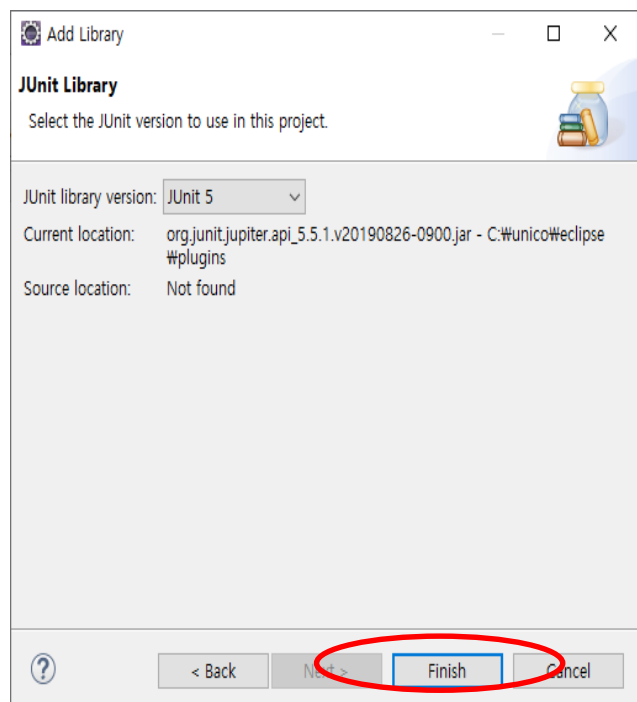
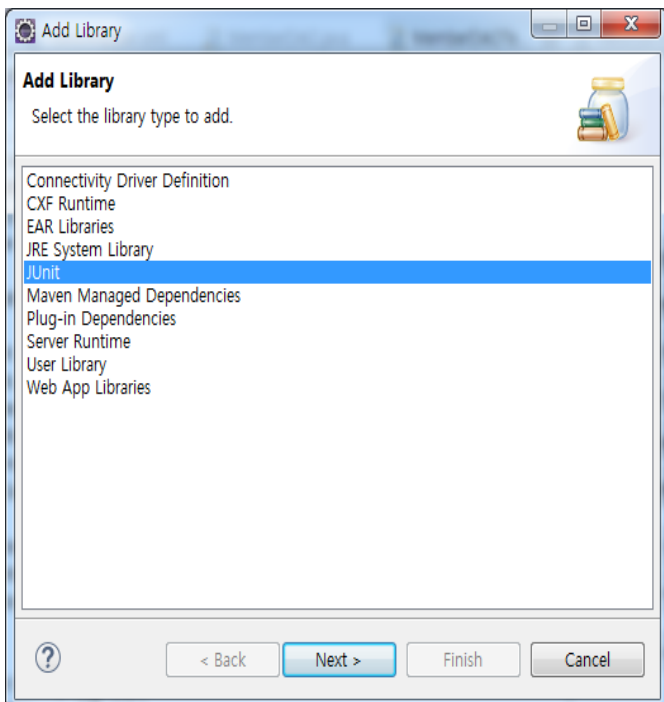
