POM:

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

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>Cucumber</groupId>

<artifactId>Cigna</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>Cigna</name>

<!-- FIXME change it to the project's website -->

<url>http://www.example.com</url>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.source>1.7</maven.compiler.source>

<maven.compiler.target>1.7</maven.compiler.target>

</properties>

<dependencies>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-junit -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-junit</artifactId>

<version>7.17.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.logging.log4j/log4j-core -->

<dependency>

<groupId>org.apache.logging.log4j</groupId>

<artifactId>log4j-core</artifactId>

<version>2.23.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.testng/testng -->

<dependency>

<groupId>org.testng</groupId>

<artifactId>testng</artifactId>

<version>7.10.1</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/net.masterthought/cucumber-reporting -->

<dependency>

<groupId>net.masterthought</groupId>

<artifactId>cucumber-reporting</artifactId>

<version>5.8.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-java -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-java</artifactId>

<version>7.17.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-junit -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-junit</artifactId>

<version>7.17.0</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/junit/junit -->

<dependency>

<groupId>junit</groupId>

<artifactId>junit</artifactId>

<version>4.13.2</version>

<scope>test</scope>

</dependency>

<!-- https://mvnrepository.com/artifact/io.cucumber/cucumber-testng -->

<dependency>

<groupId>io.cucumber</groupId>

<artifactId>cucumber-testng</artifactId>

<version>7.17.0</version>

</dependency>

<!-- https://mvnrepository.com/artifact/info.cukes/cucumber-testng -->

<dependency>

<groupId>info.cukes</groupId>

<artifactId>cucumber-testng</artifactId>

<version>1.2.5</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>5.2.5</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.apache.poi/poi-ooxml -->

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>5.2.5</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml-schemas</artifactId>

<version>4.1.2</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.seleniumhq.selenium/selenium-java -->

<dependency>

<groupId>org.seleniumhq.selenium</groupId>

<artifactId>selenium-java</artifactId>

<version>4.19.1</version>

</dependency>

<!-- https://mvnrepository.com/artifact/org.testng/testng -->

</dependencies>

<build>

<pluginManagement><!-- lock down plugins versions to avoid using Maven defaults (may be moved to parent pom) -->

<plugins>

<!-- clean lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#clean\_Lifecycle -->

<plugin>

<artifactId>maven-clean-plugin</artifactId>

<version>3.1.0</version>

</plugin>

<!-- default lifecycle, jar packaging: see https://maven.apache.org/ref/current/maven-core/default-bindings.html#Plugin\_bindings\_for\_jar\_packaging -->

<plugin>

<artifactId>maven-resources-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-compiler-plugin</artifactId>

<version>3.8.0</version>

</plugin>

<plugin>

<artifactId>maven-surefire-plugin</artifactId>

<version>2.22.1</version>

</plugin>

<plugin>

<artifactId>maven-jar-plugin</artifactId>

<version>3.0.2</version>

</plugin>

<plugin>

<artifactId>maven-install-plugin</artifactId>

<version>2.5.2</version>

</plugin>

<plugin>

<artifactId>maven-deploy-plugin</artifactId>

<version>2.8.2</version>

</plugin>

<!-- site lifecycle, see https://maven.apache.org/ref/current/maven-core/lifecycles.html#site\_Lifecycle -->

<plugin>

<artifactId>maven-site-plugin</artifactId>

<version>3.7.1</version>

</plugin>

<plugin>

<artifactId>maven-project-info-reports-plugin</artifactId>

<version>3.0.0</version>

</plugin>

</plugins>

</pluginManagement>

</build>

</project>

--CignaFieldElements--

package CignaWebsite;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

public class CignaFieldElements {

public WebDriver driver;

public CignaFieldElements(WebDriver driver) {

this.driver = driver;

PageFactory.initElements(driver, this);

}

@FindBy(id = "username")

WebElement username;

@FindBy(id = "password")

WebElement password;

@FindBy(id = "loginbutton")

WebElement Loginbutton;

@FindBy(xpath = "//\*[text()='Disclaimer']")

WebElement Disclaimersec;

public WebElement Username() {

return username;

}

public WebElement Password() {

return password;

}

public WebElement loginbutton() {

return Loginbutton;

}

public WebElement Disclaimer() {

return Disclaimersec;

}

}

-- CignaStepDef--

package CignaWebsite;

import static org.testng.Assert.assertTrue;

import java.io.IOException;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.chrome.ChromeDriver;

import org.testng.Assert;

import io.cucumber.java.en.And;

import io.cucumber.java.en.Given;

import io.cucumber.java.en.Then;

import io.cucumber.java.en.When;

public class CignaStepDef {

WebDriver driver;

@Given("User invokes chrome")

public void User\_invokes\_chrome() {

driver = new ChromeDriver();

return;

}

@And("Invokes Cigna customer portal")

public void invokes\_cigna\_customer\_portal() {

ReuseObaj cigna = new ReuseObaj();

cigna.CignaPage(driver);

}

@And("Verify URL")

public void verify\_url() {

String url = driver.getCurrentUrl();

Assert.assertEquals(url, "https://www.cigna.com/");

System.out.println(url);

}

@And("Verify Title")

public void Verify\_Title( ) {

String title = driver.getTitle();

Assert.assertEquals(title, "Cigna Healthcare | Health Insurance, Dental Plans & Medicare");

System.out.println(title);

}

@And("Take a snapshot of the Desclaimer")

public void take\_a\_snapshot\_of\_the\_desclaimer() throws IOException, InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.DisCl(driver);

ReuseObaj sn = new ReuseObaj();

sn.snapdis(driver);

Thread.sleep(3000);

}

@And("Verify Healthplan page")

public void verify\_healthplan\_page() throws InterruptedException {

WebElement Healthplan = driver.findElement(By.linkText("Check out supplemental health insurance plans"));

JavascriptExecutor js = (JavascriptExecutor) driver;

js.executeScript("arguments[0].scrollIntoView()",Healthplan);

js.executeScript("window.scrollBy(0,-200)");

Thread.sleep(3000);

driver.findElement(By.linkText("Check out supplemental health insurance plans")).click();

Thread.sleep(5000);

driver.navigate().back();

Thread.sleep(3000);

js.executeScript("window.scrollBy(0,-2000)");

Thread.sleep(3000);

}

@And("Verify Heading")

public void verify\_heading() throws InterruptedException {

WebElement heading = driver.findElement(By.xpath("//h1[text()='Welcome to Cigna Healthcare']"));

Boolean verifyHeading = heading.getText().equalsIgnoreCase("Welcome to Cigna Healthcare");

assertTrue(verifyHeading);

System.out.println(heading);

Thread.sleep(3000);

}

@And("Check Medical Coverage")

public void check\_medical\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Medical(driver);

}

@And("Check Medicare Coverage")

public void check\_medicare\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Medicare(driver);

}

@And("Check Dental Coverage")

public void check\_dental\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Dental(driver);

}

@And("Check SupplimentalHealth Coverage")

public void check\_supplimental\_health\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.SupplimentalHealth(driver);

}

@And("Check Planstroughemployer Coverage")

public void check\_planstroughemployer\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Planstroughemployer(driver);

}

@And("Navigate to Login page")

public void navigate\_to\_login\_page() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.login(driver);

Thread.sleep(3000);

}

@When("User Submit Details")

public void user\_submit\_details(io.cucumber.datatable.DataTable dataTable) {

CignaFieldElements cf = new CignaFieldElements(driver);

cf.Username().sendKeys("sree");

cf.Password().sendKeys("password");

cf.loginbutton().click();

}

@Then("User validate the Error message")

public void user\_validate\_the\_error\_message() {

System.out.println("Scripting in progress");

}

}

--ReuseObaj --

package CignaWebsite;

import java.io.File;

import java.io.FileInputStream;

import java.io.IOException;

import java.text.SimpleDateFormat;

import java.time.Duration;

import java.util.Date;

import java.util.Iterator;

import java.util.List;

import java.util.Set;

import java.util.concurrent.TimeUnit;

import org.apache.commons.io.FileUtils;

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

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

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

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

import org.openqa.selenium.OutputType;

import org.openqa.selenium.TakesScreenshot;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.interactions.Actions;

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

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

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

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

import org.testng.annotations.DataProvider;

public class ReuseObaj {

public void snapdis(WebDriver driver) throws IOException

{

TakesScreenshot screenshot = (TakesScreenshot)driver;

File source = screenshot.getScreenshotAs(OutputType.FILE);

FileUtils.copyFile(source, new File("C:\\Users\\srina\\OneDrive\\Pictures\\Screenshots\\"+timestamp()+".png"));

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

}

public void login(WebDriver driver) throws InterruptedException

{

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

JavascriptExecutor js = (JavascriptExecutor) driver;

Wait<WebDriver> wait = new WebDriverWait(driver, Duration.ofSeconds(10));

WebElement loginbutton =(WebElement) js.executeScript("return document.querySelector('chc-header>li>leaf-button')");

wait.until(ExpectedConditions.visibilityOf(loginbutton));

Actions actions = new Actions(driver);

actions.moveToElement(loginbutton).click().build().perform();

Thread.sleep(5000);

}

public void Medical(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement a = (WebElement)js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"#coveragetype-leaf\")");

Select coverage = new Select(a);

coverage.selectByIndex(1);

Thread.sleep(3000);

WebElement exp =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"div > form > leaf-button\").shadowRoot.querySelector(\"button\")");

Actions actions = new Actions(driver);

actions.moveToElement(exp).click().build().perform();

Thread.sleep(3000);

}

public void Medicare(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement a = (WebElement)js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"#coveragetype-leaf\")");

Select coverage = new Select(a);

coverage.selectByIndex(2);

Thread.sleep(3000);

WebElement exp =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"div > form > leaf-button\").shadowRoot.querySelector(\"button\")");

Actions actions = new Actions(driver);

actions.moveToElement(exp).click().build().perform();

Thread.sleep(3000);

}

public void Dental(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement a = (WebElement)js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"#coveragetype-leaf\")");

Select coverage = new Select(a);

coverage.selectByIndex(3);

Thread.sleep(3000);

WebElement exp =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"div > form > leaf-button\").shadowRoot.querySelector(\"button\")");

Actions actions = new Actions(driver);

actions.moveToElement(exp).click().build().perform();

Thread.sleep(3000);

}

public void SupplimentalHealth(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement a = (WebElement)js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"#coveragetype-leaf\")");

Select coverage = new Select(a);

coverage.selectByIndex(4);

Thread.sleep(3000);

WebElement exp =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"div > form > leaf-button\").shadowRoot.querySelector(\"button\")");

Actions actions = new Actions(driver);

actions.moveToElement(exp).click().build().perform();

Thread.sleep(3000);

}

public void Planstroughemployer(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement a = (WebElement)js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"#coveragetype-leaf\")");

Select coverage = new Select(a);

coverage.selectByIndex(5);

Thread.sleep(3000);

WebElement exp =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"div > form > leaf-button\").shadowRoot.querySelector(\"button\")");

Actions actions = new Actions(driver);

actions.moveToElement(exp).click().build().perform();

Thread.sleep(3000);

}

public void medical(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement a = (WebElement)js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"#coveragetype-leaf\")");

Select coverage = new Select(a);

coverage.selectByIndex(1);

Thread.sleep(3000);

WebElement exp =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > chcapp-shop-plans-banner\").shadowRoot.querySelector(\"div > form > leaf-button\").shadowRoot.querySelector(\"button\")");

Actions actions = new Actions(driver);

actions.moveToElement(exp).click().build().perform();

Thread.sleep(3000);

driver.navigate().back();

Thread.sleep(3000);

}

public void CignaPage(WebDriver driver)

{

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

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

}

public void DisCl(WebDriver driver) throws InterruptedException

{

WebElement Disclaimer = driver.findElement(By.xpath("//\*[text()='Disclaimer']"));

JavascriptExecutor js = (JavascriptExecutor) driver;

js.executeScript("arguments[0].scrollIntoView();", Disclaimer);

Thread.sleep(3000);

