

1. Create 2 test cases, disable one using enabled = false, and run only the active test.

```
package asserts;
```

```
import org.testng.annotations.Test;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.BeforeMethod;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.SkipException;
import org.testng.annotations.AfterClass;
import org.testng.annotations.AfterMethod;
```

```
public class Task1 {
    WebDriver driver;
```

```
    @BeforeMethod
```

```
    public void beforeMethod() {
        driver=new ChromeDriver();
        driver.get("https://www.selenium.dev/");
        driver.manage().window().maximize();
    }
```

```
    @Test(enabled=false)
```

```
    public void title() {
        String expectedTitle="Selenium dev";
        String actualTitle=driver.getTitle();
        Assert.assertEquals(expectedTitle, actualTitle,"Title validation failed");
        System.out.println("expectedTitle "+expectedTitle);
        System.out.println("actualTitle "+actualTitle);
    }
```

```
@Test
public void url() {
    String expectedUrl="https://www.selenium.dev/";
    String actualUrl=driver.getCurrentUrl();
    Assert.assertEquals(expectedUrl, actualUrl,"Url validation failed");
    System.out.println("expectedUrl "+expectedUrl);
    System.out.println("actualUrl "+actualUrl);
}

@AfterMethod
public void afterMethod() {
    driver.close();
}

}
```

2. Write a test to run the same test multiple times.

```
package asserts;

import org.testng.annotations.Test;
import org.testng.annotations.BeforeClass;

import java.time.Duration;

import org.openqa.selenium.By;
import org.openqa.selenium.Keys;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.WebElement;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterClass;

public class Task2 {
    WebDriver driver;

    @BeforeClass
    public void beforeClass() {
        driver=new ChromeDriver();
        driver.get("https://www.ebay.com/");
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(5));
    }

    @Test
    public void search() throws InterruptedException {
        for(int i=1;i<=2;i++) {
            String[] list= {"Watch","Electronics","Sports"};
            for(String sr:list) {
                WebElement search=driver.findElement(By.id("gh-ac"));
```

```
        search.clear();
        search.sendKeys(sr);
        search.sendKeys(Keys.ENTER);
        Thread.sleep(2000);
    }
}

@AfterClass
public void afterClass() {
    driver.close();
}

}
```

3. Write test cases for a dummy login page using @Parameters in testng.xml.

```
package asserts;

import org.testng.annotations.Test;
import org.testng.annotations.Parameters;
import org.testng.annotations.BeforeClass;
import org.testng.annotations.Optional;
import java.time.Duration;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterClass;

public class Task3 {
    WebDriver driver;

    @BeforeClass
    public void beforeTest() {
        driver=new ChromeDriver();
        driver.get("http://zero.webappsecurity.com/login.html");
        driver.manage().window().maximize();
        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(5));
    }

    @Test
    @Parameters({"username", "password"})

    public void loginTest(@Optional("defaultUser") String username,
        @Optional("defaultPass") String password) {
        System.out.println("Username: " + username + ", Password: " + password);
        driver.findElement(By.id("user_login")).sendKeys(username);
        driver.findElement(By.id("user_password")).sendKeys(password);
    }
}
```

```

        driver.findElement(By.name("submit")).click();
    }
    @AfterClass
    public void afterTest() {
        driver.close();
    }
}

```

Testng.xml

```

<suite name="LoginSuite" parallel="classes" thread-count="1" >
    <test name="ValidLogin">
        <parameter name="username" value="username"/>
        <parameter name="password" value="password"/>
        <classes>
            <class name="asserts.Task3"/>
        </classes>
    </test>

    <test name="InvalidLogin">
        <parameter name="username" value="username"/>
        <parameter name="password" value="pass"/>
        <classes>
            <class name="asserts.Task3"/>
        </classes>
    </test>
</suite>

```

4. Write dependent test cases:

login()

search Product() (depends on login)

logout() (depends on search)

```
package asserts;
```

```
import org.testng.annotations.Test;
```

```
import org.testng.annotations.BeforeClass;
```

```
import java.time.Duration;
```

```
import org.openqa.selenium.By;
```

```
import org.openqa.selenium.Keys;
```

```
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.WebDriverWait;
```

```
import org.testng.annotations.AfterClass;
```

```
public class Task4 {
```

```
    WebDriver driver;
```

```
    @BeforeClass
```

```
    public void beforeClass() {
```

```
        driver=new ChromeDriver();
```

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

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

```
        driver.manage().timeouts().implicitlyWait(Duration.ofSeconds(5));
```

```
    }
```

```

@Test
public void login() {
    WebElement
btn1=driver.findElement(By.xpath("//*[@id=\"gh\"]/nav/div[1]/span[1]/span/a"));
    btn1.click();

    WebElement email=driver.findElement(By.id("userid"));
    email.sendKeys("ninadnasikkar1907@gmail.com");

    WebElement btn=driver.findElement(By.id("signin-continue-btn"));
    btn.click();

    WebElement password=driver.findElement(By.id("pass"));
    password.sendKeys("*****");

    WebElement signin=driver.findElement(By.id("sgnBt"));
    signin.click();
}

