\_MM\_SS");

Calendar cal = Calendar.getInstance();

currdatentimestamp = dateFormat.format(cal.getTime());

TakesScreenshot scrshot = ((TakesScreenshot)driver);

String filepath = ".//snapshots//"+fileSuffix;

File srcFile = scrshot.getScreenshotAs(OutputType.FILE);

String filename = filepath+"\_"+currdatentimestamp+".png";

System.out.println("Screenshot destination: "+filename);

File destFile = new File(filename);

FileUtils.copyFile(srcFile, destFile);

System.out.println("Screenshot captured");

} catch (Exception e) {

e.printStackTrace();

}

}

}

Sep2019 sep 3:

Testing:

Apptest:

**package** com.sep.tesing.automationtesting;

**import** java.awt.Robot;

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

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

**import** org.openqa.selenium.Point;

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

**import** org.openqa.selenium.interactions.Action;

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

**import** org.testng.Assert;

**import** org.testng.annotations.AfterClass;

**import** org.testng.annotations.AfterMethod;

**import** org.testng.annotations.AfterSuite;

**import** org.testng.annotations.AfterTest;

**import** org.testng.annotations.BeforeClass;

**import** org.testng.annotations.BeforeMethod;

**import** org.testng.annotations.BeforeSuite;

**import** org.testng.annotations.BeforeTest;

**import** org.testng.annotations.Test;

**import** com.codoid.products.fillo.Connection;

**import** com.codoid.products.fillo.Fillo;

**import** com.codoid.products.fillo.Recordset;

**public** **class** AppTest **extends** BaseClass

{

@Test

**public** **static** **void** test1()

{

**try** {

*driver*.findElement(By.*xpath*("//a[.='MOVIES']")).click();

*driver*.manage().timeouts().implicitlyWait(5, TimeUnit.***SECONDS***);

*takeSnapshot*(*driver*,"MOVIES");

String headlines = *driver*.findElement(By.*xpath*("//a[.='Headlines']")).getAttribute("href");

System.***out***.println("The href of headlines is : "+headlines );

String images = *driver*.findElement(By.*xpath*("//a[.='Images']")).getAttribute("href");

System.***out***.println("The href of images is : "+images );

String interviews = *driver*.findElement(By.*xpath*("//a[.='Interviews']")).getAttribute("href");

System.***out***.println("The href of images is : "+interviews );

String reviews = *driver*.findElement(By.*xpath*("//a[.='Reviews']")).getAttribute("href");

System.***out***.println("The href of reviews is : "+reviews );

String webseries = *driver*.findElement(By.*xpath*("//a[.='Web Series']")).getAttribute("href");

System.***out***.println("The href of webseries is : "+webseries );

String television = *driver*.findElement(By.*xpath*("//a[.='Television']")).getAttribute("href");

System.***out***.println("The href of television is : "+television );

String videos = *driver*.findElement(By.*xpath*("//a[.='Videos']")).getAttribute("href");

System.***out***.println("The href of videos is : "+videos );

String southcinema = *driver*.findElement(By.*xpath*("//a[.='South Cinema']")).getAttribute("href");

System.***out***.println("The href of southcinema is : "+southcinema );

WebElement holly = *driver*.findElement(By.*xpath*("//div[@class='subnavbar movies']"));

String hollywood = holly.getAttribute("href");

System.***out***.println("The href of hollywood is : "+hollywood );

String starsspotted = *driver*.findElement(By.*xpath*("//a[.='Stars Spotted']")).getAttribute("href");

System.***out***.println("The href of starsspotted is : "+starsspotted );

String expText="Cricket headlines";

WebElement act = *driver*.findElement(By.*xpath*("//a[.='CRICKET']"));

String actText = act.getAttribute("title");

Assert.*assertEquals*(actText, expText);

System.***out***.println("Actual Text is equal to Expected Text!!!");

Thread.*sleep*(10000);

WebElement tooltip = *driver*.findElement(By.*xpath*("//a[.='CRICKET']"));

Robot robot;

robot = **new** Robot();

Point point;

point = tooltip.getLocation();

**int** x = point.getX();

**int** y = point.getY();

System.***out***.println(x+","+y);

robot.mouseMove(561,285);

Actions builder = **new** Actions(*driver*);

Action acti = builder.moveToElement(tooltip).build();

acti.perform();

