**Pages**

package in.amazon.pages;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.interactions.Actions;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

public class LandingPage {

private Actions actions;

@FindBy(linkText = "Mobiles")

private WebElement mobiles;

@FindBy(id = "nav-link-accountList")

private WebElement signInMenu;

@FindBy(linkText = "Sign in")

private WebElement signInBtn;

public LandingPage(WebDriver driver) {

PageFactory.initElements(driver, this);

actions = new Actions(driver);

}

public void clickMobiles() {

mobiles.click();

}

public void hoverOverHelloSignInMenu() {

actions.moveToElement(signInMenu).build().perform();

}

public void clickSignInBtn() {

signInBtn.click();

}

}

package in.amazon.pages;

import java.time.Duration;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.interactions.Actions;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

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

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

public class AllMobileBrands {

private Actions actions;

private WebDriverWait wait;

@FindBy(linkText = "Mobiles & Accessories")

private WebElement mobilesAndAccessories;

@FindBy(linkText = "Apple")

private WebElement apple;

public AllMobileBrands(WebDriver driver) {

PageFactory.initElements(driver, this);

actions = new Actions(driver);

wait = new WebDriverWait(driver, Duration.ofSeconds(60));

}

public void hoverOverMobilesAndAccessories() {

actions.moveToElement(mobilesAndAccessories).build().perform();

}

public void clickApple()

{

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

apple.click();

}

}

package in.amazon.pages;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

public class ApplePhones {

@FindBy(xpath = "(//div[contains(@class, 'card-container')])[2]")

private WebElement secondMobile;

public ApplePhones(WebDriver driver) {

PageFactory.initElements(driver, this);

}

public void clickSecondMobile() {

secondMobile.click();

}

}

package in.amazon.pages;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

public class BuyPhone {

@FindBy(id = "buy-now-button")

private WebElement buyNowBtn;

public BuyPhone(WebDriver driver) {

PageFactory.initElements(driver, this);

}

public void clickBuyNowBtn() {

buyNowBtn.click();

}

}

package in.amazon.pages;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.WebElement;

import org.openqa.selenium.support.FindBy;

import org.openqa.selenium.support.PageFactory;

public class SignIn {

@FindBy(xpath = "//h1[contains(@class,'small')]")

private WebElement signintext;

@FindBy(id ="ap\_email")

private WebElement emailTextBox;

@FindBy(id = "continue")

private WebElement continueBtn;

//span[contains(@class,'a-list-item')]

@FindBy(xpath ="//span[contains(@class,'a-list-item')]")

private WebElement errMsg;

public SignIn(WebDriver driver) {

PageFactory.initElements(driver, this);

}

public String getSignInText() {

String text= signintext.getText();

return text;

}

public void enterEMail(String email) {

emailTextBox.sendKeys(email);

}

public void clickContinueBtn() {

continueBtn.click();

}

public String getErrMsg() {

String message = errMsg.getText();

return message;

}

}

**TestScripts**

BeforeAfter

**package** in.amazon.testSteps;

**import** io.cucumber.java.After;

**import** io.cucumber.java.Before;

**import** io.cucumber.java.Scenario;

**public** **class** BeforeAfter **extends** Tools {

@Before

**public** **void** setUp(Scenario scenario) {

Driver.*init*();

}

@After

**public** **void** tearDown() {

*driver*.quit();

}

}

**package** in.amazon.testSteps;

**import** java.util.ArrayList;

**import** org.testng.Assert;

**import** io.cucumber.java.en.Given;

**import** io.cucumber.java.en.Then;

**import** io.cucumber.java.en.When;

**public** **class** BuyMobilePhoneSteps **extends** Driver{

@Given("a user is on the landing page of Amazon")

**public** **void** a\_user\_is\_on\_the\_landing\_page\_of\_amazon() {

}

@When("he clicks on Mobiles in the navigation bar")

**public** **void** he\_clicks\_on\_mobiles\_in\_the\_navigation\_bar() {

*landingPage*.clickMobiles();

}

@When("he hovers the pointer over Mobiles & Accessories")