}

public void ParentWin(WebDriver driver) throws InterruptedException

{

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

Iterator<String> iterator = windows.iterator();

String ParentWin = iterator.next();

String ChildWin = iterator.next();

driver.switchTo().window(ParentWin);

}

public void ChildWin(WebDriver driver) throws InterruptedException

{

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

Iterator<String> iterator = windows.iterator();

String ParentWin = iterator.next();

String ChildWin = iterator.next();

driver.switchTo().window(ChildWin);

}

public void Errormessage(WebDriver driver) throws InterruptedException

{

CignaFieldElements cf = new CignaFieldElements(driver);

Wait<WebDriver> wait = new WebDriverWait(driver, Duration.ofSeconds(30));

wait.until(ExpectedConditions.visibilityOf(cf.Username()));

wait.until(ExpectedConditions.visibilityOf(driver.findElement(By.xpath("//div[contains(text(),'entered does not match our records')]"))));

String message = driver.findElement(By.xpath("//div[contains(text(),'entered does not match our records')]")).getText().trim();

//System.out.println(message);

if(message.equals("The username and password combination you entered does not match our records."))

{

cf.Username().clear();

cf.Password().clear();

Thread.sleep(5000);

wait.until(ExpectedConditions.visibilityOf(cf.Username()));

}

}

public void AnotherCoverage(WebDriver driver) throws InterruptedException

{

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement anothercov =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1709087660143 > div > chc-sticky-banner > div:nth-child(3) > leaf-button-link\")");

Actions actions = new Actions(driver);

actions.moveToElement(anothercov).click().build().perform();

}

public void Snpcoverage(WebDriver driver) throws IOException

{

TakesScreenshot screenshot = (TakesScreenshot)driver;

File source = screenshot.getScreenshotAs(OutputType.FILE);

FileUtils.copyFile(source, new File("C:\\Users\\srina\\OneDrive\\Pictures\\Screenshots\\"+timestamp()+".png"));

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

}

public void Alllinks(WebDriver driver)

{

List<WebElement> allLinks = driver.findElements(By.tagName("a"));

System.out.println("Total number of links: "+ allLinks.size());

for(int i=0;i<allLinks.size();i++)

{

System.out.println("Links are: " + allLinks.get(i).getAttribute("href"));

}

}

public static String timestamp()

{

return new SimpleDateFormat("YYYY-MM-DD HH-MM-SS").format(new Date());

}

@DataProvider

public String[][] UserDetails() throws IOException {

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

File s = new File("C:\\Users\\srina\\OneDrive\\Documents\\Test\\Book2.xlsx");

FileInputStream stream = new FileInputStream(s);

XSSFWorkbook work\_book = new XSSFWorkbook(stream);

XSSFSheet sheet = work\_book.getSheet("Sheet1");

int NoRow = sheet.getPhysicalNumberOfRows();

int NoCol = sheet.getRow(0).getLastCellNum();

// Iterate through the rows and columns to read the data

String[][] data = new String[NoRow - 1][NoCol];

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

for (int j = 0; j < NoCol; j++) {

DataFormatter df = new DataFormatter();

data[i][j] = df.formatCellValue(sheet.getRow(i + 1).getCell(j));

}

}

work\_book.close();

stream.close();

return data;

}

}

-- CignaporSteDef—

package cucumberCigna;

import static org.testng.Assert.assertTrue;

import java.io.IOException;

import java.time.Duration;

import java.util.List;

import java.util.Map;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

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.support.ui.ExpectedConditions;

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

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

import org.testng.Assert;

import CignaWebsite.CignaFieldElements;

import CignaWebsite.ReuseObaj;

import io.cucumber.java.en.And;

import io.cucumber.java.en.Given;

import io.cucumber.java.en.Then;

import io.cucumber.java.en.When;

public class CignaporSteDef {

static WebDriver driver;

@Given("User invokes chrome")

public void User\_invokes\_chrome() {

driver = new ChromeDriver();

return;

}

@And("Invokes Cigna customer portal")

public void invokes\_cigna\_customer\_portal() {

ReuseObaj cigna = new ReuseObaj();

cigna.CignaPage(driver);

