

## MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Hiểu được các khái niệm liên quan Annotation
- ✓ Sử dụng Annotation linh hoạt trong lớp kiểm thử

## PHẦN I

### Bài 1 (3 điểm)

Sử dụng Annotation @BeforeTest, @AfterTest thiết lập thứ tự thực thi các test case trong lớp testNG

- Xây dựng lớp kiểm thử tên “firsttestngfile” kiểm thử tiêu đề trang web “http://http://lms.poly.edu.vn” và sử dụng trình duyệt Google Chrome.
- Lớp kiểm thử gồm 3 test case có thể viết theo thứ tự bất kỳ, tham khảo code:

```
package fpoly;
import org.openqa.selenium.*;
import org.openqa.selenium.chrome.ChromeDriver;
import org.testng.Assert;
import org.testng.annotations.*;
public class firsttestngfile {
    public String baseUrl = "http://http://lms.poly.edu.vn/";
    String driverPath = "C:\\\\chromedriver.exe";
    public WebDriver driver ;

    @BeforeTest
    public void launchBrowser() {
        System.out.println("launching chrome browser");
        System.setProperty("webdriver.chrome.driver", driverPath);
        driver = new FirefoxDriver();
        driver.get(baseUrl);
    }
}
```

```

@Test
public void verifyHomepageTitle() {
    String expectedTitle = "Hệ thống quản lý trị học tập Fpt PolyTechnic";
    String actualTitle = driver.getTitle();
    Assert.assertEquals(actualTitle, expectedTitle);
}

@AfterTest
public void terminateBrowser(){
    driver.close();
}
}

```

- Thực thi lớp kiểm thử bên trên và lưu ý thứ tự các test case thực thi
  - 1st - launchBrowser()
  - 2nd - verifyHomepageTitle()
  - 3rd - terminateBrowser()

Methods in chronological order		
<b>firsttestngpackage.FirstTestNGFile</b>		
launchBrowser		0 ms
verifyHomepageTitle		4641 ms
terminateBrowser		4664 ms

- Sinh viên thực hiện lại các thao tác trong bài 1 với trình duyệt Firefox và trang web: <http://ap.poly.edu.vn/index.php>

## Bài 2 (2 điểm)

Sử dụng Annotation @BeforeMethod, @AfterMethod thiết lập thứ tự thực thi các test case trong lớp testNG

- Thiết kế lớp kiểm thử tên "TestClass"
- Thiết kế test case "beforeMethod()" thực thi trước mỗi test case khác

```

@BeforeMethod
public void beforeMethod() {
    System.out.println("testClass: before method");
}

```

- Thiết kế test case "@AfterMethod" thực thi sau mỗi test case bất kỳ

```
@AfterMethod
public void afterMethod() {
    System.out.println("testClass: after method");
}
```

➤ Tham khảo code :

```
01 package com.fpoly.testng;
02
03 import org.testng.annotations.AfterClass;
04 import org.testng.annotations.AfterMethod;
05 import org.testng.annotations.AfterTest;
06 import org.testng.annotations.BeforeClass;
07 import org.testng.annotations.BeforeMethod;
08 import org.testng.annotations.BeforeTest;
09 import org.testng.annotations.Parameters;
10 import org.testng.annotations.Test;
11
12 public class TestClass {
13
14     @BeforeTest
15     public void beforeTest() {
16         System.out.println("testClass: before test");
17     }
18
19     @Test
20     public void unitLevel1() {
21         System.out.println("testClass: Unit level1 testing");
22     }
23
24     @Test
25     public void unitLevel2() {
26         System.out.println("testClass: Unit level2 testing");
27     }
28
29     @BeforeMethod
30     public void beforeMethod() {
31         System.out.println("testClass: before method");
32     }
```

```

33
34 @BeforeMethod
35 public static void staticBeforeMethod() {
36     System.out.println("testClass: static before method");
37 }
38
39 @Parameters({ "param" })
40 @BeforeMethod
41 public void beforeMethodWithParam(String p) {
42     System.out.println("testClass: before method with param " + p);
43 }
44
45 @AfterMethod
46 public void afterMethod() {
47     System.out.println("testClass: after method");
48 }
49
50 @BeforeClass
51 public void beforeClass() {
52     System.out.println("testClass: before class");
53 }
54
55 @AfterClass
56 public void afterClass() {
57     System.out.println("testClass: after class");
58 }
59
60 @AfterTest
61 public void afterTest() {
62     System.out.println("testClass: after test");
63 }
64 }

