**Assisted Practice: 2.2 Group Test Cases and Parallel Test Execution**

This section will guide you to:

* Work with groups attribute of @Test
* Perform cross browser execution (parallel execution)

**Development Environment**

* Eclipse IDE for Enterprise Java Developers v2019-03 (4.11.0)
* JRE: OpenJDK Runtime Environment 11.0.2
* TestNG
* Selenium WebDriver Jar

This lab has eight subsections, namely:

2.2.1 Creating a simple Java project

2.2.2 Downloading Selenium WebDriver jar, chromdriver.exe, and forefoxdriver.exe

2.2.3 Adding the Web Driver dependency in the project

2.2.4 Installing TestNG

2.2.5 Adding TestNG libraries to the Class Path

2.2.6 Creating a Java class named ParallelTest.java

2.2.7 Running the project

2.2.8 Pushing the code to your GitHub repositories

**Step 2.2.1:** Creating a simple Java project

* Open Eclipse
* Go the **File** menu. Choose **New->Java Project**
* Enter the project name as **Parallel Tests**. Click on **Next**
* This will create the project files in the Project Explorer

**Step 2.2.2:** Downloading Selenium WebDriver jar, chromdriver.exe, and forefoxdriver.exe

* Go to <https://www.seleniumhq.org/download/> to download the Selenium WebDriver dependency
* Under the section **Selenium Client & WebDriver Language Bindings,** click on **Download** for **Java client version: 3.141.59**
* On the same page, under **Third Party Drivers, Bindings, and Plugins,** click on **Latest** for **Mozilla Gecko Driver**
* Select the file suitable for your operating system
* Go back to the previous page. Click on **Latest** for **Google Chrome Driver**
* From the current releases, select the appropriate file per your Chrome version

**Step 2.2.3:** Adding the Web Driver dependency in the project

* In the Project Explorer, right click on **Parallel Tests**
* Select **Properties**. Select **Java Build Pat**h from the list. Go to **Libraries**.
* Click on **Add External JARs** and browse the location where you have downloaded the JAR files
* Select JARs from the **root** folder and the **libs** folder
* Click on **Apply and Close**
* Copy the **chromedriver.exe** and **geckodriver.exe**, and paste it your project creating a resource folder

**Step 2.2.4:** Installing TestNG

* TestNG is installed as an eclipse plugin in your practice lab. (Refer FSD: Lab Guide - Phase 5)

**Step 2.2.5:** Adding TestNG libraries to the Class Path

* In the Project Explorer, right click on **Parallel Tests**
* Select **Properties**. Select **Java Build Pat**h from the list. Go to **Libraries**
* Click on **Add Library.** Select **TestNG**. Click on **Next**. Click on **Finish**
* Click on **Apply and Close**

**Step 2.2.6:** Creating a Java class named ParallelTest.java

* In the Project Explorer, expand **Parallel Tests->Java Resources**
* Right click on **src** and choose **New->Class**
* In **Class Name,** enter **ParallelTests** and click on **Finish.** In **Package Name,** enter **com.parallel** and click on **Finish**
* Enter the following code:

package com.parallel;

import org.openqa.selenium.By;

import org.openqa.selenium.WebDriver;

import org.openqa.selenium.chrome.ChromeDriver;

import org.openqa.selenium.firefox.FirefoxDriver;

import org.testng.annotations.Test;

public class ParallelTests {

WebDriver driver;

@Test(groups="Chrome")

public void LaunchChrome() {

System.setProperty("webdriver.chrome.driver", "./Resources/chromedriver.exe");

driver = new ChromeDriver();

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

try {

Thread.sleep(2000);

} catch (Exception e) {

e.printStackTrace();

}

}

@Test(groups="Chrome", dependsOnMethods="LaunchChrome")

public void TryFacebook1() {

System.out.println(Thread.currentThread().getId());

driver.findElement(By.id("email")).sendKeys("madhugadekar56@gmail.com");

driver.findElement(By.id("pass")).sendKeys("12345");

driver.findElement(By.id("loginbutton")).click();

}

@Test(groups="Firefox")

public void LaunchFirefox() {

System.setProperty("webdriver.gecko.driver", "./Resources/geckodriver.exe");

driver = new FirefoxDriver();

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

try {

Thread.sleep(4000);

} catch (Exception e) {

e.printStackTrace();

}

}

@Test(groups="Firefox", dependsOnMethods="LaunchFirefox")

public void TryFacebook2() {

System.out.println(Thread.currentThread().getId());

driver.findElement(By.id("email")).sendKeys("ravi10thstudent@gmail.com");

driver.findElement(By.id("pass")).sendKeys("ravi28394");

driver.findElement(By.id("loginbutton")).click();

System.out.println(Thread.currentThread().getId());

}

}

**Step 2.2.7** Running the project

* Right click on **ParallelTests** class. Click on **TestNG->Convert to TestNG**
* Click on **Finish.** It will create a **TestNG.xml** file. Open that file
* Right click. Select **Run As ->TestNG Suite**

**Step 2.2.8:** Pushing the code to your GitHub repositories

* Open your command prompt and navigate to the folder where you have created your files.

**cd <folder path>**

* Initialize your repository using the following command:

**git init**

* Add all the files to your git repository using the following command:

**git add .**

* Commit the changes using the following command:

**git commit . -m “Changes have been committed.”**

* Push the files to the folder you initially created using the following command:

**git push -u origin master**