}

@And("Verify URL")

public void verify\_url() {

String url = driver.getCurrentUrl();

Assert.assertEquals(url, "https://www.cigna.com/");

System.out.println("URL : "+url);

}

@And("Verify Title")

public void Verify\_Title( ) {

String title = driver.getTitle();

Assert.assertEquals(title, "Cigna Healthcare | Health Insurance, Dental Plans & Medicare");

System.out.println("Title : "+title);

}

@And("Take a snapshot of the Desclaimer")

public void take\_a\_snapshot\_of\_the\_desclaimer() throws IOException, InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.DisCl(driver);

ReuseObaj sn = new ReuseObaj();

sn.snapdis(driver);

Thread.sleep(3000);

}

@And("Verify Healthplan page")

public void verify\_healthplan\_page() throws InterruptedException {

WebElement Healthplan = driver.findElement(By.linkText("Check out supplemental health insurance plans"));

JavascriptExecutor js = (JavascriptExecutor) driver;

js.executeScript("arguments[0].scrollIntoView()",Healthplan);

js.executeScript("window.scrollBy(0,-200)");

Thread.sleep(3000);

driver.findElement(By.linkText("Check out supplemental health insurance plans")).click();

Thread.sleep(5000);

driver.navigate().back();

Thread.sleep(3000);

js.executeScript("window.scrollBy(0,-2000)");

Thread.sleep(3000);

}

@And("Verify Heading")

public void verify\_heading() throws InterruptedException {

WebElement heading = driver.findElement(By.xpath("//h1[text()='Welcome to Cigna Healthcare']"));

Boolean verifyHeading = heading.getText().equalsIgnoreCase("Welcome to Cigna Healthcare");

assertTrue(verifyHeading);

System.out.println(heading);

Thread.sleep(3000);

}

@When("User select Medical Coverage")

public void check\_medical\_coverage() throws InterruptedException {

System.out.println("Following are the Coverage headings: ");

ReuseObaj cigna = new ReuseObaj();

cigna.Medical(driver);

}

@Then("Verify Medical Coverage Screen")

public void Verify\_Medical\_Coverage\_Screen() throws InterruptedException {

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement MedicalHeading =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1695257321650 > chc-hero > h1\")");

System.out.println(MedicalHeading.getText());

driver.navigate().back();

}

@When("User select Medicare Coverage")

public void check\_medicare\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Medicare(driver);

}

@Then("Verify Medicare Coverage Screen")

public void Verify\_Medicare\_coverage\_screen() throws InterruptedException {

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement MedicareHeading =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1695257321650 > chc-hero > h1\")");

System.out.println(MedicareHeading.getText());

driver.navigate().back();

}

@When("User select Dental Coverage")

public void check\_dental\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Dental(driver);

}

@Then("Verify Dental Coverage Screen")

public void Verify\_dental\_Coverage\_Screen() throws InterruptedException {

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement DentalHeading =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1695257321650 > chc-hero > h1\")");

System.out.println(DentalHeading.getText());

driver.navigate().back();

}

@When("User select SupplimentalHealth Coverage")

public void check\_supplimental\_health\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.SupplimentalHealth(driver);

}

@Then("Verify SupplimentalHealth Coverage Screen")

public void Verify\_SupplimentalHealth\_coverage\_screen() throws InterruptedException {

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement SupplimentaryHeading =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1695257321587 > chc-hero > h1\")");

System.out.println(SupplimentaryHeading.getText());

driver.navigate().back();

}

@When("User select Planstroughemployer Coverage")

public void check\_planstroughemployer\_coverage() throws InterruptedException {

ReuseObaj cigna = new ReuseObaj();

cigna.Planstroughemployer(driver);

}

@Then("Verify Planstroughemployer Coverage Screen")

public void Verify\_MPlanstroughemployer\_coverage\_screen() throws InterruptedException {

JavascriptExecutor js = (JavascriptExecutor) driver;

WebElement PlanthroughEmpHeading =(WebElement) js.executeScript("return document.querySelector(\"#iw\_comp1695257321650 > chc-hero > h1\")");

System.out.println(PlanthroughEmpHeading.getText());

driver.navigate().back();

