coverFoxPOM Uisng TestNG

package coverFoxPOM;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

import org.testng.Reporter;

public class CoverFoxHomePage

{

//1. variables-->WebElements

@FindBy(xpath = "//div[text()='Male']") private WebElement gender;

//2. constructor

public CoverFoxHomePage(WebDriver driver)

{

PageFactory.initElements(driver, this);

}

//3. methods

public void clickOnGender()

{

Reporter.log("clicking on gender", true);

gender.click();

}

}

package coverFoxPOM;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

import org.testng.Reporter;

public class CoverFoxHealthPlanPage

{

//variables-->WebElements

@FindBy(className = "next-btn") private WebElement nextButton;

//constructor

public CoverFoxHealthPlanPage(WebDriver driver)

{

PageFactory.initElements(driver, this);

}

//methods

public void clickOnNextButtonHealthPlanPage()

{

Reporter.log("clicking on next button of health plan page", true);

nextButton.click();

}

}

package coverFoxPOM;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

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

import org.testng.Reporter;

public class CoverFoxMemberDetailsPage {

//variables

@FindBy(name = "You") private WebElement ageDropDown;

@FindBy(className = "next-btn") private WebElement nextButton;

//constructor

public CoverFoxMemberDetailsPage(WebDriver driver)

{

PageFactory.initElements(driver, this);

}

//methods

public void handleAgeDropDown(String age)

{

Reporter.log("handeling age drop down", true);

Select selectAge= new Select(ageDropDown);

selectAge.selectByValue(age+"y");

}

public void clickOnNextButtonOfMemberDetailsPage()

{

Reporter.log("clicking On NextButton Of MemberDetails Page", true);

nextButton.click();

}

}

package coverFoxPOM;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

import org.testng.Reporter;

public class CoverFoxAddressDetailsPage {

@FindBy(className = "mp-input-text")

private WebElement pinCodeFiled;

@FindBy(id = "want-expert")

private WebElement mobileNumberField;

@FindBy(xpath = "//div[text()='Continue']")

private WebElement continueButton;

public CoverFoxAddressDetailsPage(WebDriver driver) {

PageFactory.initElements(driver, this);

}

public void enterPinCode(String pinCode) {

Reporter.log("entering PinCode", true);

pinCodeFiled.sendKeys(pinCode);

}

public void enterMobileNumber(String mobileNumber) {

Reporter.log("entering mobileNumber", true);

mobileNumberField.sendKeys(mobileNumber);

}

public void clickOnContinueButton() {

Reporter.log("clicking On Continue Button", true);

continueButton.click();

}

}

package coverFoxPOM;

import java.util.List;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

import org.testng.Reporter;

public class CoverFoxResultPage {

@FindBy(xpath = "//div[contains(text(),'matching')]") private WebElement planResult;

@FindBy(className = "plan-card-container") private List<WebElement> planCards;

public CoverFoxResultPage(WebDriver driver)

{

PageFactory.initElements(driver, this);

}

public void validatePlans() throws InterruptedException

{

String result = planResult.getText();

String[] result1 = result.split(" ");

//48 matching Health Insurance Plans

int resultInNumber = Integer.parseInt(result1[0]);

Thread.sleep(8000);

int totalPlans = planCards.size();

if(totalPlans==resultInNumber)

{

System.out.println("TC is passed");

}

else {

System.out.println("TC is failed");

}

}

public int getPlanNumberFromText()

{

Reporter.log("getting plan number from text", true);

String result = planResult.getText();

String[] result1 = result.split(" ");

int planNumber = Integer.parseInt(result1[0]);

return planNumber;

}

public int getPlanNumberFromPlanCards()

{

Reporter.log("getting plan number from plan cards", true);

int planNumberFromCards = planCards.size();

return planNumberFromCards;

}

}

**package** coverFoxPOM;

**import** java.io.FileInputStream;

**import** java.io.IOException;

**import** java.time.Duration;

**import** org.apache.poi.EncryptedDocumentException;

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

**import** org.apache.poi.ss.usermodel.WorkbookFactory;

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

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

**import** org.testng.Assert;

**import** org.testng.Reporter;

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

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

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

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

**public** **class** CoverFoxTestUsingTestNG {

WebDriver driver;

FileInputStream myFile;

Sheet mySheet;

CoverFoxHomePage coverFoxHomePage;

CoverFoxMemberDetailsPage coverFoxMemberDetailsPage;

CoverFoxAddressDetailsPage coverFoxAddressDetailsPage;