```

- Tạo file Testng.xml cấu hình truyền tham số cho “beforeMethod”

```

1 <?xml version="1.0" encoding="UTF-8"?>
2 <suite name="Feature" parallel="false">
3     <test name="UnitLevelTest">
4         <classes>
5             <parameter name="param" value="'I am beforeMethod'"/>
6             <class name="fpoly.TestClass"/>
7         </classes>
8     </test>
9 </suite>

```

- Thực thi lớp kiểm thử và xem kết quả

```

04 | testClass: before test
05 | testClass: before class
06 | testClass: before method with param 'I am beforeMethod'
07 | testClass: before method
08 | testClass: static before method
09 | testClass: Unit level1 testing
10 | testClass: after method
11 | testClass: before method with param 'I am beforeMethod'
12 | testClass: before method
13 | testClass: static before method
14 | testClass: Unit level2 testing
15 | testClass: after method
16 | testClass: after class
17 | testClass: after test
18 |
19 | =====
20 | Feature
21 | Total tests run: 2, Failures: 0, Skips: 0
22 | =====

```

### Bài 3 (3 điểm)

Thiết kế lớp kiểm thử sử dụng các trường hợp của @Parameters.

- Xây dựng lớp kiểm thử tên “ParametersTesting” có chứa 2 phương thức
  - OpenBrowser(): sử dụng annotation @Parameters(“BrowserName”) và nhận tham số đầu vào một chuỗi ký tự.
  - FillLoginForm(): sử dụng annotation @Parameters({ “UserName”, “Passcode” }) và nhận 2 tham số đầu vào là 2 chuỗi ký tự

```
package com.fpoly.testng;

import org.testng.annotations.Parameters;
import org.testng.annotations.Test;

public class ParametersTesting {

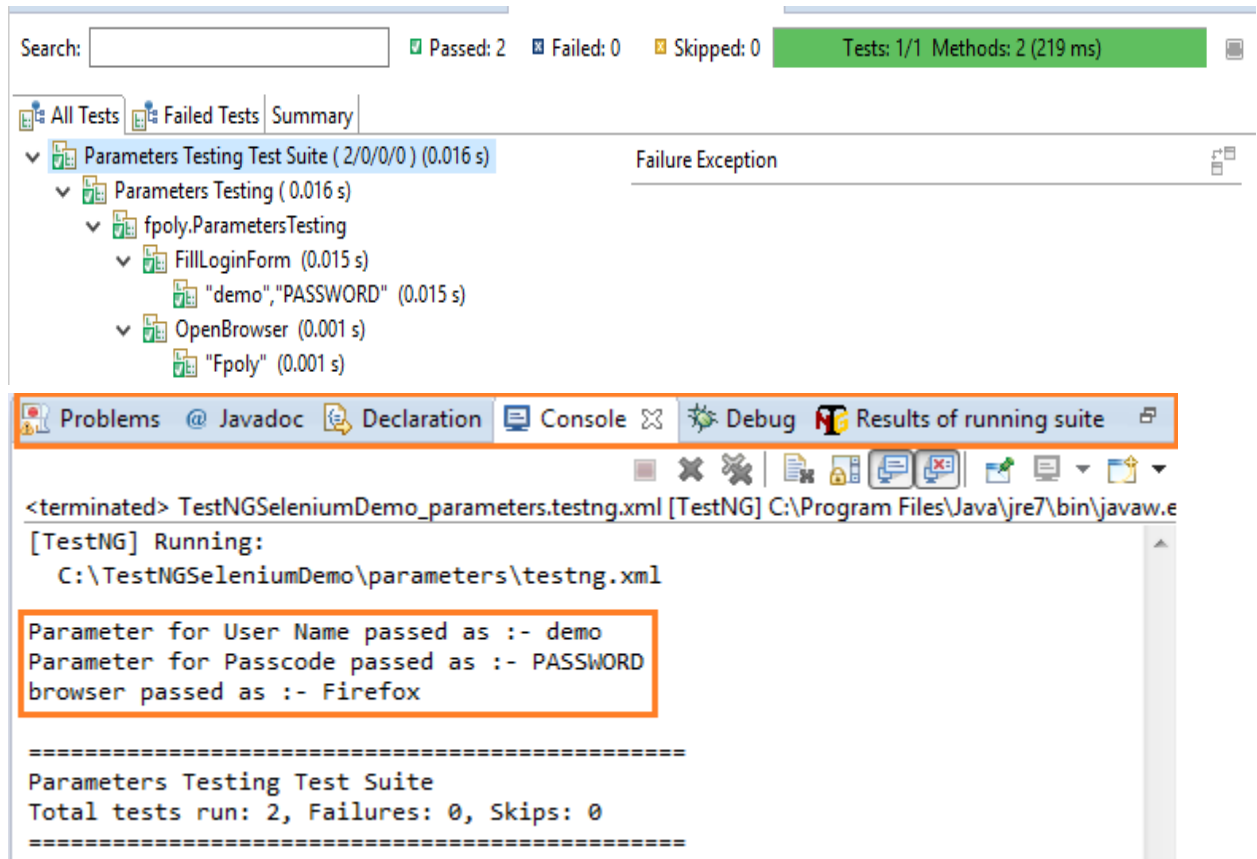
    @Parameters({ "BrowserName" })
    @Test
    public void OpenBrowser(String BrowserName) {
        System.out.println("browser passed as :- " + BrowserName);
    }

    @Parameters({ "UserName", "Passcode" })
    @Test
    public void FillLoginForm(String UserName, String Passcode) {
        System.out.println("Parameter for User Name passed as :- " +
UserName);
        System.out.println("Parameter for Passcode passed as :- " +
Passcode);
    }
}
```

