

MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng

- ✓ Hiểu được các khái niệm liên quan đến kiểm thử Unit Test
- ✓ Cấu hình và sử dụng Junit với Eclipse
- ✓ Sử dụng Maven project với Junit

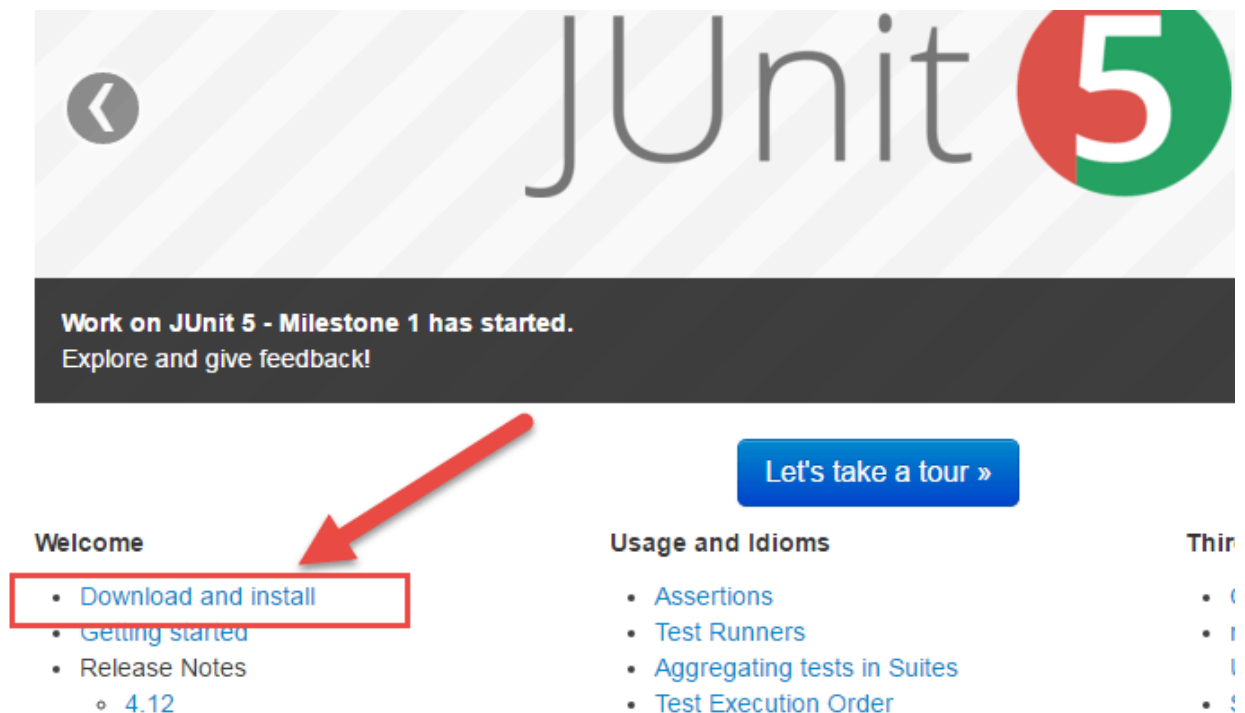
PHẦN I

Bài 1 (2 điểm)

Chuẩn bị môi trường chạy UnitTest với Junit. Cài đặt JDK và Junit.

1. Tải Junit

- Download Junit: sinh viên vào “<http://junit.org/junit4/>” chọn Download and Install



- Chọn junit.jar

junit-team / junit4

<> Code ! Issues 131 Pull requests 42 Wiki Pulse

Download and Install

Shaun Reich edited this page on Nov 29, 2015 · 20 revisions

To download and install JUnit you currently have the following options

Plain-old JAR

Download the following JARs and put them on your test classpath:

- junit.jar
- hamcrest-core.jar

- Tại màn hình “the central repository” có nhiều version Junit, chọn version 4.12

The Central Repository

SEARCH | ADVANCED SEARCH | BROWSE | QUICK STATS

g:"junit" AND a:"junit"