```

```

@Test(dependsOnMethods= {"login"})
public void search() throws InterruptedException{
    String[] list= {"Watch","Electronics","Sports"};
    for(String sr:list) {
        WebElement search=driver.findElement(By.id("gh-ac"));
        search.clear();
        search.sendKeys(sr);
        search.sendKeys(Keys.ENTER);
        Thread.sleep(2000);
    }
}

```



```
}
```

```
@Test(dependsOnMethods= {"login"})
```

```
public void signOut() throws InterruptedException {
```

```
    Actions act=new Actions(driver);
```

```
    WebElement
```

```
name=driver.findElement(By.cssSelector("div>button[aria-controls=\"s0-1-4-9-3[0]-0-9-dialog\""
]));
```

```
    act.moveToElement(name).perform();
```

```
    driver.findElement(By.partialLinkText("Sign out")).click();
```

```
}
```

```
@AfterClass
```

```
public void afterClass() {
```

```
    driver.close();
```

```
}
```

```
}
```

5. Use Data Provider to supply multiple sets of usernames/passwords to a login test.

```
package testNG_package;

import java.io.FileInputStream;
import java.io.IOException;
import org.openqa.selenium.By;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.DataProvider;
import org.testng.annotations.Test;
import org.apache.poi.ss.usermodel.*;

public class ExcelTest {

    @DataProvider(name="excelData")
    public Object[][] excelDataProvider() throws IOException {
        String excelPath="C:\\Wipro_Training\\LoginData.xlsx";
        String sheetName="Sheet1";

        FileInputStream fis=new FileInputStream(excelPath);
        Workbook workbook=WorkbookFactory.create(fis);
        Sheet sheet=workbook.getSheet(sheetName);

        int rows=sheet.getPhysicalNumberOfRows();
        int cols=sheet.getRow(0).getLastCellNum();

        Object[][] data=new Object[rows-1][cols];

        for(int i=1;i<rows;i++) {
```

```

        Row row = sheet.getRow(i);
for (int j = 0; j < cols; j++) {
    if (row == null) {
        data[i - 1][j] = ""; // blank row
    }
    else {
        Cell cell = row.getCell(j);
        data[i - 1][j] = (cell == null) ? "" : cell.toString();
    }
}

```

```

        workbook.close();

```

```

        fis.close();

```

```

        return data;

```

```

    }

```

```

@Test(dataProvider="excelData")

```

```

    public void testLogin(String username,String password) {
        System.out.println("Username: "+username+" | Password: "+password);

```

```

        WebDriver driver=new ChromeDriver();

```

```

        driver.get("http://zero.webappsecurity.com/login.html");

```

```

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

```

```

        driver.findElement(By.id("user_login")).sendKeys("Username1");

```

```

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

```

```

        driver.findElement(By.id("user_password")).sendKeys("Password1");

```

```

        driver.findElement(By.name("submit")).click();

```

```

    }

```

```

}

```

6. Run test cases in parallel (methods, classes, tests) using parallel attribute in testng.xml.

```
package parallel_testing;

import org.testng.annotations.Test;
import org.testng.annotations.BeforeClass;
import org.openqa.selenium.WebDriver;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.annotations.AfterClass;

public class TestClass1 {
    WebDriver driver;

    @BeforeClass
    public void beforeClass() throws InterruptedException {
        Thread.sleep(3000);
    }

    @Test
    public void tiraBeauty() {
        driver = new ChromeDriver();

        driver.get("https://www.tirabeauty.com/?srsId=AfmBOorwVJ-wwgjgzVoGyPn4j8ZJVqNG7qTWvJ0obqx2B3slUmXYNlsa");
        System.out.println("Tira beauty Thread ID = "+Thread.currentThread().getId());
        driver.quit();
    }

    @AfterClass
    public void afterClass() {
        driver.quit();
    }
}
```

```
}
```

```
}
```

```
*****
```

```
package parallel_testing;
```

```
import org.testng.annotations.Test;
```

```
import org.testng.annotations.BeforeClass;
```

```
import org.openqa.selenium.WebDriver;
```

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

```
import org.testng.annotations.AfterClass;
```

```
public class TestClass2 {
```

```
    WebDriver driver;
```

```
    @BeforeClass
```

```
    public void beforeClass() throws InterruptedException {
```

```
        Thread.sleep(3000);
```

```
    }
```

```
    @Test
```

```
    public void motorola() {
```

```
        driver = new ChromeDriver();
```

```
        driver.get("http://motorola.in/");
```

```
        System.out.println("Ebay Thread ID = "+Thread.currentThread().getId());
```

```
        driver.quit();
```

```
    }
```

```
    @AfterClass
```

```
public void afterClass() {  
    driver.quit();  
}  
  
}
```

Testng.xml

```
<suite name="ParallelSuite" parallel="methods" thread-count="4">  
  <test name="ParallelTestExecution">  
    <classes>  
      <class name="parallel_testing.TestClass1"/>  
      <class name="parallel_testing.TestClass2"/>  
      <class name="parallel_testing.Para_testing_demo"/>  
    </classes>  
  </test>  
</suite>
```