CoverFoxHealthPlanPage coverFoxHealthPlanPage;

CoverFoxResultPage coverFoxResultPage;

@BeforeClass

**public** **void** launchBrowser() **throws** EncryptedDocumentException, IOException

{

driver = **new** ChromeDriver();

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

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

Reporter.*log*("opening browser, launching url", **true**);

driver.manage().timeouts().implicitlyWait(Duration.*ofMillis*(2000));

// excel data

myFile = **new** FileInputStream("D:\\VCITY\\Java Class\\23 Nov 2024\\23rd Nov Morning2024.xlsx");

mySheet = WorkbookFactory.*create*(myFile).getSheet("Sheet7");

coverFoxHomePage= **new** CoverFoxHomePage(driver);

coverFoxAddressDetailsPage= **new** CoverFoxAddressDetailsPage(driver);

coverFoxMemberDetailsPage= **new** CoverFoxMemberDetailsPage(driver);

coverFoxHealthPlanPage= **new** CoverFoxHealthPlanPage(driver);

coverFoxResultPage= **new** CoverFoxResultPage(driver);

}

@BeforeMethod

**public** **void** coverFoxPreconditions() **throws** InterruptedException

{

coverFoxHomePage.clickOnGender();

coverFoxHealthPlanPage.clickOnNextButtonHealthPlanPage();

coverFoxMemberDetailsPage.handleAgeDropDown(mySheet.getRow(0).getCell(0).getStringCellValue());

coverFoxMemberDetailsPage.clickOnNextButtonOfMemberDetailsPage();

coverFoxAddressDetailsPage.enterPinCode(mySheet.getRow(0).getCell(1).getStringCellValue());

coverFoxAddressDetailsPage.enterMobileNumber(mySheet.getRow(0).getCell(2).getStringCellValue());

coverFoxAddressDetailsPage.clickOnContinueButton();

Thread.*sleep*(4000);

}

@Test

**public** **void** validateCoverFoxPlans() {

**int** planNumberFromText = coverFoxResultPage.getPlanNumberFromText();

**int** planNumberFromCards = coverFoxResultPage.getPlanNumberFromPlanCards();

Assert.*assertEquals*(planNumberFromText, planNumberFromCards,"Test case failed, number are not matching");

Reporter.*log*("Plan number are matching TC is passed", **true**);

}

@AfterClass

**public** **void** closeBrowser() **throws** InterruptedException

{

Reporter.*log*("closing browser", **true**);

Thread.*sleep*(2000);

driver.quit();

}

}

Utility Class

**package** coverFoxPOMBaseTestUtility;

**import** java.io.File;

**import** java.io.FileInputStream;

**import** java.io.IOException;

**import** org.apache.poi.EncryptedDocumentException;

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

**import** org.apache.poi.ss.usermodel.WorkbookFactory;

**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.io.FileHandler;

**import** org.testng.Reporter;

**public** **class** Utility {

//readData from excel

**public** **static** String readDataFromExcel(String sheetName, **int** row, **int** cell) **throws** EncryptedDocumentException, IOException

{

FileInputStream myFile = **new** FileInputStream("D:\\VCITY\\Java Class\\23 Nov 2024\\23rd Nov Morning2024.xlsx");

Sheet mySheet = WorkbookFactory.*create*(myFile).getSheet(sheetName);

String data = mySheet.getRow(row).getCell(cell).getStringCellValue();

Reporter.*log*("reading data from excel", **true**);

**return** data;

}

//screenshot

**public** **static** **void** takeScreenshot(WebDriver driver, String screenShotName) **throws** IOException

{

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

File dest= **new** File("D:\\new eclipse\\Nov23Morning24\_Selenium\\screenshot\\"+screenShotName+".png");

FileHandler.*copy*(src, dest);

Reporter.*log*("taking screenshot, svaed at"+dest, **true**);

}

//scroll into view

**public** **static** **void** scroolIntoView(WebDriver driver, WebElement element)

{

JavascriptExecutor js = (JavascriptExecutor)driver;

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

Reporter.*log*("scrolling into view", **true**);

}

}

**public** **static** String readDataFromPropertiesFile(String key) **throws** IOException

{

Properties prop= **new** Properties();

FileInputStream myFile= **new** FileInputStream(System.*getProperty*("user.dir")+"\\coverFox.properties");

prop.load(myFile);

String value = prop.getProperty(key);

Reporter.*log*("reading "+key+ " from properties file", **true**);