SEARCH

New: About Central Advanced Search API Guide Help

Search Results < 1 2 ≥ displaying 1 to 20 of 24

| Groupid | ArtifactId | Version | Updated | Download |
|---------|------------|-------------|-------------|---|
| junit | junit | 4.12 | 04-Dec-2014 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.12-beta-3 | 09-Nov-2014 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.12-beta-2 | 25-Sep-2014 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.12-beta-1 | 28-Jul-2014 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.11 | 15-Nov-2012 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.11-beta-1 | 16-Oct-2012 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.10 | 30-Sep-2011 | pom jar javadoc.jar sources.jar |
| junit | junit | 4.9 | 22-Aug-2011 | pom jar javadoc.jar sources.jar |


- Quay lại màn hình download để tiếp tục chọn file “hamcrest-core.jar”

Plain-old JAR

Download the following JARs and put them on your test classpath:

- `junit.jar`
- `hamcrest-core.jar`

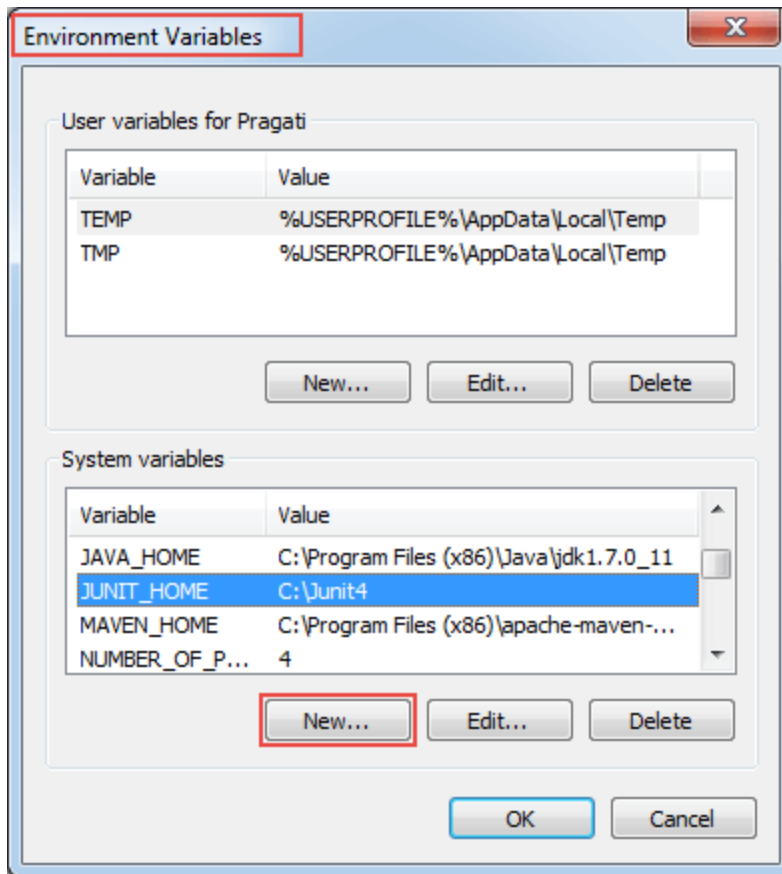
- Tiến hành download file jar
 Search Results



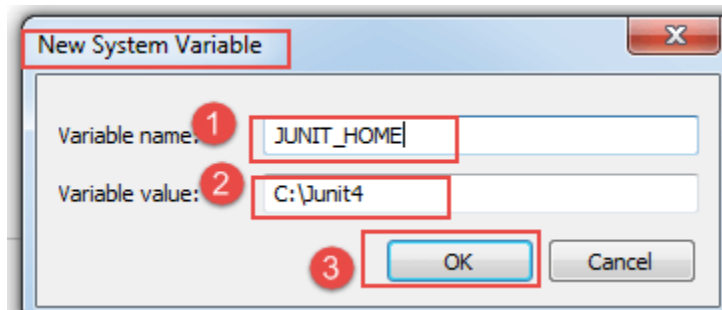
| GroupId | ArtifactId | Version | Updated | Download |
|------------------------------|-------------------------------|-------------------------|-------------|---|
| org.hamcrest | hamcrest-core | 1.3 | 10-Jul-2012 | pom jar javadoc.jar sources.jar |
| org.hamcrest | hamcrest-core | 1.3.RC2 | 22-Dec-2010 | pom jar javadoc.jar sources.jar |
| org.hamcrest | hamcrest-core | 1.3.1 | 22-Dec-2010 | pom jar javadoc.jar sources.jar |

