

---

# SVN, Mantis, JUnit 사용법

---

## Team Information

양승민 200911400

정세진 200911418

한종철 200911429

# 목 차

1. SVN 사용법

2. Mantis 사용법

3. JUnit 사용법

---

01

---

# SVN 사용법 - SVN 이란?

버전 관리 도구

# 01

## SVN 사용법

### - 버전 관리 프로그램의 장점

- 개발 버전과 릴리즈 버전이 섞이지 않고 쉽게 관리 할 수 있음
- 소스 수정, 추가, 삭제 등 기록이 로그로 남고 되돌리기가 쉬움(많은 파일의 경우 유용)
- 충돌 없이 팀 단위의 동시 개발이 가능함 (Copy-Modify-Merge 패턴)
- 개발자들이 따로 따로 백업을 하지 않아도 됨

---

# 01 SVN 사용법 - 용어

---

- SVN Server : 팀원은 SVN Client 툴을 사용하여 SVN Server에 접속하여 소스코드를 관리한다.
- SVN Client : SVN Server에 접근하기 위한 개발자용 Client 툴
- Repository : SVN Server가 관리하는 소스의 모든 정보를 담고 있는 저장소
- Revision : 갱신 번호. 매번 수정이 발생할 때마다 증가

---

# 01 SVN 사용법 - 용어

---

- Trunk
  - 프로젝트의 중심이 되는 폴더
  - 실제 개발 진행은 이곳에서 진행
- Branches
  - Trunk에서 뺀어 나온 또 하나의 프로젝트로서, 메인 프로젝트에서 또 다른 작은 분류로 배서 따로 개발을 진행
- Tags
  - 버전 업이나 특정 시점에서 프로젝트 전체를 복사하여 보관하는 곳

---

# 01 SVN 사용법 - 명령어

---

- Import : 최초로 Repository에 자료를 올림
- Export : Repository에서 자료를 내려 받음.  
버전 관리 정보 제외
- Checkout : Repository에서 자료를 내려 받음.  
버전 관리 정보 포함
- Commit : Checkout 한 소스의 변경내역을  
Repository에 반영하는 명령어. Revision이 1  
증가
- Update : Checkout 한 소스를 Repository의  
최신 소스로 갱신