**public** **void** he\_hovers\_the\_pointer\_over\_mobiles\_accessories() {

*allMobileBrands*.hoverOverMobilesAndAccessories();

}

@When("he clicks on Apple in the sub-menu")

**public** **void** he\_clicks\_on\_apple\_in\_the\_sub\_menu() {

*allMobileBrands*.clickApple();

}

@When("he clicks on the second available phone")

**public** **void** he\_clicks\_on\_the\_second\_available\_phone() {

*applePhones*.clickSecondMobile();

}

@When("he switches focus on the new tab")

**public** **void** he\_switches\_focus\_on\_the\_new\_tab() {

ArrayList<String> tabs = **new** ArrayList<>(*driver*.getWindowHandles());

*driver*.switchTo().window(tabs.get(1));

}

@When("he clicks on Buy Now button")

**public** **void** he\_clicks\_on\_buy\_now\_button() {

*buyPhone*.clickBuyNowBtn();

}

@Then("he must be able to purchase the mobiles phone successfully.")

**public** **void** he\_must\_be\_able\_to\_purchase\_the\_mobiles\_phone\_successfully() {

String expectedText = "Sign in";

String actualText = *signIn*.getSignInText();

Assert.*assertEquals*(actualText,expectedText);

}

}

package in.amazon.testSteps;

import org.openqa.selenium.chrome.ChromeDriver;

import in.amazon.pages.AllMobileBrands;

import in.amazon.pages.ApplePhones;

import in.amazon.pages.BuyPhone;

import in.amazon.pages.LandingPage;

import in.amazon.pages.SignIn;

public class Driver extends Tools {

//protected static WebDriver driver;

protected static AllMobileBrands allMobileBrands;

protected static LandingPage landingPage;

protected static ApplePhones applePhones;

protected static BuyPhone buyPhone;

protected static SignIn signIn;

public static void init() {

driver = new ChromeDriver();

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

driver.get("https://amazon.in");

//allMobileBrands = new AllMobileBrands(driver);

landingPage = new LandingPage(driver);

allMobileBrands = new AllMobileBrands(driver);

applePhones = new ApplePhones(driver);

buyPhone = new BuyPhone(driver);

signIn = new SignIn(driver);

}

}

package in.amazon.testSteps;

import io.cucumber.testng.AbstractTestNGCucumberTests;

import io.cucumber.testng.CucumberOptions;

@CucumberOptions(

features = "Features",

glue = "in.amazon.testscripts"

//dryRun = true

//tags = "@product\_purchase or @login\_functionality"

)

public class TestRunner extends AbstractTestNGCucumberTests{

}

**package** in.amazon.testSteps;

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

**public** **class** Tools {

**protected** **static** WebDriver *driver*;

}

Feature: Buy Mobile Phone

@TC\_101 @product\_purchase

Scenario: A user must be able to buy a mobile phone on Amazon

Given a user is on the landing page of Amazon

When he clicks on Mobiles in the navigation bar

And he hovers the pointer over Mobiles & Accessories

And he clicks on Apple in the sub-menu

And he clicks on the second available phone

And he switches focus on the new tab

And he clicks on Buy Now button

Then he must be able to purchase the mobiles phone successfully.

Feature: Login functionality

Background:

Given a user is on the loading page of Amazon

When he hovers the pointer over Hello Sign-in menu

And he clicks on Sign-in button in the sub-menu

@TC\_102 @login\_functionality

Scenario: A user with invalid credential must get an error message while logging in

And he enteers invalid email address in the Email text-box

And he clicks on continue button

Then he must see the error message - "We cannot find an account with that email address"

Scenario Outline: A user with multiple invalid credentials must get an error message while logging in

And he enters an invalid email address "<emailId>" in the Email text-box

And he clicks on continue button

Then he must see the error message - "We cannot find an account with that email address"

Examples:

|emailId|

|batman554466@gmail.com|

|superman554466@gmail.com|

|spiderman554466@gmail.com|

|ironman554466@gmail.com|

|captainplanet554466@gmail.com|