2. Thiết lập môi trường chạy Junit

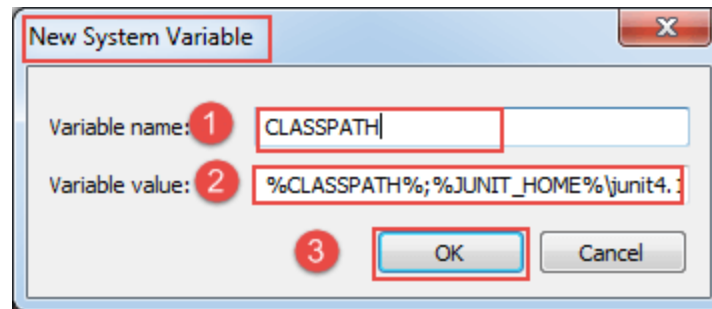
- Thiết lập biến môi trường `JUNIT_HOME` tham chiếu đến thư mục đang chứa file “junit.jar” đã download ở bước trên.
- Chuột phải my computer-> advanced system settings->Environment Variables



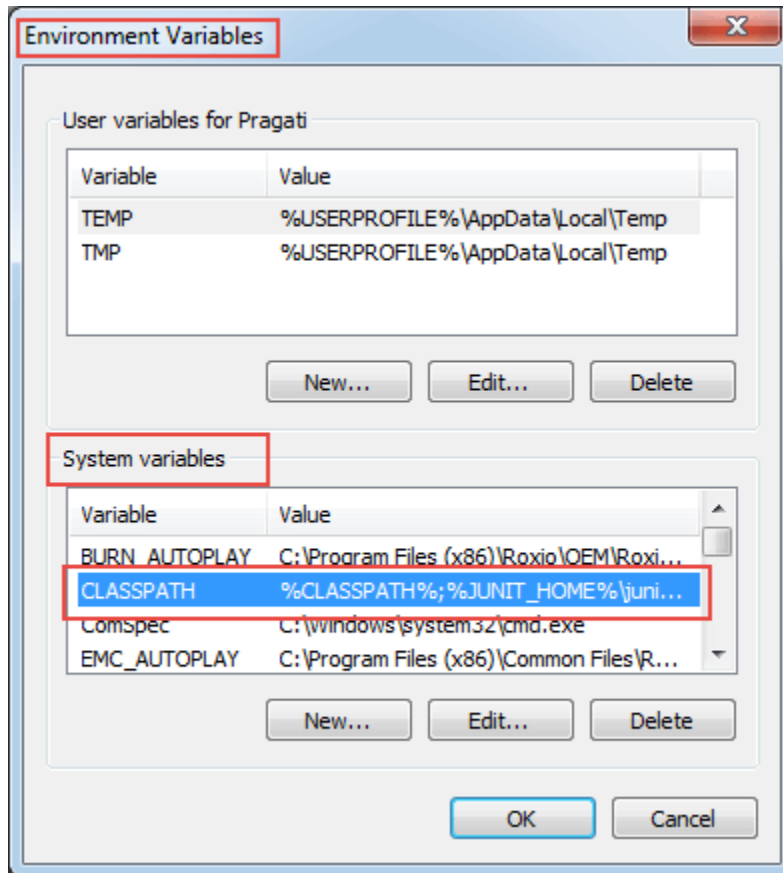
- Sau khi chọn nút “New” thì xuất hiện cửa sổ “New System Variable”, điền thông tin (lưu ý thay đổi đường dẫn lưu file junit.jar):
 - variable name : JUNIT_HOME
 - variable value: C:\JUnit



- Chọn nút “Ok” hoàn thành tạo biến môi trường kết nối với junit.jar
- Tương tự tạo biến môi trường CLASSPATH với thông tin
 - Variable Name: CLASSPATH
 - Variable Value: %CLASSPATH%;%JUNIT_HOME%\JUnit4.12.jar;.
 - Chọn nút OK