Thread.*sleep*(5000);

*CaptureScreen*("Cricket Headlines tool tip");

acti = builder.release(tooltip).build();

acti.perform();

Thread.*sleep*(4000);

*driver*.manage().timeouts().implicitlyWait(5, TimeUnit.***SECONDS***);

Thread.*sleep*(3000);

String bgr = holly.getCssValue("background");

System.***out***.println("The Background color is : "+bgr);

String bgrsub = bgr.substring(0, 16);

String expbgr = "rgb(199, 49, 89)";

Assert.*assertEquals*(bgrsub, expbgr);

{

System.***out***.println("The color is 'Old Rose'");

}

String bgrclr = holly.getCssValue("background-color");

System.***out***.println("The Background color is : "+bgrclr);

String color = holly.getCssValue("color");

System.***out***.println("The color is "+color);

} **catch** (Exception e) {

e.printStackTrace();

}

}

@Test

**public** **void** test2()

{

**try** {

*driver*.findElement(By.*xpath*("//a[.='MOVIES']")).click();

*driver*.manage().timeouts().implicitlyWait(5, TimeUnit.***SECONDS***);

WebElement head = *driver*.findElement(By.*xpath*("//a[.='Headlines']"));

String actheadlines = head.getAttribute("href");

WebElement img = *driver*.findElement(By.*xpath*("//a[.='Images']"));

String actimages = img.getAttribute("href");

WebElement inter = *driver*.findElement(By.*xpath*("//a[.='Interviews']"));

String actinterviews = inter.getAttribute("href")+".html";

WebElement rev = *driver*.findElement(By.*xpath*("//a[.='Reviews']"));

String actreviews = rev.getAttribute("href")+".html";

Fillo fillo = **new** Fillo();

Connection connection = fillo.getConnection("src/test/resources/TestData/testdata.xlsx");

String strquery = "Select \* from data";

Recordset recordset = connection.executeQuery(strquery);

**int** count = recordset.getCount();

System.***out***.println("Count ="+count);

**for** (**int** i=0;i<count;i++)

{

recordset.moveNext();

String Menu = recordset.getField("Menu");

String SubMenu = recordset.getField("SubMenu");

String Href = recordset.getField("Href");

System.***out***.println("Iteration"+i);

System.***out***.println("Menu : "+Menu+"\n"+"SubMenu : "+SubMenu);

**if**(Menu.equalsIgnoreCase("Movies") && SubMenu.equalsIgnoreCase("Headline"))

{

System.***out***.println("Inside condition 1");

System.***out***.println("Actual : "+actheadlines+"\n"+"Expected :"+Href);

Assert.*assertEquals*(actheadlines, Href);

System.***out***.println("The href of headlines submenu is same as in the excel");

head.click();

Thread.*sleep*(2000);

*takeSnapshot*(*driver*,"Question2\_Headline");

*driver*.navigate().back();

Thread.*sleep*(10000);

}

**else** **if**(Menu.equalsIgnoreCase("Movies") && SubMenu.equalsIgnoreCase("Images"))

{

System.***out***.println("Inside condition 2");

System.***out***.println("Actual : "+actimages+"\n"+"Expected :"+Href);

Assert.*assertEquals*(actimages, Href);

System.***out***.println("The href of images submenu is same as in the excel");

*driver*.findElement(By.*xpath*("//a[.='Images']")).click();

Thread.*sleep*(2000);

*takeSnapshot*(*driver*,"Question2\_Images");

*driver*.navigate().back();

}

**else** **if**(Menu.equalsIgnoreCase("Movies") && SubMenu.equalsIgnoreCase("Interviews"))

{

System.***out***.println("Inside condition 3");

System.***out***.println("Actual : "+actinterviews+"\n"+"Expected :"+Href);

Assert.*assertEquals*(actinterviews, Href);

System.***out***.println("The href of interviews submenu is same as in the excel");

*driver*.findElement(By.*xpath*("//a[.='Interviews']")).click();

Thread.*sleep*(2000);

*takeSnapshot*(*driver*,"Question2\_Interviews");

*driver*.navigate().back();

}

**else** **if**(Menu.equalsIgnoreCase("Movies") && SubMenu.equalsIgnoreCase("Reviews"))

{

System.***out***.println("Inside condition 4");