- Tạo file.xml và cấu hình truyền giá trị cho 3 tham số: BrowserName, UserName và Passcode.

```
<?xml version="1.0" encoding="UTF-8"?>
<suite name="Parameters Testing Test Suite">
    <test name="Parameters Testing">
        <parameter name="BrowserName" value="Firefox"/>
        <parameter name="UserName" value="demo"/>
        <parameter name="Passcode" value="PASSWORD"/>
        <classes>
            <class name="com.fpoly.testng.ParametersTesting" />
        </classes>
    </test>
</suite>
```

➤ Kết quả thực thi lớp kiểm thử



Search:  Passed: 2 Failed: 0 Skipped: 0 Tests: 1/1 Methods: 2 (219 ms)

All Tests Failed Tests Summary

- Parameters Testing Test Suite ( 2/0/0/0 ) (0.016 s)
  - Parameters Testing ( 0.016 s)
    - fpoly.ParametersTesting
      - FillLoginForm (0.015 s)
        - "demo", "PASSWORD" (0.015 s)
      - OpenBrowser (0.001 s)
        - "Fpoly" (0.001 s)

Failure Exception

Problems Javadoc Declaration Console Debug Results of running suite

```
<terminated> TestNGSeleniumDemo_parameters.testng.xml [TestNG] C:\Program Files\Java\jre7\bin\javaw.e
[TestNG] Running:
C:\TestNGSeleniumDemo\parameters\testng.xml

Parameter for User Name passed as :- demo
Parameter for Passcode passed as :- PASSWORD
browser passed as :- Firefox

=====
Parameters Testing Test Suite
Total tests run: 2, Failures: 0, Skips: 0
=====
```

- Để thiết lập thứ tự thực thi các test case, dùng <priority>
- Tạo lớp kiểm thử theo trình tự: đăng ký thành công->đăng nhập- gửi mail

