**Name: Princia Melita Dsouza**

**Emp id: 2483929**

**Github:** **https://github.com/princia03/Phase\_5.git**

**Automate a Web Application**

DESCRIPTION

**Project Objective:**

As a Full Stack Developer, you have to build an automation script that automated the basic functionalities like registration and login.

**Background of the problem statement:**

As the project is in the final stage, management has asked you to automate the basic functionalities like registration and login for all the internal employees. This will help the development and DevOps team to work efficiently with the application.

**You must use the following:**

● Eclipse  
● Selenium WebDriver  
● GitHub

**REDIFFDEMO:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class RediffDemo {

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

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

driver.get("http://register.rediff.com/register/register.php?FormName=user\_details");

// driver.findElements(By.xpath("//input[@type='text']")).get(0).sendKeys("hari");

driver.findElement(By.xpath("(//input[@type='text'])[1]")).sendKeys("hari gadhe");

Thread.sleep(2000);

driver.findElement(By.xpath("(//input[@type='text'])[2]")).sendKeys("admin123");

Thread.sleep(2000);

driver.findElement(By.xpath("(//input[@type='button'])[1]")).click();

Thread.sleep(2000);

driver.findElement(By.xpath("(//input[@type='password'])[1]")).sendKeys("password@123");

}

}

**CSSS SELECTOR DEMO:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class CSSSelectorDemo {

public static void main(String[] args) {

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

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

// 1. find element using tag and id ==> tagname#idvalue

driver.findElement(By.cssSelector("input#first\_name")).sendKeys("hari");

//driver.findElement(By.cssSelector("input.required")).sendKeys("Gadhe");

driver.findElement(By.cssSelector("input[name=last\_name]")).sendKeys("Gadhe");

}

}

**WEB ELEMENT DEMO:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class WebelementDemo {

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

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

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

// store the location of the element in an object of type WebElement

WebElement e1 = driver.findElement(By.id("searchInput"));

e1.isDisplayed();

e1.isEnabled();

e1.sendKeys("Automation testing");

Thread.sleep(3000);

// Name locator

WebElement e2 = driver.findElement(By.name("search")) ;

e2.clear();

e2.sendKeys("New data for automation");

}

}

**XPATH DEMO:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class XPATHDemo {

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

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

// Find an element using XPATH locator

// XPATh : Relative XPATH : //tag[@attribute='value']

driver.findElement(By.xpath("//input[@name='search']")).sendKeys("findelement");

// element 2 to click on button

Thread.sleep(2000);

driver.findElement(By.xpath("//button[@type='submit']")).click();

}

}

**LINKS DEMO:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

public class LinksDemo {

public static void main(String[] args) {

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

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

driver.manage().deleteAllCookies();

driver.findElement(By.xpath("//\*[@id='searchInput']")).sendKeys("Testing");

driver.findElement(By.cssSelector("button[type=submit]")).click();

// click on the link

WebElement li= driver.findElement(By.linkText("Current events"));

li.isDisplayed();

li.isEnabled();

li.click();

driver.findElement(By.partialLinkText("Log")).click();

driver.close();

}

}

**LOCATORS ID:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class LocatorsID {

public static void main(String[] args) {

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

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

// Check if the element is displayed

boolean dis = driver.findElement(By.id("searchInput")).isDisplayed();

System.out.println("IS the element displayed ?" + dis);

// check if the element is enabled or not

boolean enb = driver.findElement(By.id("searchInput")).isEnabled();

System.out.println("IS the element enabled ?" + enb);

// Enter data in the webelement - input box

if(enb==true)

{

driver.findElement(By.id("searchInput")).sendKeys("Automation testing");

}

else

{

System.out.println("textbox is not enabled");

}

}

}

**LOCATOR TAGS:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class Locatortag {

public static void main(String[] args) {

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

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

// wherever out attribute value is not unique, then go for findElements & get

driver.findElements(By.tagName("input")).get(2).sendKeys("data");

}

}

**NAVIGATION METHOD:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

public class NavigationMethods {

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

// TODO Auto-generated method stub

WebDriver driver = new ChromeDriver();

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

driver.manage().deleteAllCookies();

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

String expctedtitle= "Wikipedia123";

String actualtitle = driver.getTitle(); // will fetch the title of the page

if(expctedtitle.equals(actualtitle))

{

System.out.println("title of the page is correct");

}

else {

System.out.println("title of the page is not correct");

}

driver.navigate().to("https://www.selenium.dev/downloads/");

String title1 = driver.getTitle(); // will fetch the title of the page

System.out.println("Title of Page2 =" + title1);

driver.navigate().back(); // navigates back to previous url

Thread.sleep(2000);

driver.navigate().forward();

Thread.sleep(2000);

driver.close();

}

}

**SETUP CHECK:**

package com.qa.SeleniumScripts;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

public class SetUpcheck {

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

{

// WebDriver

// can open a chrome browser window

WebDriver driver = new ChromeDriver();

// Maxamize the browser window

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

// Open a webpage-URL on the browser

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

// do some testing

//Close the browser window

Thread.sleep(2000); // add wait time before closing the window

driver.close(); // will close that particular browser window

}

}