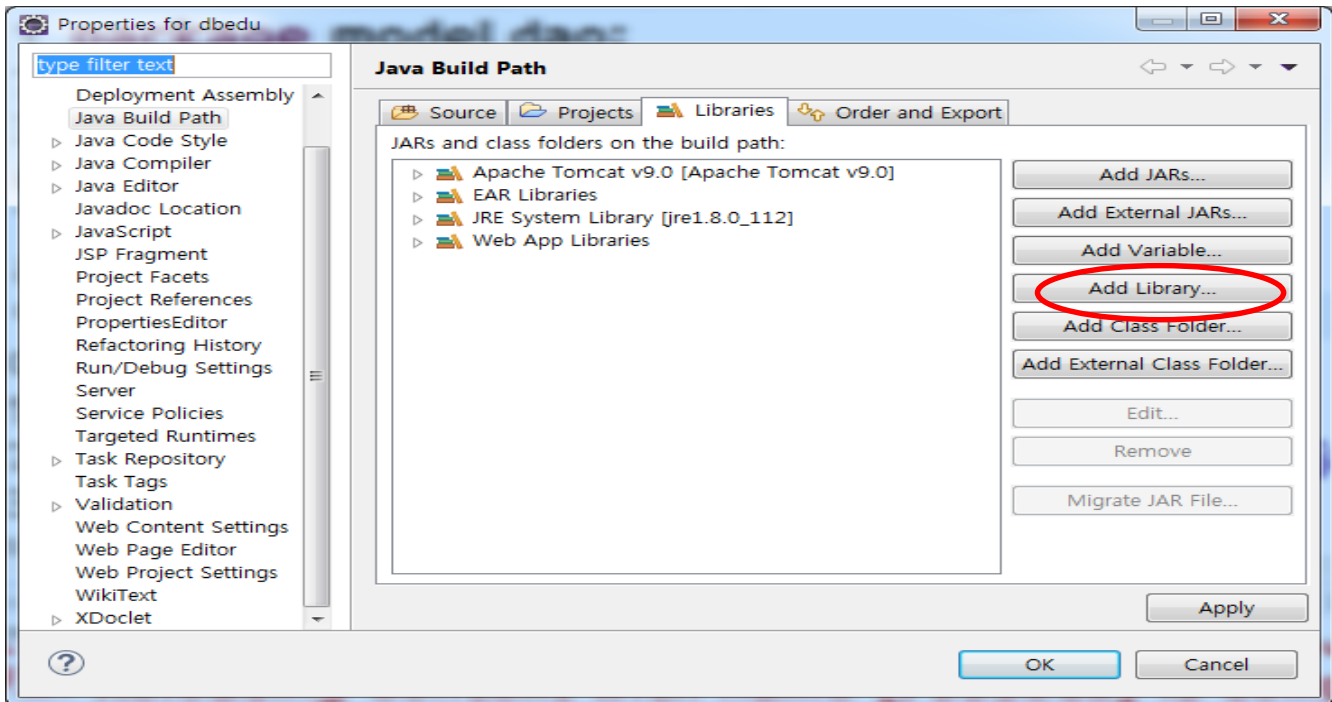