Thread.sleep(6000);

}

@Given("User Navigate to Login page")

public void user\_navigate\_to\_login\_page() throws InterruptedException {

ReuseObaj cigna1 = new ReuseObaj();

cigna1.login(driver);

}

@Given("User toggle between pages")

public void user\_toggle\_between\_pages() throws InterruptedException {

ReuseObaj cigna1 = new ReuseObaj();

cigna1.ParentWin(driver);

Thread.sleep(3000);

cigna1.ChildWin(driver);

Thread.sleep(3000);

}

@When("User Submit Details")

public void user\_submit\_details(io.cucumber.datatable.DataTable dataTable) throws InterruptedException {

List<Map<String, String>> userlist = dataTable.asMaps(String.class, String.class);

CignaFieldElements cf = new CignaFieldElements(driver);

Wait<WebDriver> wait = new WebDriverWait(driver, Duration.ofSeconds(30));

wait.until(ExpectedConditions.visibilityOf(cf.Username()));

for (int i=0;i<=2;i++) {

cf.Username().sendKeys(userlist.get(i).get("Username"));

cf.Password().sendKeys(userlist.get(i).get("Password"));

cf.loginbutton().click();

Thread.sleep(5000);

ReuseObaj cigna1 = new ReuseObaj();

cigna1.Errormessage(driver);

}}

@Then("User validate the Error message")

public void user\_validate\_the\_error\_message() {

String message = driver.findElement(By.xpath("//div[contains(text(),'entered does not match our records')]")).getText().trim();

System.out.println(message);

}

@Given("User Navigate to Cigna health care page")

public void User\_Navigate\_to\_Cigna\_health\_care\_page() throws InterruptedException {

ReuseObaj cigna1 = new ReuseObaj();

cigna1.ParentWin(driver);

Thread.sleep(3000);

}

@Then("Verify all the links in the webpage")

public void Verify\_all\_the\_links\_in\_the\_webpage() throws InterruptedException {

ReuseObaj cigna1 = new ReuseObaj();

cigna1.Alllinks(driver);

}

@When("User look for another kind of coverages")

public void User\_look\_for\_another\_kind\_of\_coverages() throws InterruptedException {

ReuseObaj cigna1 = new ReuseObaj();

cigna1.AnotherCoverage(driver);

}

@Then("capture screenshot")

public void capture\_all\_links\_in\_the\_page( ) throws IOException {

ReuseObaj cigna1 = new ReuseObaj();

cigna1.Snpcoverage(driver);

}}

-- CignaRunner—

package cucumberCigna;

import org.junit.runner.RunWith;

import cucumber.api.CucumberOptions;

import io.cucumber.junit.Cucumber;

@RunWith(Cucumber.class)

@CucumberOptions(

features= "src/test/java/cucumberCigna/", tags = {"@tag1","@tag2","@tag3"},

glue = {"CignaWebsite","CucumberCigna"},

plugin= {"pretty","html:target/cucumber-html","json:target/cucumber.json"})

public class CignaRunner {

}

Cigna.feature

@tag

Feature: Title of your feature

I want to use this template for my feature file

@tag1

Scenario: Verify Cigna coverages

Given User invokes chrome

And Invokes Cigna customer portal

And Verify URL

And Verify Title

And Take a snapshot of the Desclaimer

And Verify Healthplan page

And Verify Heading

When User select Medical Coverage

Then Verify Medical Coverage Screen

When User select Medicare Coverage

Then Verify Medicare Coverage Screen

When User select Dental Coverage

Then Verify Dental Coverage Screen

When User select SupplimentalHealth Coverage

Then Verify SupplimentalHealth Coverage Screen

When User select Planstroughemployer Coverage

Then Verify Planstroughemployer Coverage Screen

@tag2

Scenario: Login to Cigna Account

Given User Navigate to Login page

And User toggle between pages

When User Submit Details

| Username | Password |

| sree | Sree123 |

| Senthil | Sen123 |

| Jerin | Jer123 |

Then User validate the Error message

@tag3

Scenario: Validate another Kind of coverages

Given User Navigate to Cigna health care page

Then Verify all the links in the webpage

When User look for another kind of coverages

Then capture screenshot

-- CignaScript--

package CignaWebsite;

import static org.testng.Assert.assertTrue;

import java.io.IOException;

import java.time.Duration;

import org.openqa.selenium.By;

import org.openqa.selenium.JavascriptExecutor;