System.***out***.println("Actual : "+actreviews+"\n"+"Expected :"+Href);

Assert.*assertEquals*(actreviews, Href);

System.***out***.println("The href of reviews submenu is same as in the excel");

*driver*.findElement(By.*xpath*("//a[.='Reviews']")).click();

Thread.*sleep*(2000);

*takeSnapshot*(*driver*,"Question2\_Reviews");

}

}

} **catch** (Exception e) {

e.printStackTrace();

}

}

@BeforeSuite

**public** **void** befSuite()

{

**try** {

*LaunchBrowser*("Chrome");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@BeforeTest

**public** **void** befTest()

{

**try** {

System.***out***.println("Before Test");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@BeforeClass

**public** **void** befClass()

{

**try** {

System.***out***.println("Before class");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@BeforeMethod

**public** **void** befMethod()

{

**try** {

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

System.***out***.println("Before method");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@AfterMethod

**public** **void** aftMethod()

{

**try** {

//driver.close();

System.***out***.println("After method");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@AfterClass

**public** **void** aftClass()

{

**try** {

System.***out***.println("After Class");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@AfterTest

**public** **void** aftTest()

{

**try** {

System.***out***.println("After Test");

} **catch** (Exception e) {

e.printStackTrace();

}

}

@AfterSuite

**public** **void** aftSuite()

{

**try** {

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

*driver*.quit();

} **catch** (Exception e) {

e.printStackTrace();

}

}

}

BaseClass:

**package** com.sep.tesing.automationtesting;

**import** java.awt.Rectangle;

**import** java.awt.Robot;

**import** java.awt.Toolkit;

**import** java.awt.image.BufferedImage;

**import** java.io.File;

**import** javax.imageio.ImageIO;

**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;

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

**import** org.openqa.selenium.firefox.FirefoxDriver;

**import** org.openqa.selenium.firefox.FirefoxOptions;

**public** **class** BaseClass {

**public** **static** WebDriver *driver*;

**public** **static** **void** LaunchBrowser(String BrowserType)

{

System.***out***.println("In Launch Browser method!!");

**if**(BrowserType.equalsIgnoreCase("Chrome"))

{

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

ChromeOptions options = **new** ChromeOptions();

options.addArguments("--start-maximized");

options.addArguments("--disable-extensions");

options.setExperimentalOption("useAutomationExtension", **false**);

options.setExperimentalOption("excludeSwitches", **new** String[] {"enable-automation"});

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

}

**else** **if**(BrowserType.equalsIgnoreCase("Firefox"))

{

System.*setProperty*("webdriver.gecko.driver", "geckodriver.exe");

FirefoxOptions options = **new** FirefoxOptions();

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

options.addArguments("--disable-extensions");

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

}

}

**public** **static** **void** takeSnapshot(WebDriver driver,String filesuffix)

{

**try** {

TakesScreenshot scrShot =((TakesScreenshot)driver);

String filename = "src//test//resources//snapshots//"+filesuffix+".png";

File SrcFile=scrShot.getScreenshotAs(OutputType.***FILE***);

System.***out***.println("File name is : "+filesuffix);

File destFile = **new** File(filename);

System.***out***.println("Screenshot captured in path: "+filename);

FileUtils.*copyFile*(SrcFile, destFile);

System.***out***.println("Screenshot Captured ");

} **catch** (Exception e) {

e.printStackTrace();

}

}

**public** **static** String CaptureScreen(String filename)

{

**try**{

Robot robotClassObject = **new** Robot();

Rectangle screenSize = **new** Rectangle(Toolkit.*getDefaultToolkit*().getScreenSize());

BufferedImage tmp = robotClassObject.createScreenCapture(screenSize);

String filepath = "src//test//resources//snapshots//"+filename+".png";

ImageIO.*write*(tmp, "png",**new** File(filepath));

**return** filepath;

}**catch**(Exception e)

{

System.***out***.println("Some exception occured." + e.getMessage());

**return** "";

}

}

}

TestngXML:

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

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

<suite name=*"Suite"*>

<test name=*"Test"*>

<classes>

<class name=*"com.sep.tesing.automationtesting.AppTest"*>

<methods>

<include name=*"test1"* />

<include name=*"test2"* />

</methods>

</class>

</classes>

</test><!-- Test -->

</suite><!-- Suite -->