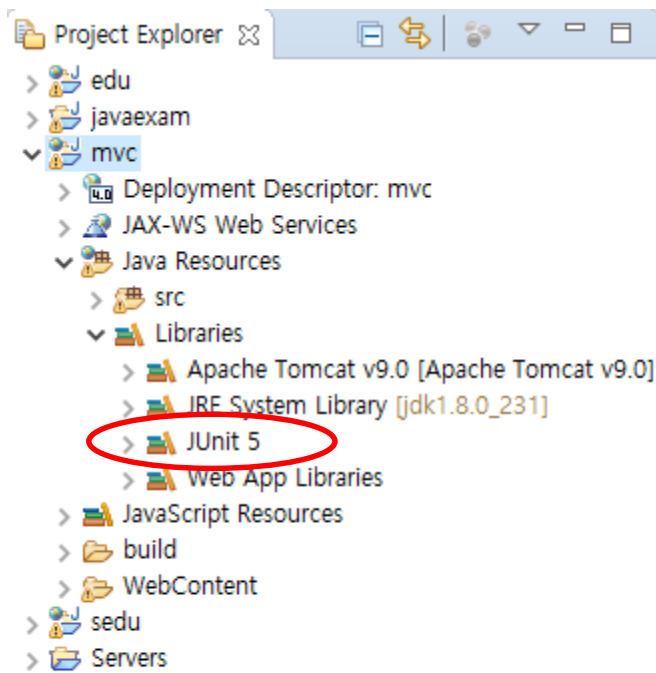
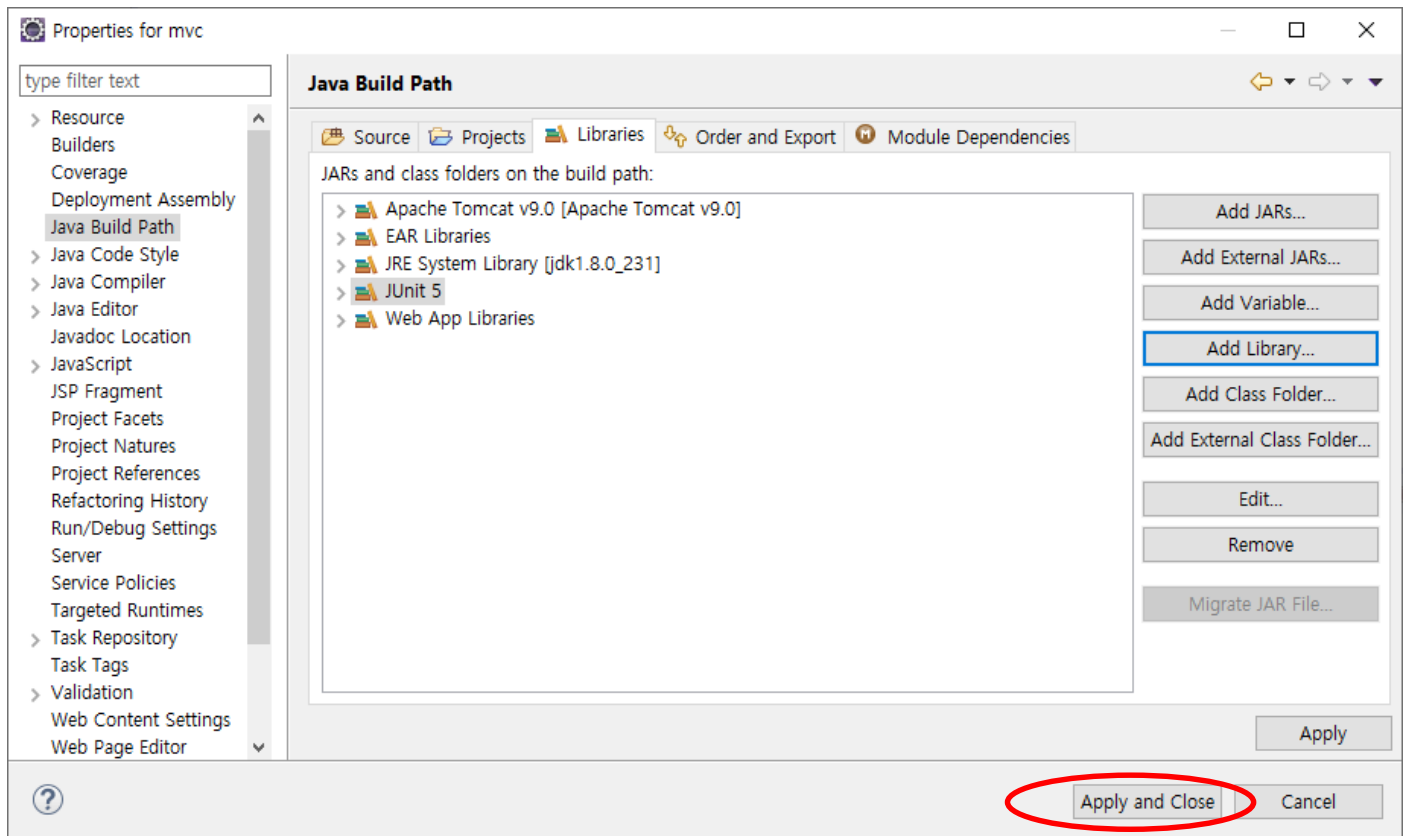
JUnit

Java 언어를 위한 테스트 프레임워크이다. JUnit 를 사용하면 클래스에 구현된 메서드가 주어진 기능을 제대로 수행하는지 단위테스트를 처리할 수 있다.

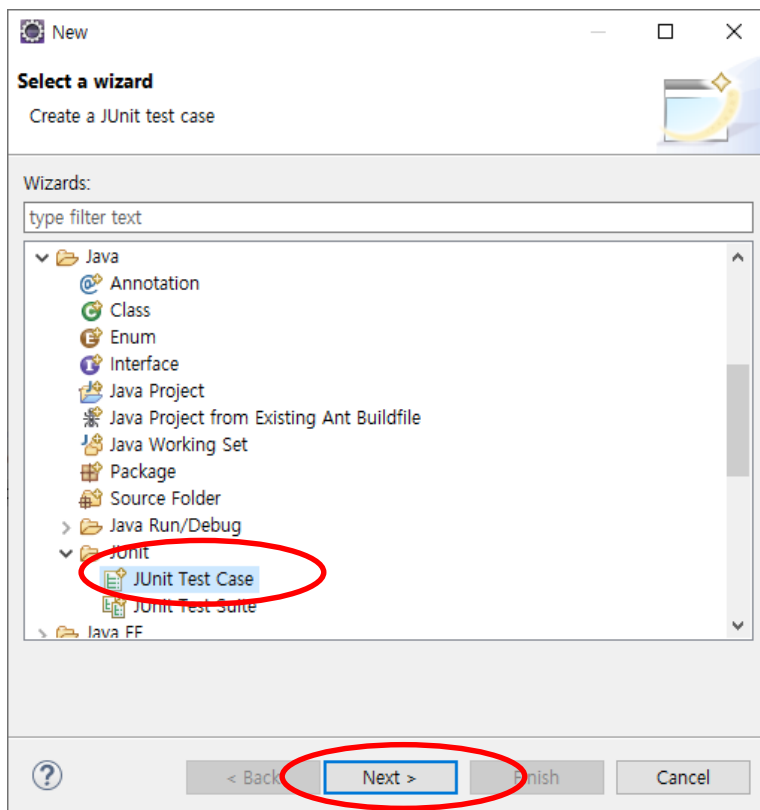
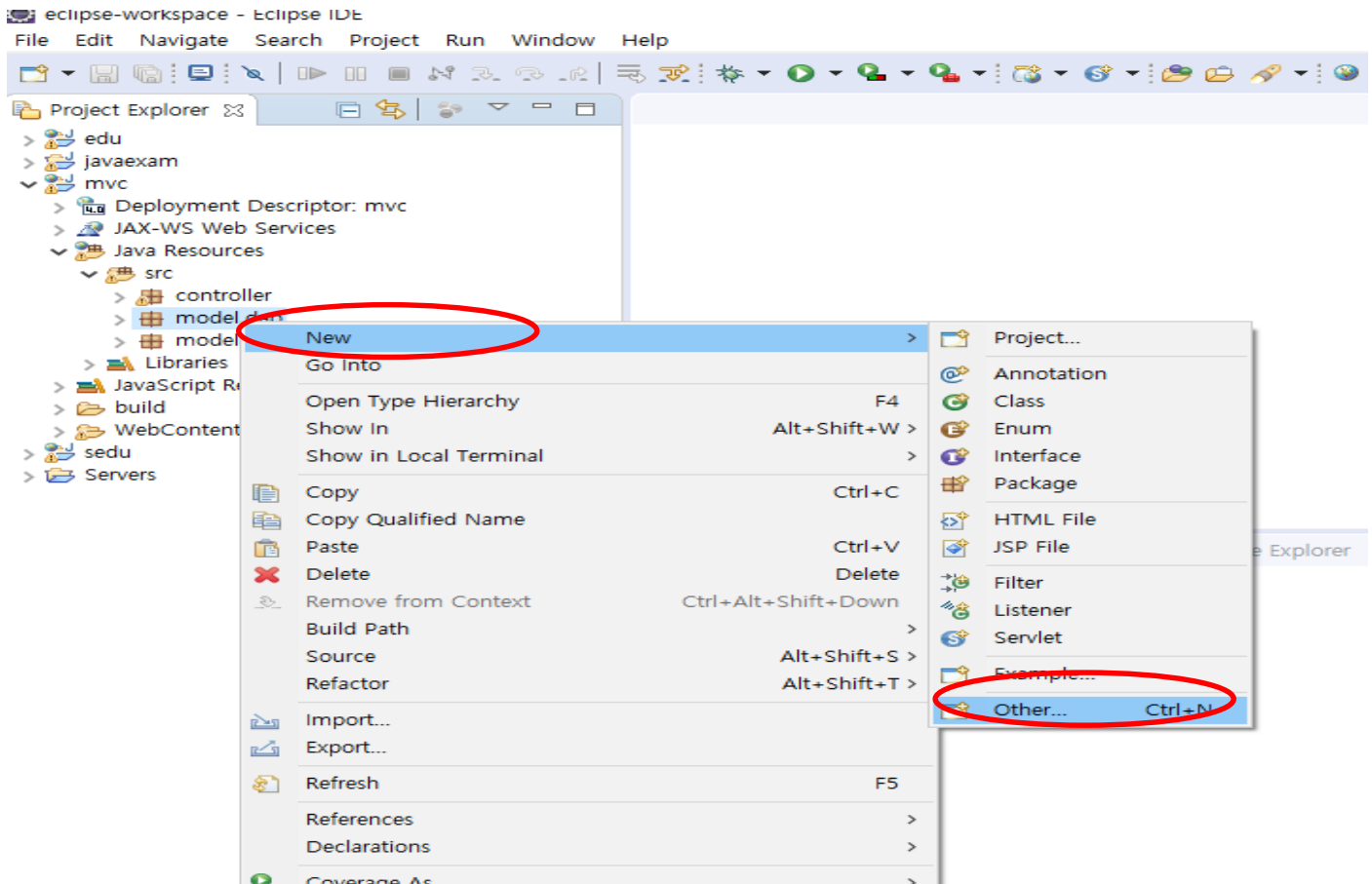
JUnit 라이브러리는 이클립스가 내장하고 있으므로 다음과 같은 **프로젝트 우클릭 -> Properties -> Java Build Path -> Libraries** 탭 클릭하여

현재 사용중인 프로젝트에 라이브러리를 추가하여 간단하게 사용환경을 구축할 수 있다.





[단위 테스트 클래스 만들기]



New JUnit Test Case

Select the name of the new JUnit test case. You have the options to specify the class under test and on the next page, to select methods to be tested.

☐ New JUnit 3 test ☐ New JUnit 4 test ☒ New JUnit Jupiter test

Source folder:

Package:

Name:

Superclass:

Which method stubs would you like to create?

☐ setUpBeforeClass() ☐ tearDownAfterClass()
☐ setUp() ☐ tearDown()
☐ constructor

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

Class under test:

eclipse-workspace - mvc/src/model/dao/VisitorDAOTest.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer

- edu
- javaexam
- mvc
 - Deployment Descriptor: mvc
 - JAX-WS Web Services
 - Java Resources
 - src
 - controller
 - model.dao
 - VisitorDAO.java
 - VisitorDAOTest.java
 - model.vo
 - Libraries
 - JavaScript Resources
 - build
 - WebContent
 - sedu
 - Servers

VisitorDAOTest.java

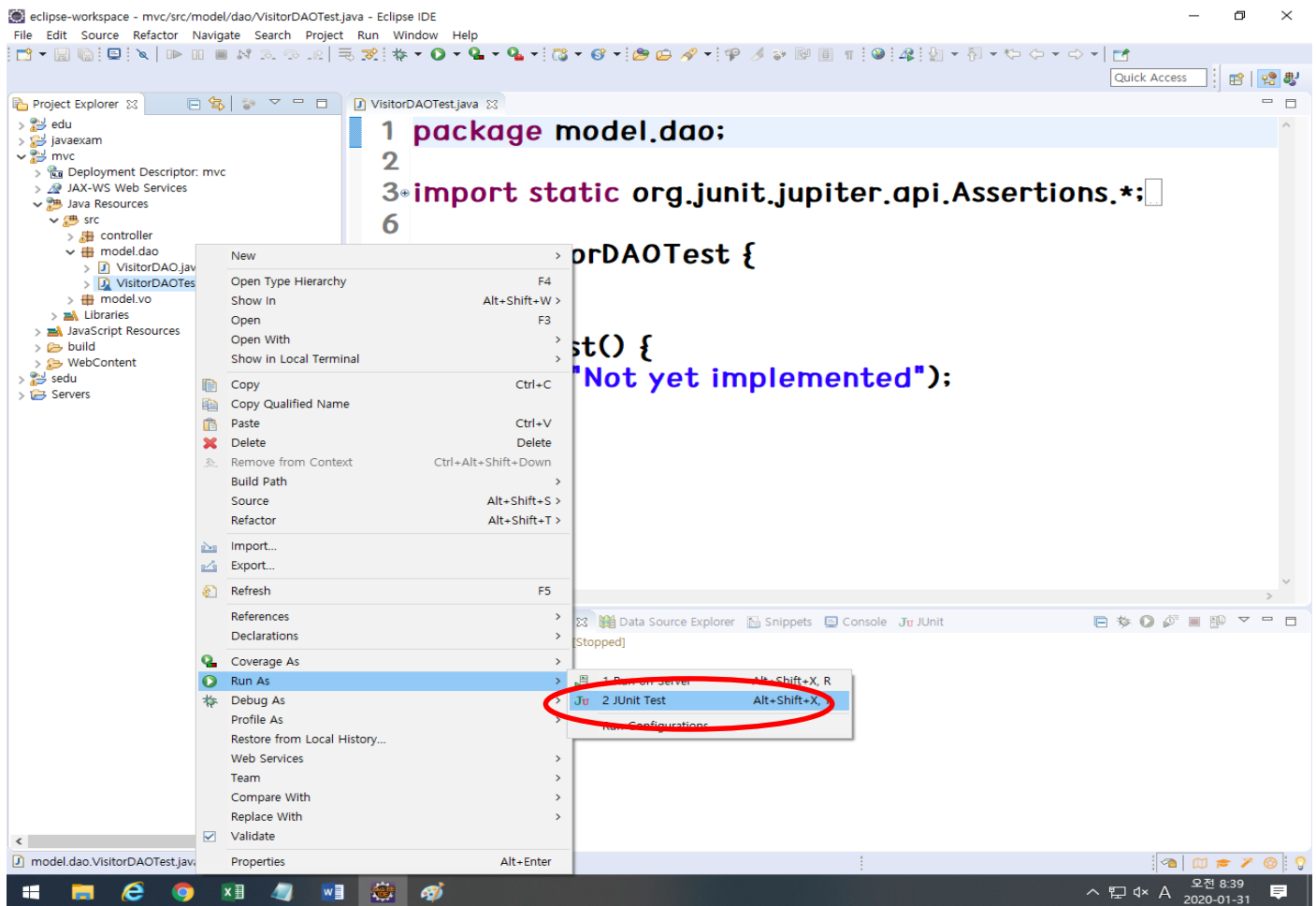
```
1 package model.dao;
2
3 import static org.junit.jupiter.api.Assertions.*;
4
5
6
7 class VisitorDAOTest {
8
9     @Test
10    void test() {
11        fail("Not yet implemented");
12    }
13 }
14
15
```

Markers Properties Servers Data Source Explorer Snippets Console JUnit

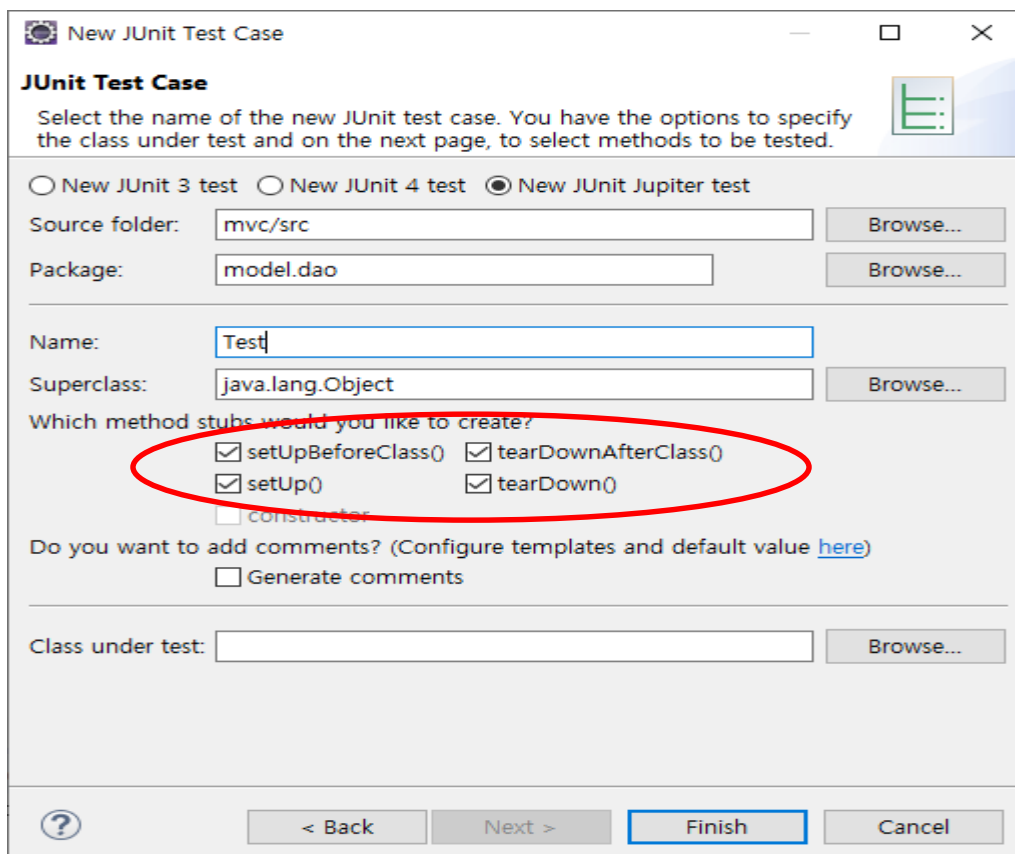
Tomcat v9.0 Server at localhost [Stopped]

Writable Smart Insert 1:1:0

오전 8:39 2020-01-31



[단위 테스트 클래스의 콜백 메서드들]



```
@BeforeClass
public static void setUpBeforeClass() throws Exception {
    System.out.println("최초에 한 번만 수행된다.");
}

@AfterClass
public static void tearDownAfterClass() throws Exception {
    System.out.println("마지막에 한 번만 수행된다.");
}

@Before
public void setUp() throws Exception {
    System.out.println("테스트 메서드가 호출될 때마다 먼저 수행된다.");
}

@After
public void tearDown() throws Exception {
    System.out.println("테스트 메서드가 호출될 때마다 나중에 수행된다.");
}

@Test
public void test1() {
    System.out.println("테스트 기능 수행 메서드 1");
}

@Test
public void test2() {
    System.out.println("테스트 기능 수행 메서드 2");
}
```