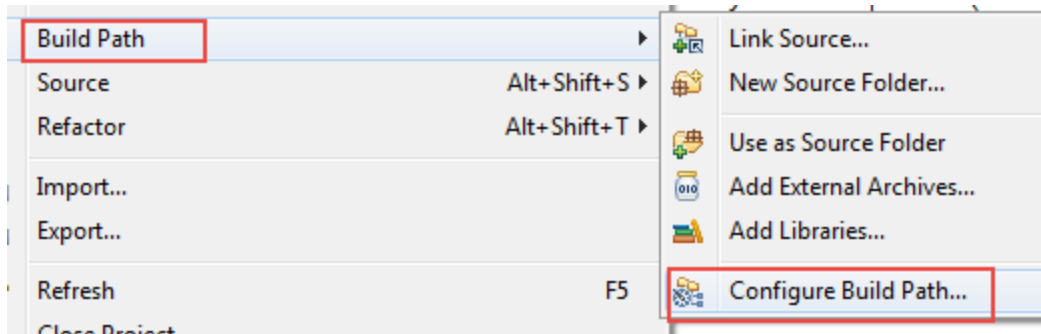
- Kiểm tra lại các công việc vừa làm xong:



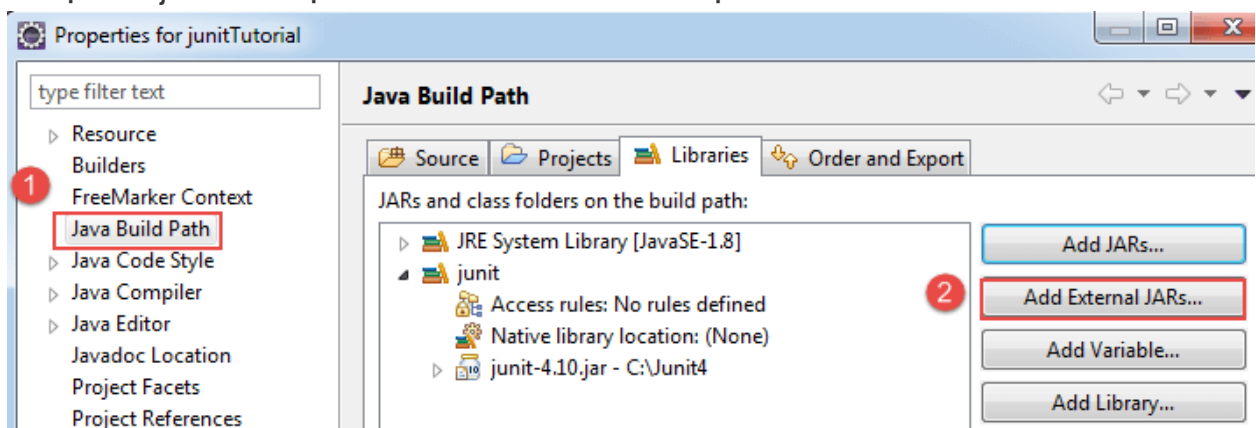
Bài 2 (3 điểm)

Cài đặt file junit.jar trên Eclipse và chạy ứng dụng unit test

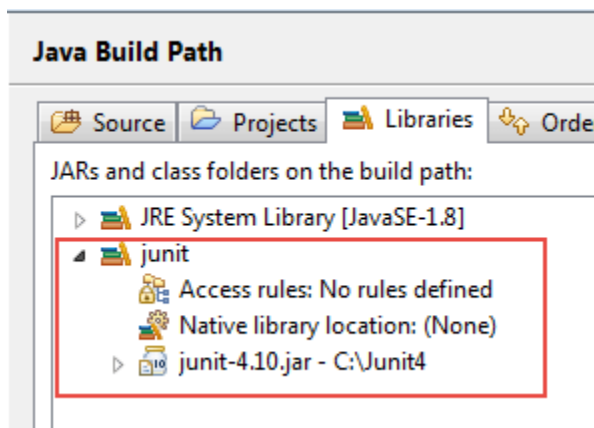
- Sinh viên tạo mới ứng dụng project java, đặt tên jnuit. Phải chuột vào project, chọn "build path" và chọn "Configure build path"



- Cửa sổ properties xuất hiện, chọn “java build path”-> Add External JARs->chọn file junit mà bạn đã download ở bài1->chọn OK



- Có thể vào Libraries để kiểm tra xem junit.jar đã thêm thành công hay không



- Thiết kế chương trình chạy kiểm thử unit test với Junit
 - Tạo class TestJUnit.java bên trong project junit, sử dụng “assert “ đơn giản so sánh 2 chuỗi ký tự

```
package fpoly.junit;

import org.junit.Test;
import static org.junit.Assert.assertEquals;
public class TestJUnit {
    @Test
    public void testSetup() {
        String str= "I am done with Junit setup";
        assertEquals("I am done with Junit setup",str);
    }
}
```