**return** value;

}

//how to add implicit wait

**public** **void** implicitWait(WebDriver driver,**int** timeInMilisec)

{

Reporter.*log*("waiting for",timeInMilisec,**true**);

driver.manage().timeouts().implicitlyWait(Duration.*ofMillis*(timeInMilisec));

}

//sendkeys

**public** **void** sendkeys(WebElement element,String text)

{

element.sendKeys(text);

}

Base class

**package** coverFoxPOMBaseTestUtility;

**import** java.time.Duration;

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

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

**import** org.testng.Reporter;

**public** **class** Base

{

WebDriver driver;

**public** **void** launchBrowser()

{

driver= **new** ChromeDriver();

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

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

driver.manage().timeouts().implicitlyWait(Duration.*ofMillis*(2000));

Reporter.*log*("launching Browser", **true**);

}

**public** **void** closeBrowser()

{

Reporter.*log*("closing Browser", **true**);

driver.quit();

}

}

Properties File

**Property file selenium**

Step 1) Creating a properties file in eclipse

1. Right-click on the main project folder and Select New-> Other->select General -> File and click on 'Next' button->

Provide a valid file name with the extension '.properties' on the new file resource window and click on 'Finish' button.

Step 2) Storing data into properties fileàData is stored in properties file in the form of key-value pairs, with the key being unique across the file.

Open file in Eclipse and store some data

url=<https://www.coverfox.com/>

userName=Test@123

Password=Pune@1234

Step 3) Reading data from properties file

Properties obj = new Properties();

FileInputStream objfile = new FileInputStream(System.getProperty("user.dir")+"\\credentials.properties");

obj.load(objfile);

String value= obj.getProperty("URL");

Create property files



Reading Property file



TestBaseUtilityTestNGTestclass

package coverFoxPOMBaseTestUtility;

import java.io.File;

import java.io.FileInputStream;

import java.io.IOException;

import java.time.Duration;

import org.apache.poi.EncryptedDocumentException;

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

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

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.io.FileHandler;

import org.testng.Assert;

import org.testng.Reporter;

import org.testng.annotations.AfterClass;

import org.testng.annotations.BeforeClass;

import org.testng.annotations.BeforeMethod;

import org.testng.annotations.Test;

import coverFoxPOMBaseTestUtility.Utility;

import net.bytebuddy.utility.RandomString;

public class CoverFoxTestUsingTestNG extends Base{

CoverFoxHomePage coverFoxHomePage;

CoverFoxMemberDetailsPage coverFoxMemberDetailsPage;

CoverFoxAddressDetailsPage coverFoxAddressDetailsPage;

CoverFoxHealthPlanPage coverFoxHealthPlanPage;

CoverFoxResultPage coverFoxResultPage;

@BeforeClass

public void launchBrowser() throws EncryptedDocumentException, IOException

{

openBrowser();

coverFoxHomePage= new CoverFoxHomePage(driver);

coverFoxAddressDetailsPage= new CoverFoxAddressDetailsPage(driver);

coverFoxMemberDetailsPage= new CoverFoxMemberDetailsPage(driver);

coverFoxHealthPlanPage= new CoverFoxHealthPlanPage(driver);

coverFoxResultPage= new CoverFoxResultPage(driver);

}

@BeforeMethod

public void coverFoxPreconditions() throws InterruptedException, EncryptedDocumentException, IOException

{

coverFoxHomePage.clickOnGender();

coverFoxHealthPlanPage.clickOnNextButtonHealthPlanPage();

coverFoxMemberDetailsPage.handleAgeDropDown(Utility.readDataFromExcel("Sheet7", 1, 0));

coverFoxMemberDetailsPage.clickOnNextButtonOfMemberDetailsPage();

coverFoxAddressDetailsPage.enterPinCode(Utility.readDataFromExcel("Sheet7", 1, 1));

coverFoxAddressDetailsPage.enterMobileNumber(Utility.readDataFromExcel("Sheet7", 1, 2));

coverFoxAddressDetailsPage.clickOnContinueButton();

Thread.sleep(4000);

}

@Test

public void validateCoverFoxPlans() throws IOException {

int planNumberFromText = coverFoxResultPage.getPlanNumberFromText();

int planNumberFromCards = coverFoxResultPage.getPlanNumberFromPlanCards();

Assert.assertEquals(planNumberFromText, planNumberFromCards,"Test case failed, number are not matching");

Reporter.log("Plan number are matching TC is passed", true);

Utility.takeScreenshot(driver, "validateCoverFoxPlans");

}

@AfterClass

public void closeBrowser() throws InterruptedException

{

closeBrowserWindow();

}

}