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.support.ui.ExpectedCondition;

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

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

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

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

import org.testng.Assert;

import org.testng.annotations.BeforeTest;

import org.testng.annotations.Test;

public class CignaScript {

WebDriver driver;

@BeforeTest

public void driversetup()

{

driver = new ChromeDriver();

return;

}

@Test(priority=1)

public void Cignawebpage()

{

ReuseObaj cigna = new ReuseObaj();

cigna.CignaPage(driver);

}

@Test(priority=2)

public void pageverificaiton() throws InterruptedException

{

String url = driver.getCurrentUrl();

String title = driver.getTitle();

System.out.println(title);

Assert.assertEquals(url, "https://www.cigna.com/");

Thread.sleep(3000);

Assert.assertEquals(title, "Cigna Healthcare | Health Insurance, Dental Plans & Medicare");

System.out.println(url);

System.out.println(title);

}

@Test(priority=3)

public void Desclimer() throws IOException, InterruptedException

{

ReuseObaj cigna = new ReuseObaj();

cigna.DisCl(driver);

ReuseObaj sn = new ReuseObaj();

sn.snapdis(driver);

Thread.sleep(3000);

}

@Test(priority=4)

public void Healthplans() throws IOException, InterruptedException

{

WebElement Healthplan = driver.findElement(By.linkText("Check out supplemental health insurance plans"));

JavascriptExecutor js = (JavascriptExecutor) driver;

js.executeScript("arguments[0].scrollIntoView()",Healthplan);

js.executeScript("window.scrollBy(0,-200)");

Thread.sleep(3000);

driver.findElement(By.linkText("Check out supplemental health insurance plans")).click();

Thread.sleep(5000);

driver.navigate().back();

Thread.sleep(3000);

js.executeScript("window.scrollBy(0,-2000)");

Thread.sleep(3000);

WebElement heading = driver.findElement(By.xpath("//h1[text()='Welcome to Cigna Healthcare']"));

Boolean verifyHeading = heading.getText().equalsIgnoreCase("Welcome to Cigna Healthcare");

assertTrue(verifyHeading);

System.out.println(heading);

Thread.sleep(3000);

}

@Test(priority=5)

public void Coverage() throws InterruptedException

{

ReuseObaj cigna = new ReuseObaj();

cigna.Medical(driver);

// cigna.Medicare(driver);

// cigna.Dental(driver);

// cigna.SupplimentalHealth(driver);

//cigna.Planstroughemployer(driver);

}

@Test(priority=6,dataProvider = "UserDetails", dataProviderClass = ReuseObaj.class)

public void logincigna(String UserName, String Password) throws InterruptedException

{

try {

ReuseObaj cigna = new ReuseObaj();

cigna.login(driver);

cigna.ParentWin(driver);

Thread.sleep(3000);

cigna.ChildWin(driver);

CignaFieldElements cf = new CignaFieldElements(driver);

Wait<WebDriver> wait = new WebDriverWait(driver, Duration.ofSeconds(30));

wait.until(ExpectedConditions.visibilityOf(cf.Username()));

cf.Username().sendKeys(UserName);

cf.Password().sendKeys(Password);

cf.loginbutton().click();

Thread.sleep(5000);

wait.until(ExpectedConditions.visibilityOf(driver.findElement(By.xpath("//div[contains(text(),'entered does not match our records')]"))));

String message = driver.findElement(By.xpath("//div[contains(text(),'entered does not match our records')]")).getText().trim();

System.out.println(message);

if(message.equals("The username and password combination you entered does not match our records."))

{

cf.Username().clear();

cf.Password().clear();

Thread.sleep(5000);

wait.until(ExpectedConditions.visibilityOf(cf.Username()));

} }

catch(NullPointerException e)

{

System.out.print("NullPointerException Caught");

}

}}