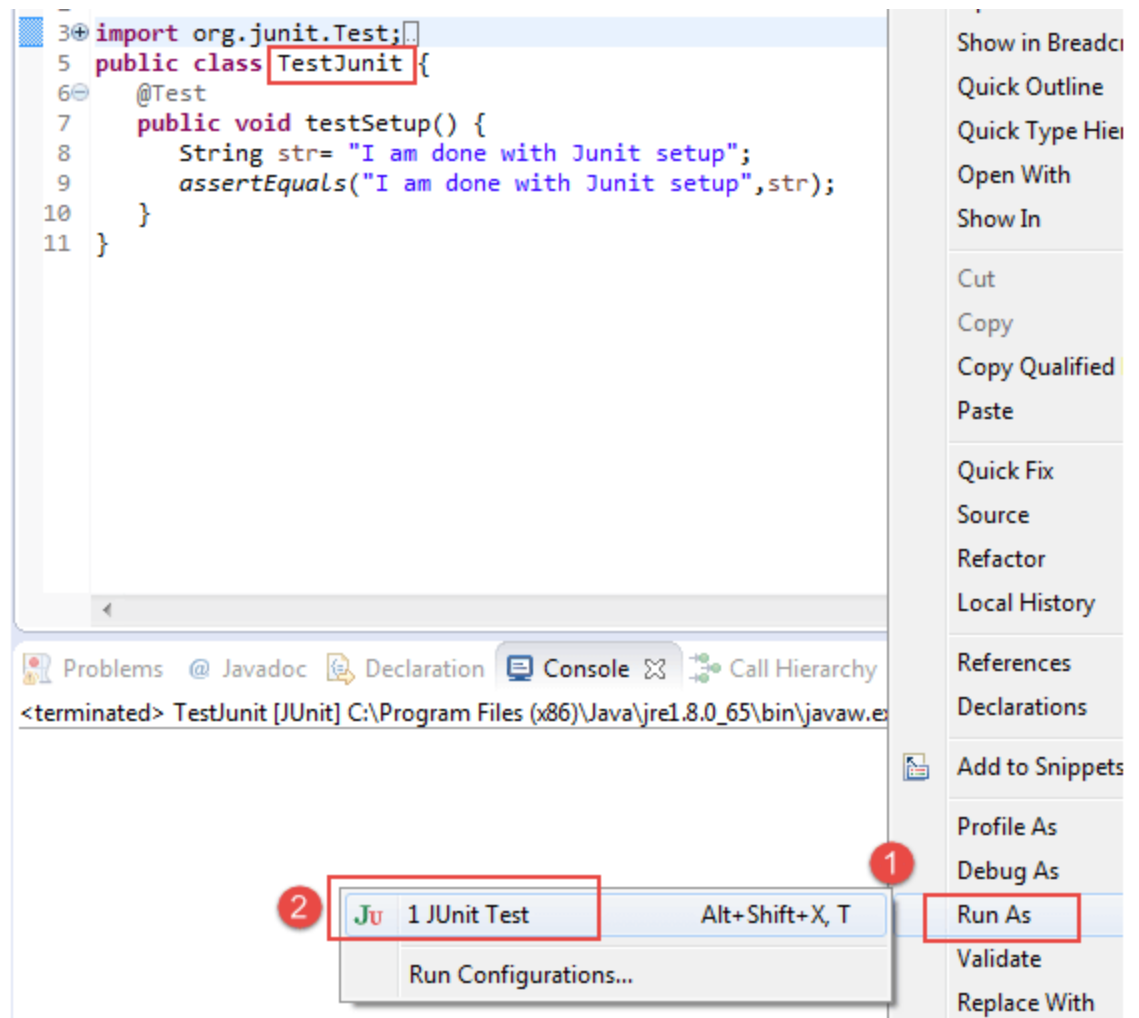
- Tạo class Test Runner thực thi lớp TestJUnit bên trên

```
package fpoly.junit;

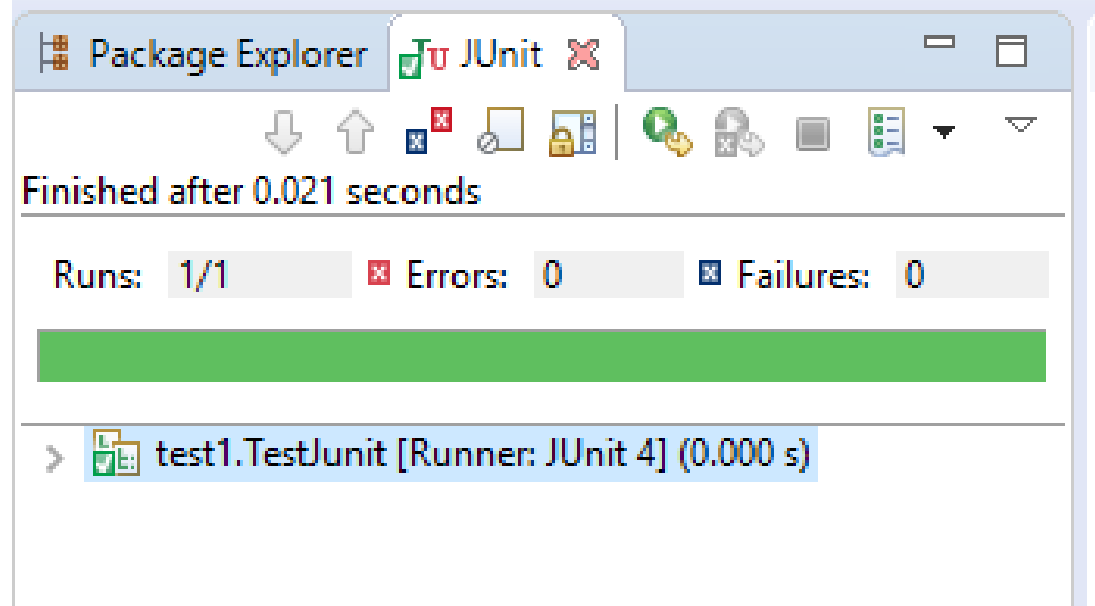
import org.junit.runner.JUnitCore;
import org.junit.runner.Result;
import org.junit.runner.notification.Failure;

public class TestRunner {
    public static void main(String[] args) {
        Result result = JUnitCore.runClasses(TestJUnit.class);
        for (Failure failure : result.getFailures()) {
            System.out.println(failure.toString());
        }
        System.out.println("Result==" + result.wasSuccessful());
    }
}
```

- Chạy và kiểm tra unit test, phải chuột lên TestJUnit-> Run As-> JUnit Test



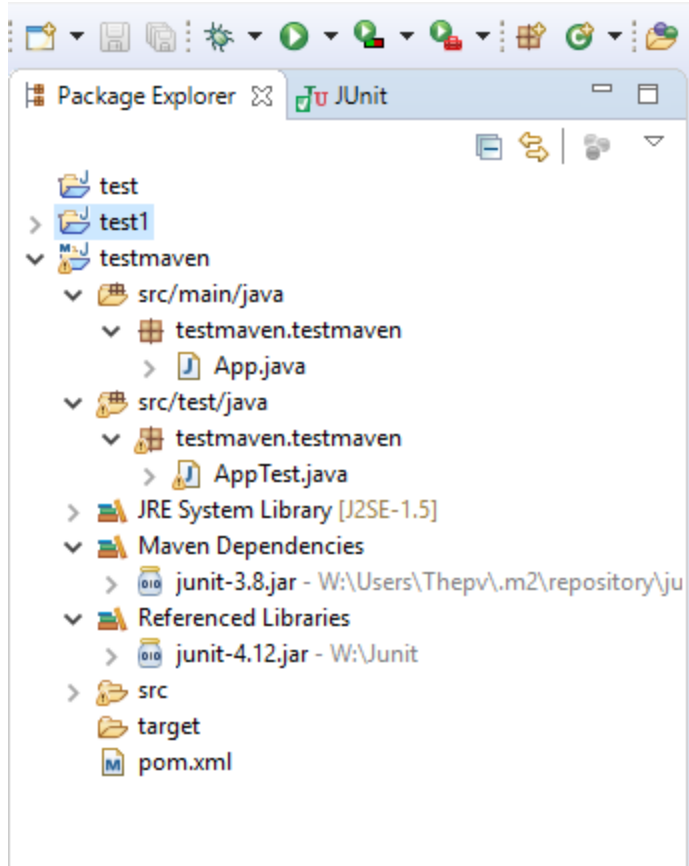
- Quan sát cửa sổ Junit thấy test pass vì 2 chuỗi cung cấp bên trên là giống nhau



Bài 3 (3 điểm)

Kết hợp Junit với Maven project.

- Thật tiện lợi khi bạn sử dụng Maven project, khi kết hợp với Junit nó sẽ tự tạo 2 thư mục `src/main/java` và folder để viết test `src/test/java`
- Tạo project maven, đặt tên “testmaven”



- Khai báo sử dụng thư viện Junit bên trong “pom.xml”

```
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.12</version>
    <scope>test</scope>
  </dependency>
</dependencies>
```

- Bên trong class App.java khai báo phương thức `isEventNumber(int input)` kiểm tra một số có là số chẵn hay không

```

7  public class App
8  {
9  public boolean isEventNumber(int input) {
10
11      if(input%2==0)
12          return true;
13      else
14          return false;
15  }
16  public static void main( String[] args )
17  {
18      System.out.println( "Hello World!" );
19  }
20 }

```

- Sử dụng JUnit để kiểm tra phương thức trên với các tham số đầu vào khác nhau, viết code trong AppTest.java

```

public class AppTest
    extends TestCase
{
    /**
     * Create the test case
     *
     * @param testName name of the test case
     */
    public AppTest( String testName )
    {
        super( testName );
    }

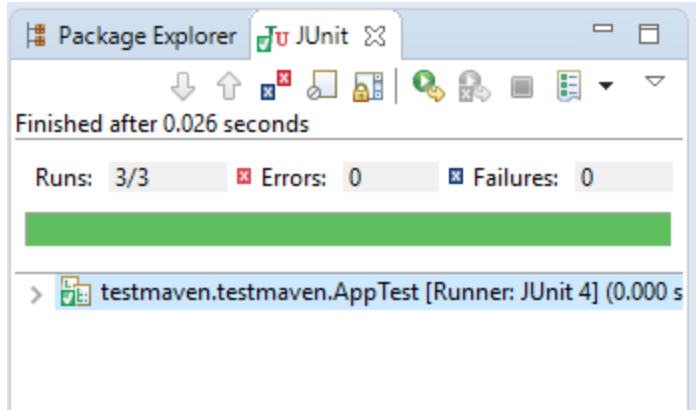
    public void testApp()
    {
        assertTrue( true );
    }

    @Test
    public void testIsEventNumber2() {
        App demo1 = new App();
        boolean result = demo1.isEventNumber(2);
        assertTrue(result);
    }

    @Test
    public void testIsEventNumber4() {
        App demo1 = new App();
        boolean result = demo1.isEventNumber(4);
        assertTrue(result);
    }
}

```

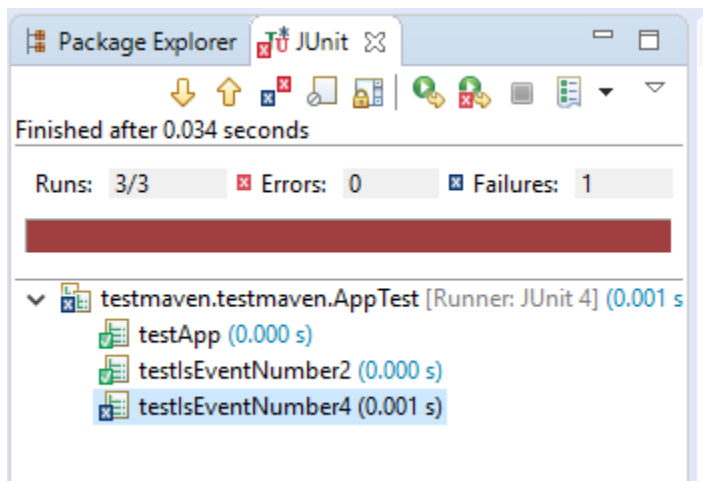
- Sinh viên lưu ý trong đoạn code trên nhập tham số đầu vào 2 và 4 nên test pass, chọn phải chuột AppTest->Run As-> 1 Junit Test



- Thay đổi tham số đầu vào 4 thành 3

```
@Test
public void testIsEventNumber4() {
    App demo1 = new App();
    boolean result = demo1.isEventNumber(3);
    assertTrue(result);
}
```

- Chạy lại junit test thì sẽ có 1 pass cho tham số 2 và một fail cho tham số 3



Bài 4 (2 điểm)

Giảng viên cho thêm