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

public class testNGPriorityExample {

    @Test
    public void registerAccount()
    {

        System.out.println("First register your account");

    }
    @Test(priority=2)
    public void sendEmail()
    {

        System.out.println("Send email after login");

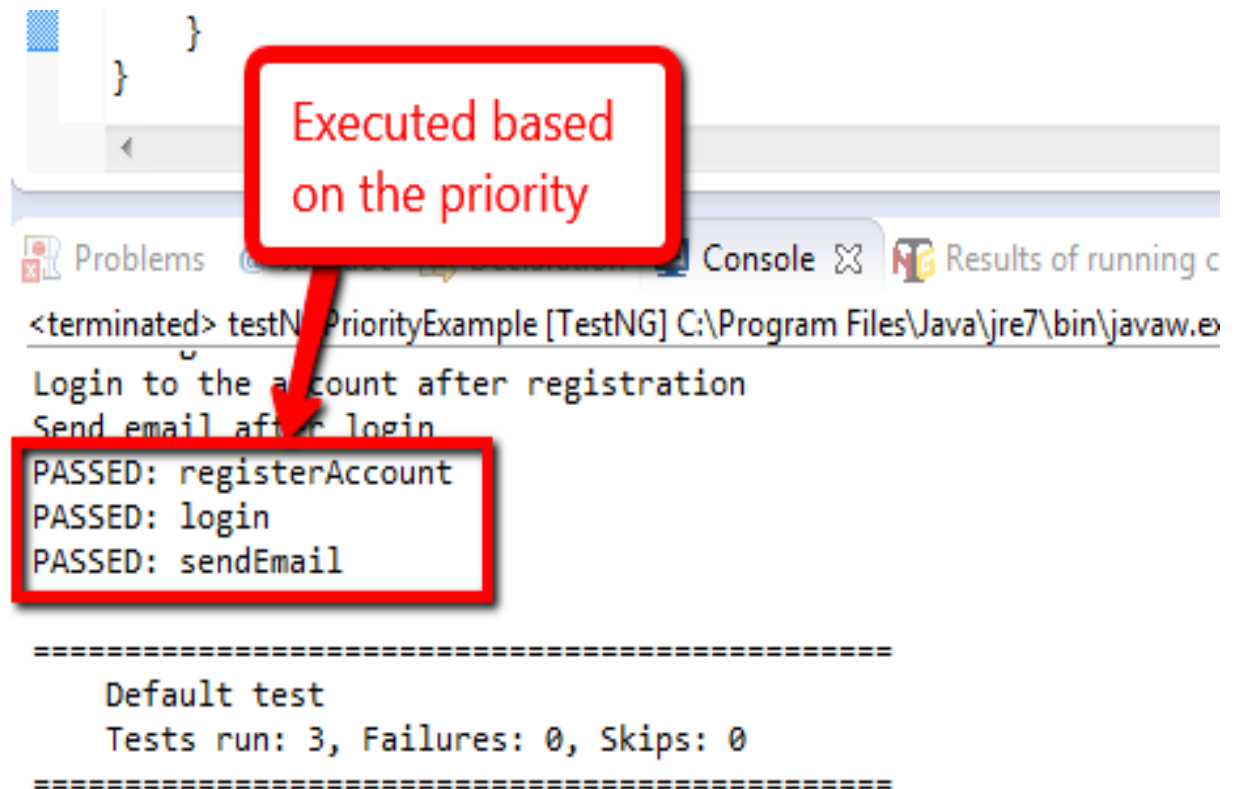
    }
    @Test(priority=1)
    public void login()
    {

        System.out.println("Login to the account after registration");

    }

}
```

➤ Kết quả thực thi:



```
<terminated> testNGPriorityExample [TestNG] C:\Program Files\Java\jre7\bin\javaw.exe
Login to the account after registration
Send email after login
PASSED: registerAccount
PASSED: login
PASSED: sendEmail

=====
Default test
Tests run: 3, Failures: 0, Skips: 0
=====
```



**Bài 4 (2 điểm)**

Giảng viên cho thêm