9)

a)FBRegistration.java

**package** com.ecommerce.test;

**import** java.util.List;

**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.support.ui.Select;

**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.DataProvider;

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

@Test(groups = "Class1")

**public** **class** FBRegistration {

**public** WebDriver driver;

**public** String baseUrl = "File:///C:\\Users\\Thaneeru Anil Kumar\\eclipse-workspace3\\hello-selinium\\src\\main\\resources\\test.html";

@Test(groups = { "Google" })

**public** **void** searchGoogle() {

System.***out***.printf("Inside %s and thread-id is %s \n", "searchGoogle", Thread.*currentThread*().getId());

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

System.***out***.println("Title of google page is " + driver.getTitle());

}

@Test(groups = { "Account Creation" })

**public** **void** fbAccount() {

System.***out***.printf("Inside %s and thread-id is %s \n", "fbAccount", Thread.*currentThread*().getId());

String baseUrl = "https://www.facebook.com/r.php?locale=en\_GB&display=page";

driver.get(baseUrl);

String cssDay = "#day";

WebElement cssDaySelect = driver.findElement(By.*cssSelector*(cssDay));

Select daySelect = **new** Select(cssDaySelect);

daySelect.selectByVisibleText("11");

WebElement cssGenderRadio = driver.findElement(By.*cssSelector*("span > span > input[type='radio'][value='2']"));

cssGenderRadio.click();

System.***out***.println("Gender is enabled = " + cssGenderRadio.isSelected());

}

@Test(groups = { "Account Creation", "Google" }, dataProvider = "googleNewUserData")

**public** **void** googleAccount(String fName, String lName) {

System.***out***.printf("Inside %s and thread-id is %s \n", "googleAccount", Thread.*currentThread*().getId());

String baseUrl = "https://accounts.google.com/signup/v2?biz=true&flowEntry=SignUp";

driver.get(baseUrl);

// Let's locate the first name text field by its id.

WebElement firstNameTF = driver.findElement(By.*id*("firstName"));

firstNameTF.sendKeys(fName) ;//("Myname");

// Let's locate the surname name text field by its name.

WebElement surNameTF = driver.findElement(By.*name*("lastName"));

surNameTF.sendKeys(lName);//("latName");

// The Next button using its class

WebElement nextButton = driver.findElement(By.*className*("VfPpkd-vQzf8d"));

System.***out***.println("Text on the button is " + nextButton.getText());

}

@Test(groups = { "Dummy" }, dependsOnGroups = { "Account Creation" })

**public** **void** f1() {

System.***out***.printf("Inside %s and thread-id is %s \n", "f1", Thread.*currentThread*().getId());

}

@Test(enabled = **false**)

**public** **void** f2() {

System.***out***.printf("Inside %s and thread-id is %s \n", "f2", Thread.*currentThread*().getId());

}

@Test()

**public** **void** testLocalPage() {

System.***out***.printf("Inside %s and thread-id is %s \n", "testLocalPage", Thread.*currentThread*().getId());

driver.get(baseUrl);

WebElement selectMonthElement = driver.findElement(By.*id*("month"));

Select selectMonth = **new** Select(selectMonthElement);

System.***out***.println("selectMonth is muti select " + selectMonth.isMultiple());

selectMonth.selectByIndex(0);

selectMonth.selectByIndex(6);

List<WebElement> optionsList = selectMonth.getAllSelectedOptions();

**for** (WebElement option : optionsList)

System.***out***.println("selectMonth selected options are " + option.getText());

}

@BeforeMethod

**public** **void** beforeMethod() {

System.***out***.printf("Inside %s and thread-id is %s \n", "beforeMethod", Thread.*currentThread*().getId());

}

@AfterMethod

**public** **void** afterMethod() {

System.***out***.printf("Inside %s and thread-id is %s \n", "afterMethod", Thread.*currentThread*().getId());

}

@DataProvider(name = "googleNewUserData")

**public** Object[][] googleNewUserData() {

**return** **new** Object[][] {

{ "fname1", "lname1" },

{ "fname2", "lname2" }

};

}

@BeforeClass

**public** **void** beforeClass() {

System.***out***.printf("Inside %s and thread-id is %s \n", "beforeClass", Thread.*currentThread*().getId());

}

@AfterClass

**public** **void** afterClass() {

System.***out***.printf("Inside %s and thread-id is %s \n ", "afterClass", Thread.*currentThread*().getId());

}

@BeforeTest

**public** **void** beforeTest() {

System.***out***.printf("Inside %s and thread-id is %s \n", "beforeTest", Thread.*currentThread*().getId());

}

@AfterTest

**public** **void** afterTest() {

System.***out***.printf("Inside %s and thread-id is %s \n", "afterTest", Thread.*currentThread*().getId());

}

@BeforeSuite

**public** **void** beforeSuite() {

System.***out***.printf("Inside %s and thread-id is %s \n", "beforeSuite", Thread.*currentThread*().getId());

// System.setProperty("webdriver.chrome.driver","C:\\Users\\HomeWk\\Downloads\\chromedriver\_win32\\chromedriver.exe");

driver = **new** ChromeDriver();

}

@AfterSuite

**public** **void** afterSuite() {

System.***out***.printf("Inside %s and thread-id is %s \n", "afterSuite", Thread.*currentThread*().getId());

driver.quit();

}

}

b)testing.xml

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

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

<suite name=*"Suite"*>

<!-- <test thread-count="5" name="Test FaceBook" parallel="methods"> -->

<test thread-count=*"5"* name=*"Test FaceBook"* parallel=*"methods"*>

<classes>

<class name=*"com.ecommerce.test.FBRegistration"*/>

</classes>

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

<!-- Test

<test thread-count="5" name="Test Google">

<classes>

<class name="com.ecommerce.test.GoogleRegistration"/>

</classes>

</test>

-->

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

Output:

