

Demo Lab1

Kiem thu nang cao

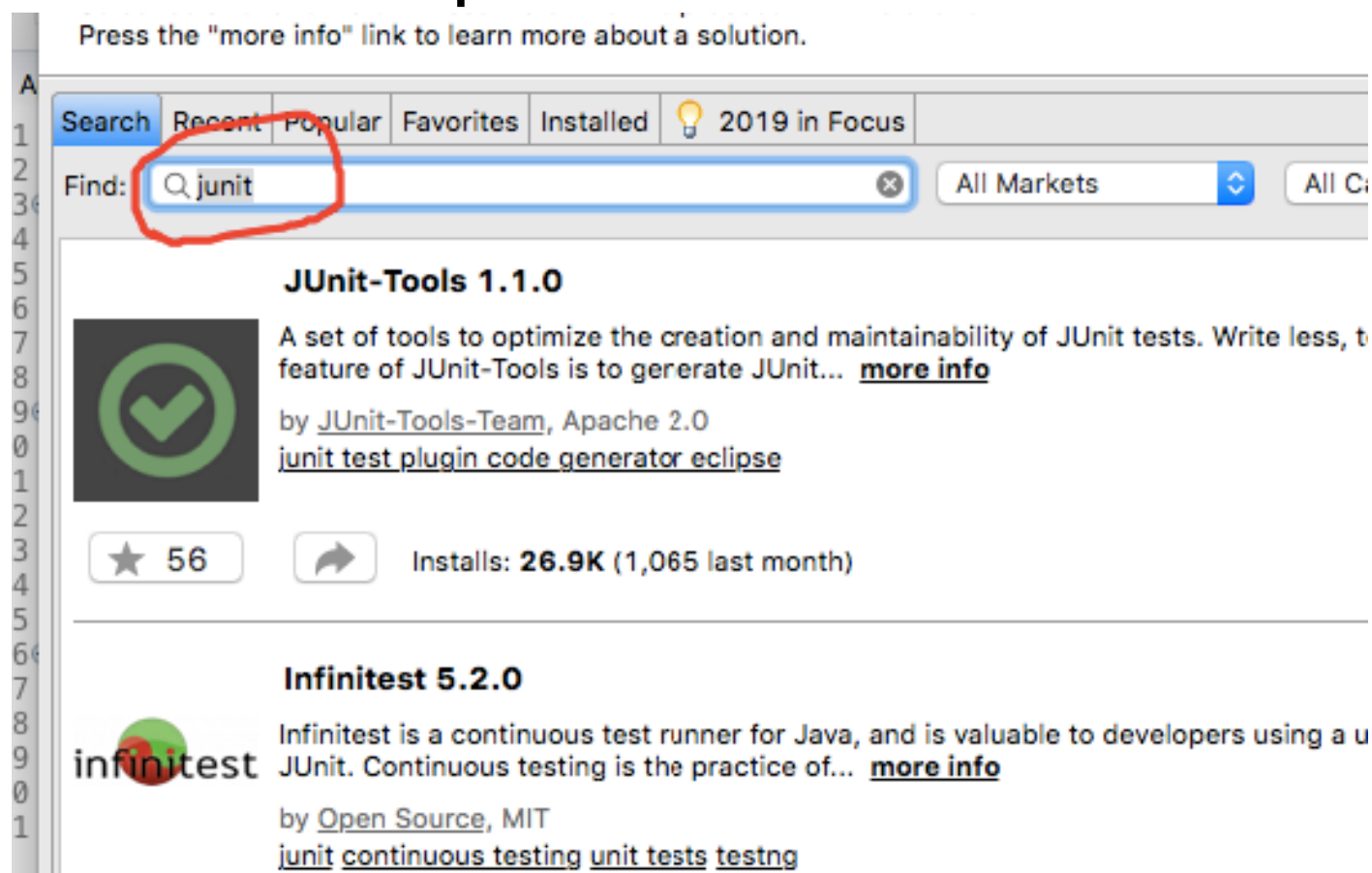
Gv HungNQ

Bài 1: Cài đặt công cụ

- Cài đặt Eclipse
- Download thư viện junit test
- <https://github.com/hungnttg/KiemThuTuDong>
- <http://junit.org/junit4/>

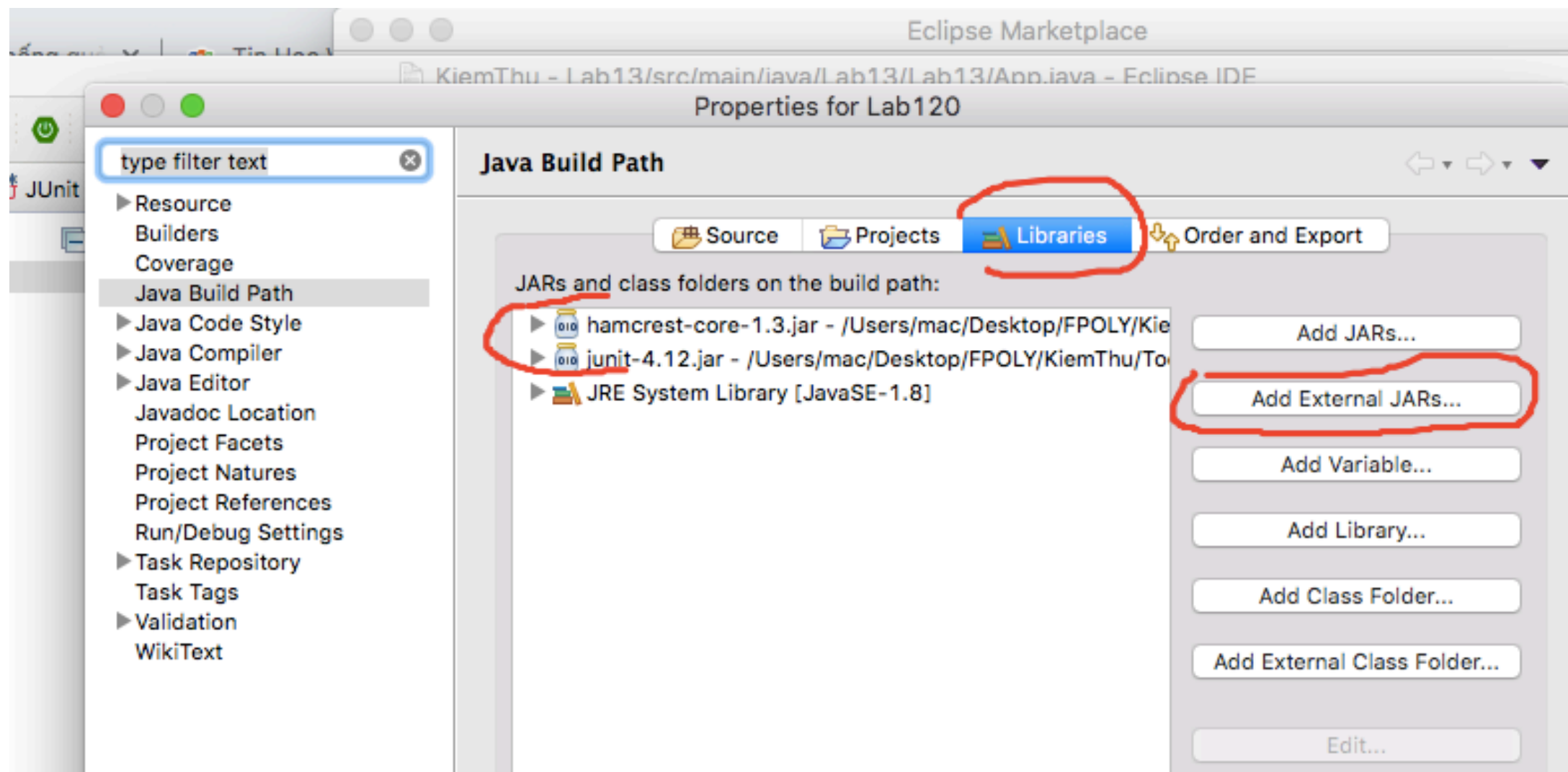
Bài 1: Cài đặt công cụ

- Vào Marketplace cài đặt Junit tool



Bài 2: Tạo ứng dụng Junit test đầu tiên

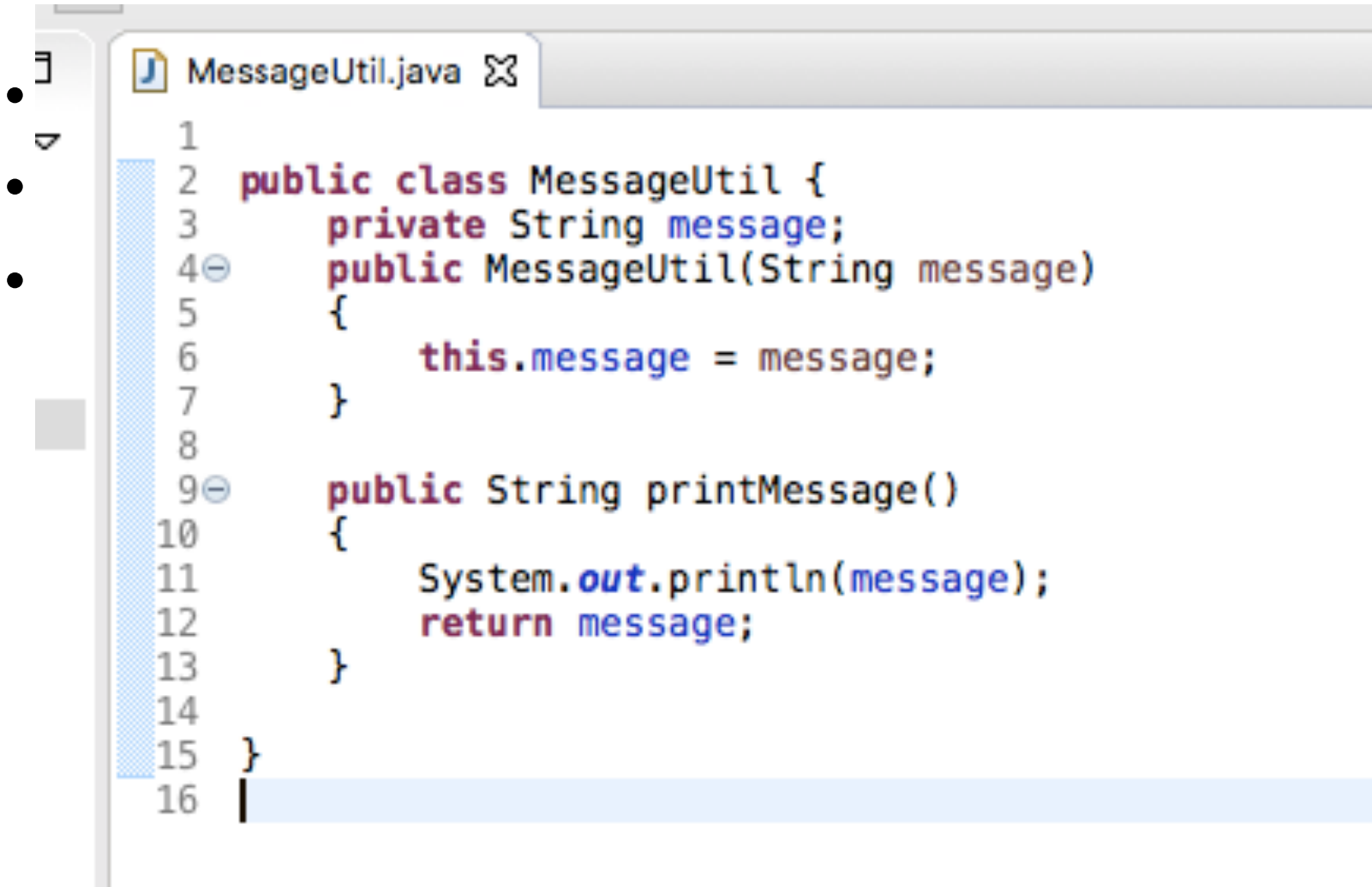
- 1. Tạo project
- 2. Nhấn phải chuột - chọn Build Path



Bài 2: Tạo ứng dụng Junit test đầu tiên

- 1. Tạo project
- 2. Nhấn phải chuột - chọn Build Path
- 3. Tạo class cần test MessageUtil.java

Bài 2: Tạo ứng dụng Junit test đầu tiên



The screenshot shows a Java IDE with a single file named `MessageUtil.java` open. The code defines a `MessageUtil` class with a private `String` attribute `message`, a constructor, and a `printMessage` method. The code is as follows:

```
1
2 public class MessageUtil {
3     private String message;
4     public MessageUtil(String message)
5     {
6         this.message = message;
7     }
8
9     public String printMessage()
10    {
11        System.out.println(message);
12        return message;
13    }
14
15 }
16
```

Bài 2: Tạo ứng dụng Junit test đầu tiên

- 1. Tạo project
- 2. Nhấn phải chuột - chọn Build Path
- 3. Tạo class cần test MessageUtil.java
- 4. Tạo class test TestJunit.java

Bài 2: Tạo ứng dụng Junit test đầu tiên



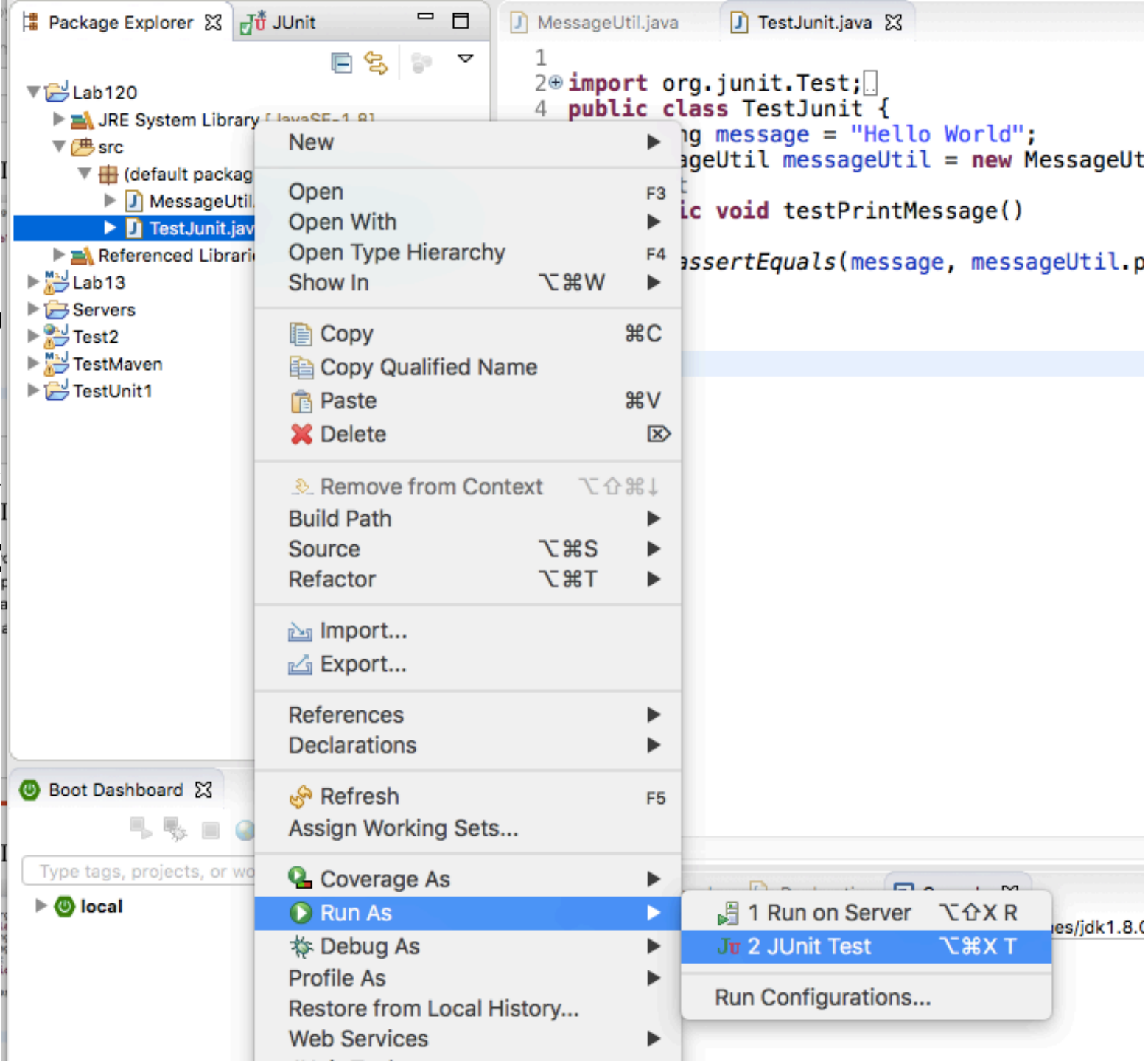
```
1
2+ import org.junit.Test;
4 public class TestJunit {
5     String message = "Hello World";
6     MessageUtil messageUtil = new MessageUtil(message);
7-     @Test
8     public void testPrintMessage()
9     {
10         assertEquals(message, messageUtil.printMessage());
11     }
12
13 }
14
```

Bài 2: Tạo ứng dụng Junit test đầu tiên

- 1. Tạo project
- 2. Nhấn phải chuột - chọn Build Path
- 3. Tạo class cần test MessageUtil.java
- 4. Tạo class test TestJunit.java
- 5. Nhấn phải chuột vào TestJunit-> Run as -> Jnunit

Bài 2:

- 1. Tạo p
- 2. Nhấn
- 3. Tạo c
- 4. Tạo c

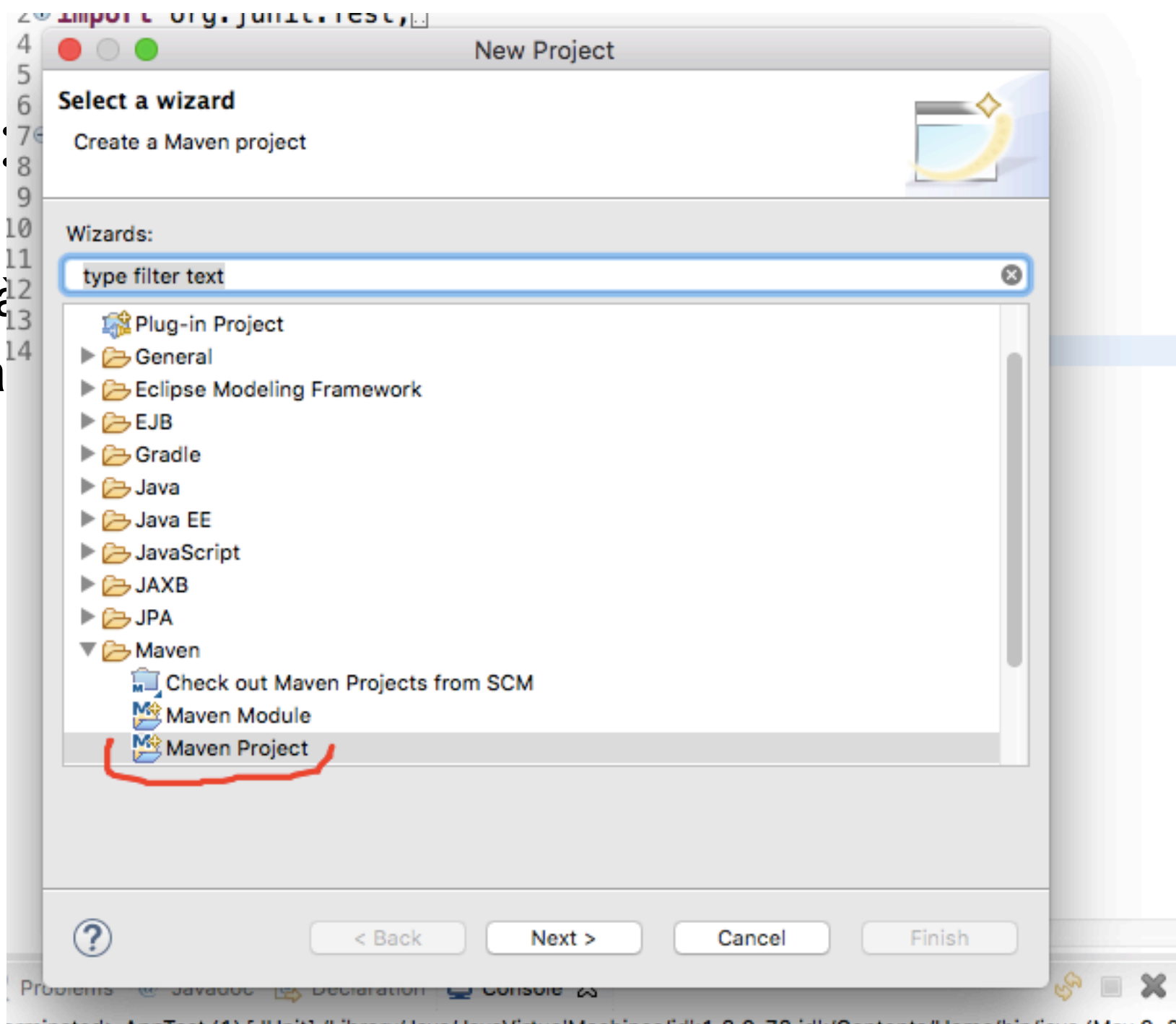


Bài 3: Test sử dụng Maven Project

- Test hàm kiểm tra 1 số xem có phải số chẵn không?
- B1. Tạo maven project

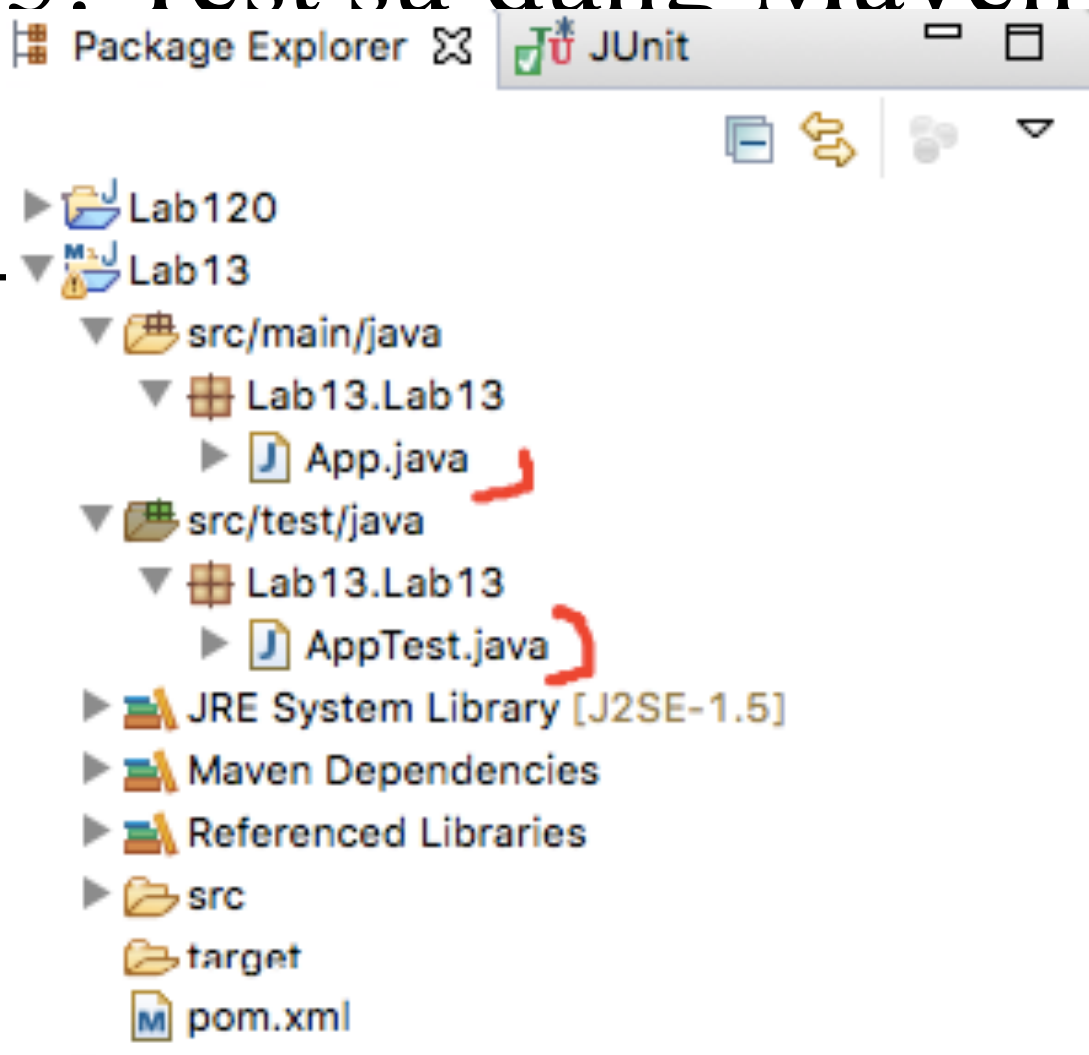
Bài 3:

- Test h
- B1. T



Bài 3: Test sử dụng Maven Project

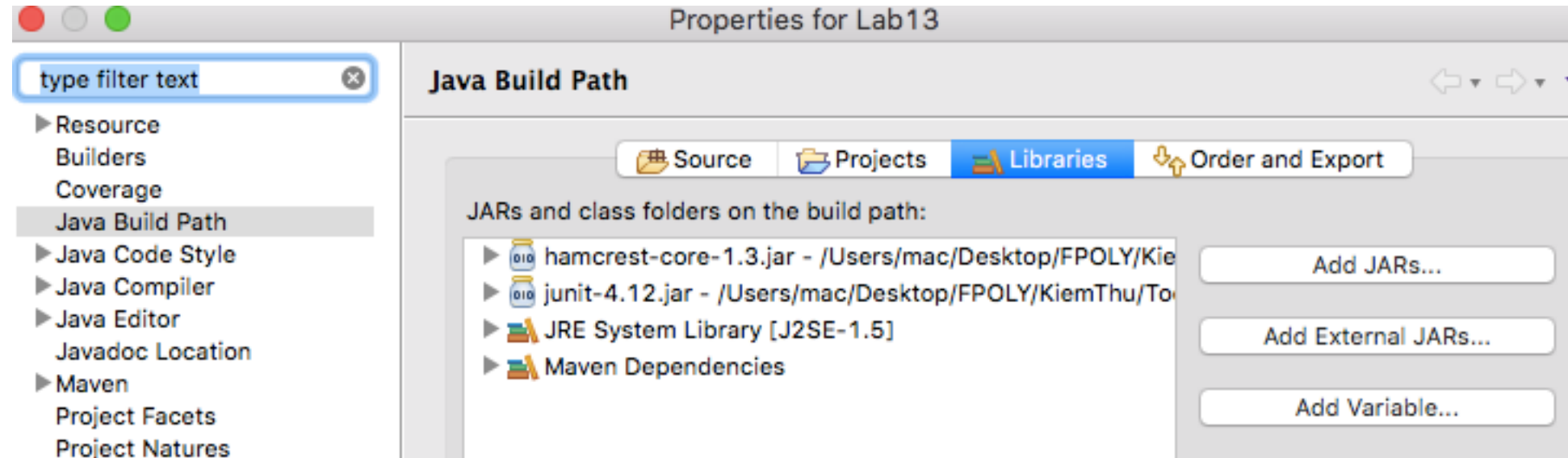
- Test
- B1.



ố chẵn không?

Bài 3: Test sử dụng Maven Project

- Test hàm kiểm tra 1 số xem có phải số chẵn không?
- B1. Tạo maven project
- B2. Build Path -> Library-> add external Jar -> Tìm đến file junit

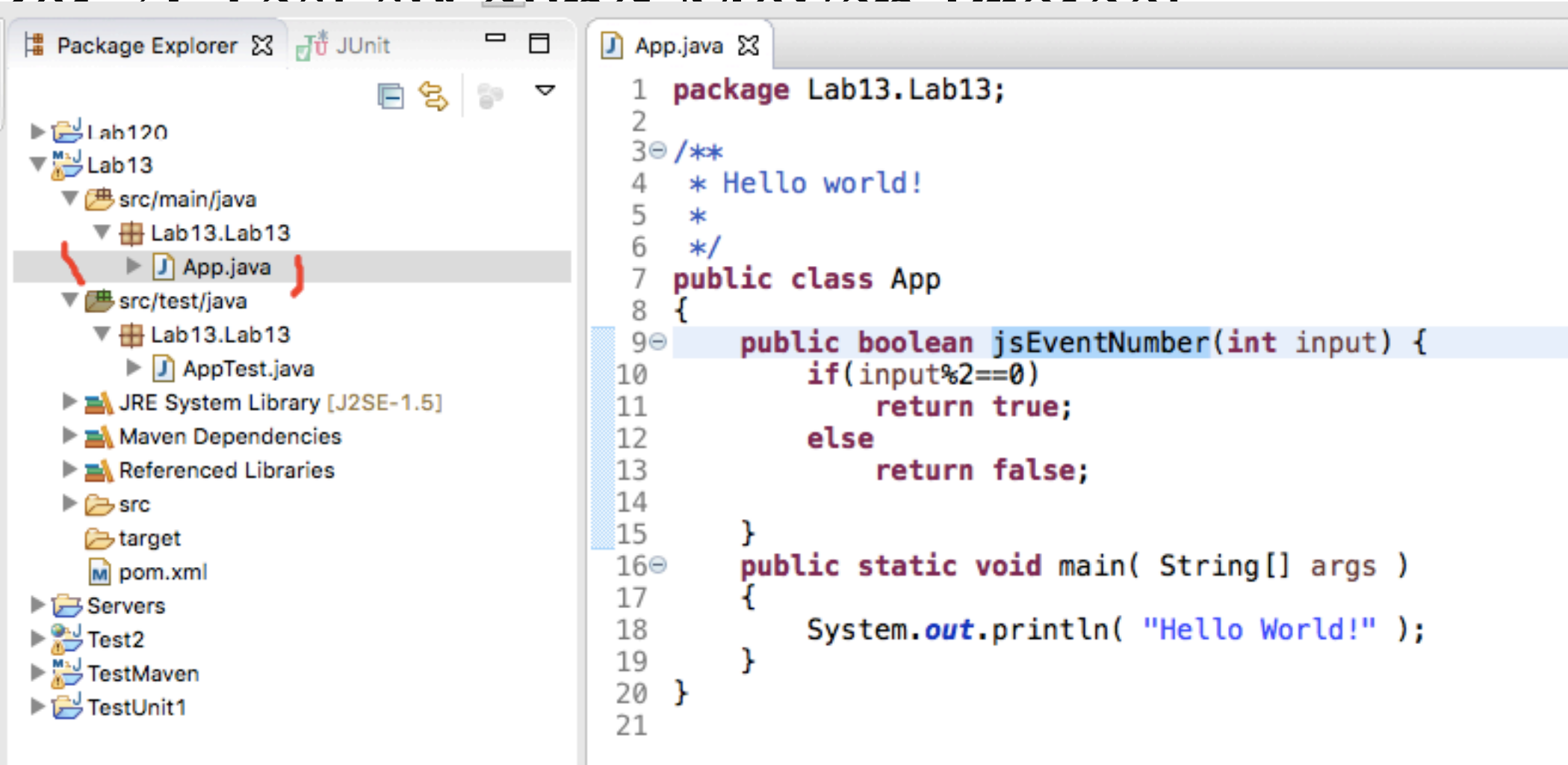


Bài 3: Test sử dụng Maven Project

- Test hàm kiểm tra 1 số xem có phải số chẵn không?
- B1. Tạo maven project
- B2. Build Path -> Library-> add external Jar -> Tìm đến file junit
- B3. Viết hàm jsEventNumber trong file App.java

Part 2: Test-driven Maven Project

-
-
-
-
-



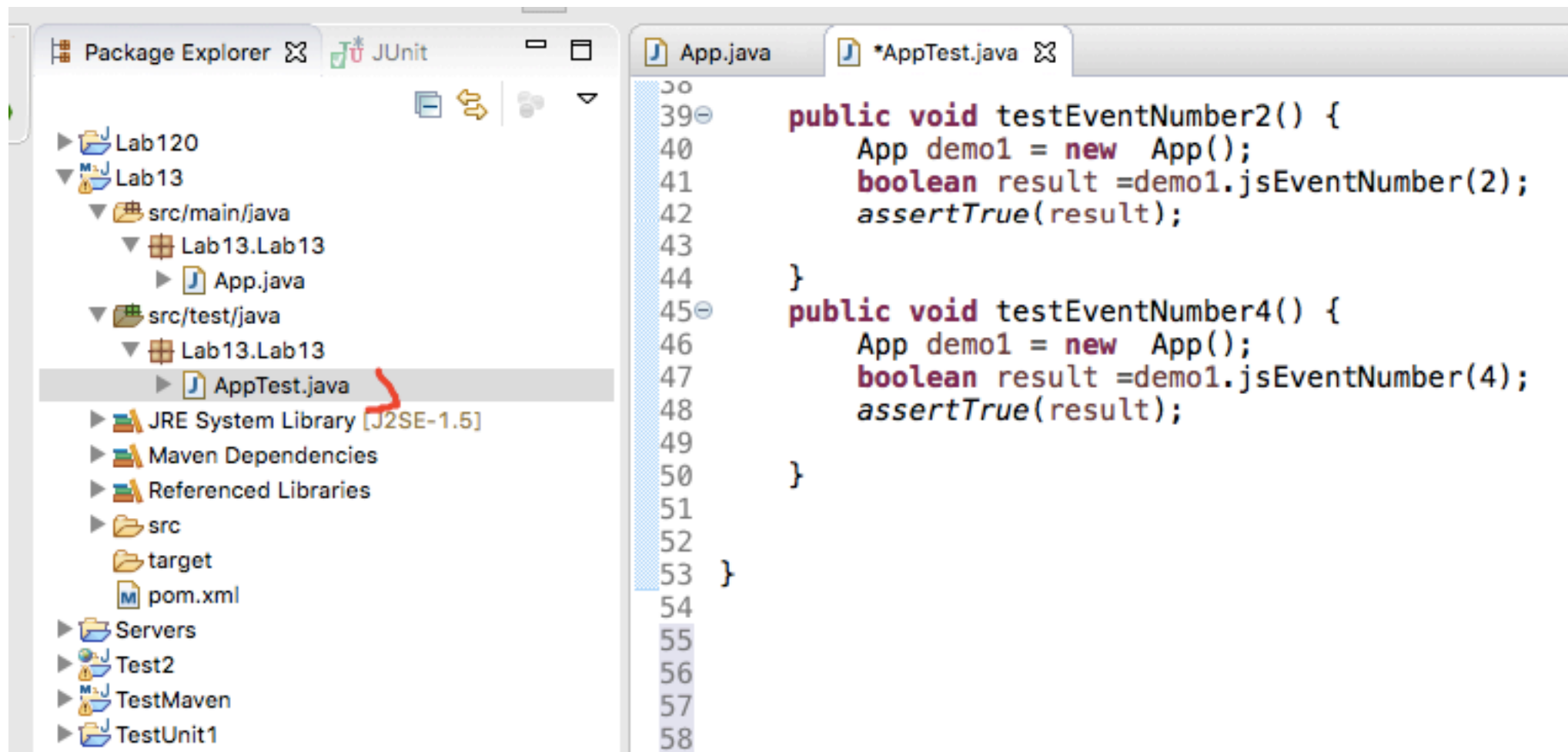
The screenshot displays an IDE interface with two main panels. The left panel, titled 'Package Explorer', shows a project structure for 'Lab13'. It includes a 'src/main/java' directory containing 'Lab13.Lab13' and 'App.java' (highlighted with a red bracket). Below this is a 'src/test/java' directory containing 'Lab13.Lab13' and 'AppTest.java'. Other items in the project include 'JRE System Library [J2SE-1.5]', 'Maven Dependencies', 'Referenced Libraries', 'src', 'target', 'pom.xml', 'Servers', 'Test2', 'TestMaven', and 'TestUnit1'. The right panel, titled 'App.java', shows the source code of the 'App' class. The code is as follows:

```
1 package Lab13.Lab13;
2
3 /**
4  * Hello world!
5  *
6  */
7 public class App
8 {
9     public boolean jsEventNumber(int input) {
10         if(input%2==0)
11             return true;
12         else
13             return false;
14     }
15
16     public static void main( String[] args )
17     {
18         System.out.println( "Hello World!" );
19     }
20 }
21
```

Bài 3: Test sử dụng Maven Project

- Test hàm kiểm tra 1 số xem có phải số chẵn không?
- B1. Tạo maven project
- B2. Build Path -> Library-> add external Jar -> Tìm đến file junit
- B3. Viết hàm jsEventNumber trong file App.java
- B4. Viết hàm kiểm thử trong AppTest.java

Bài 3: Test sử dụng Maven Project

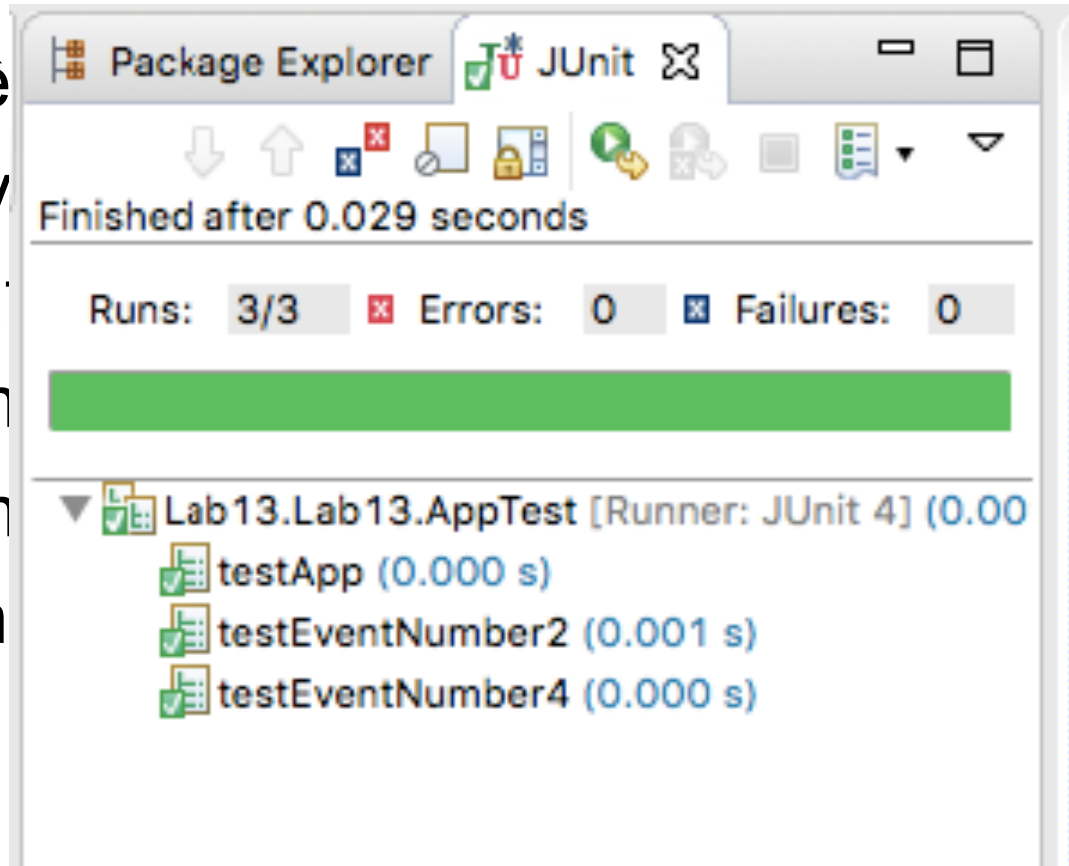


Bài 3: Test sử dụng Maven Project

- Test hàm kiểm tra 1 số xem có phải số chẵn không?
- B1. Tạo maven project
- B2. Build Path -> Library-> add external Jar -> Tìm đến file junit
- B3. Viết hàm jsEventNumber trong file App.java
- B4. Viết hàm kiểm thử trong AppTest.java
- B5. Nhấn phải chuột vào Apptest.java -> Run as -> Junit

Bài 3: Test sử dụng Maven Project

- Test hàm kiểm tra
- B1. Tạo maven project
- B2. Build Project
- B3. Viết hàm kiểm tra
- B4. Viết hàm kiểm tra
- B5. Nhấn phím



không?

> Tìm đến file junit
a

as -> Junit