---

# 01 SVN 사용법 – SVN for Eclipse

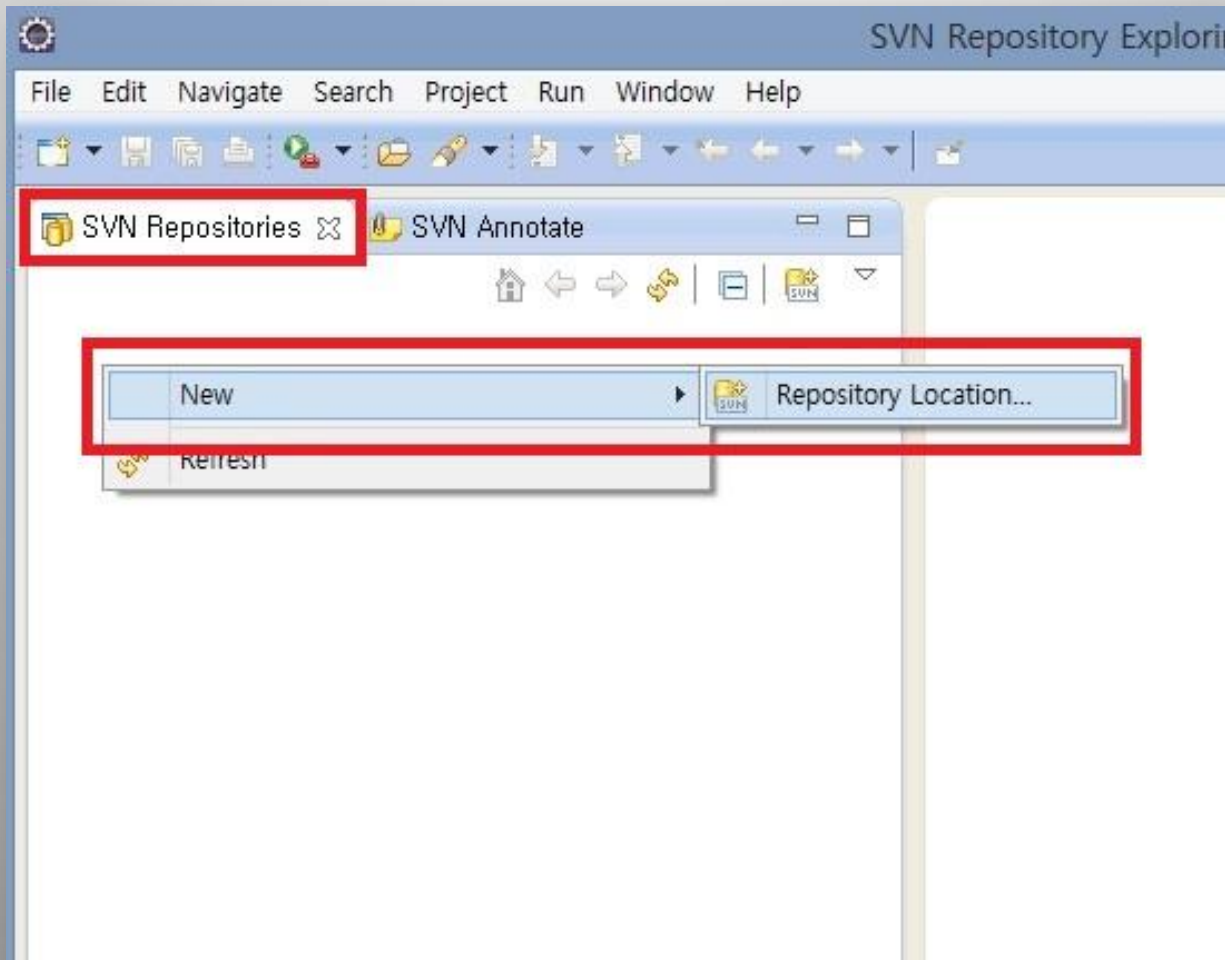
---

<http://akj61300.blog.me/80187272710>

Subclipse 설치법

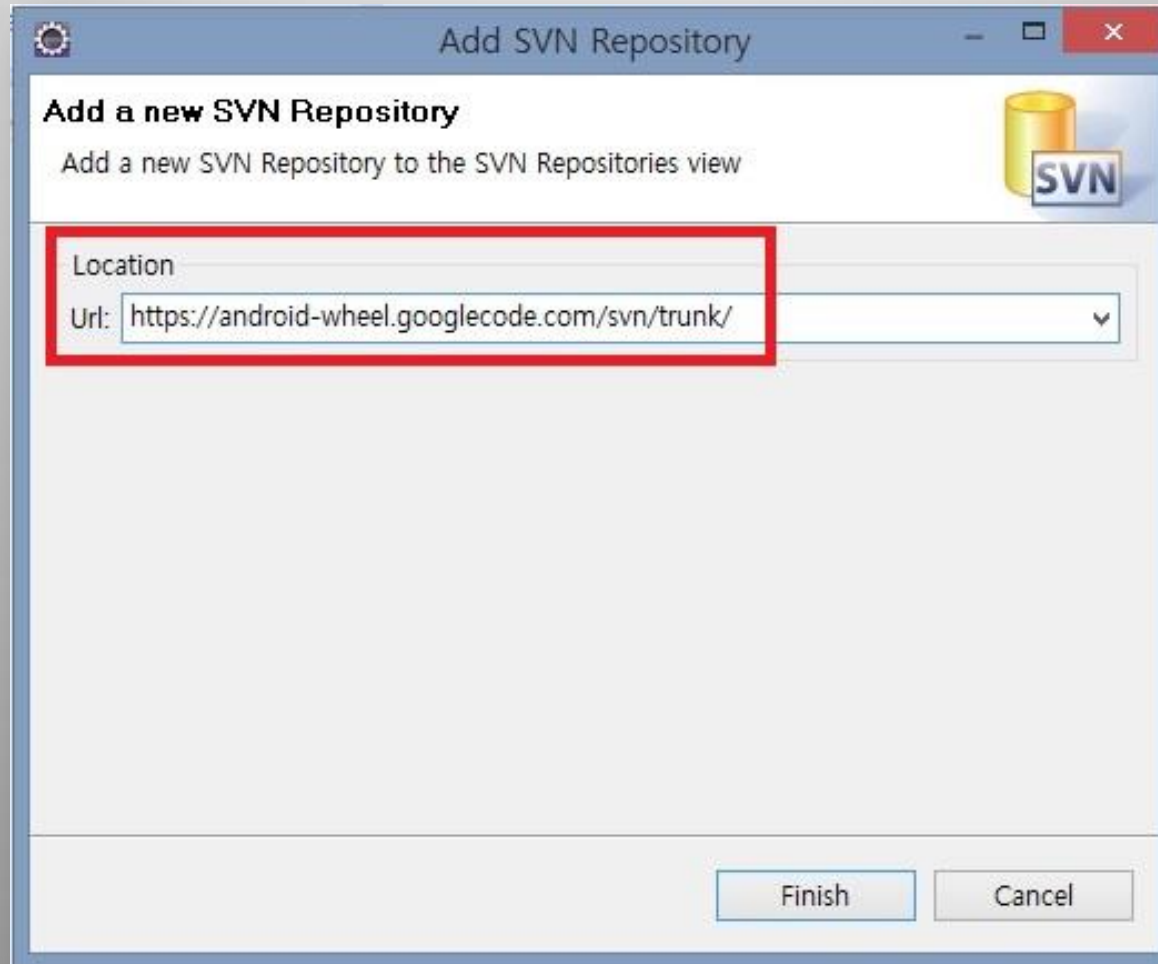


# 01 SVN 사용법 - Subclipse

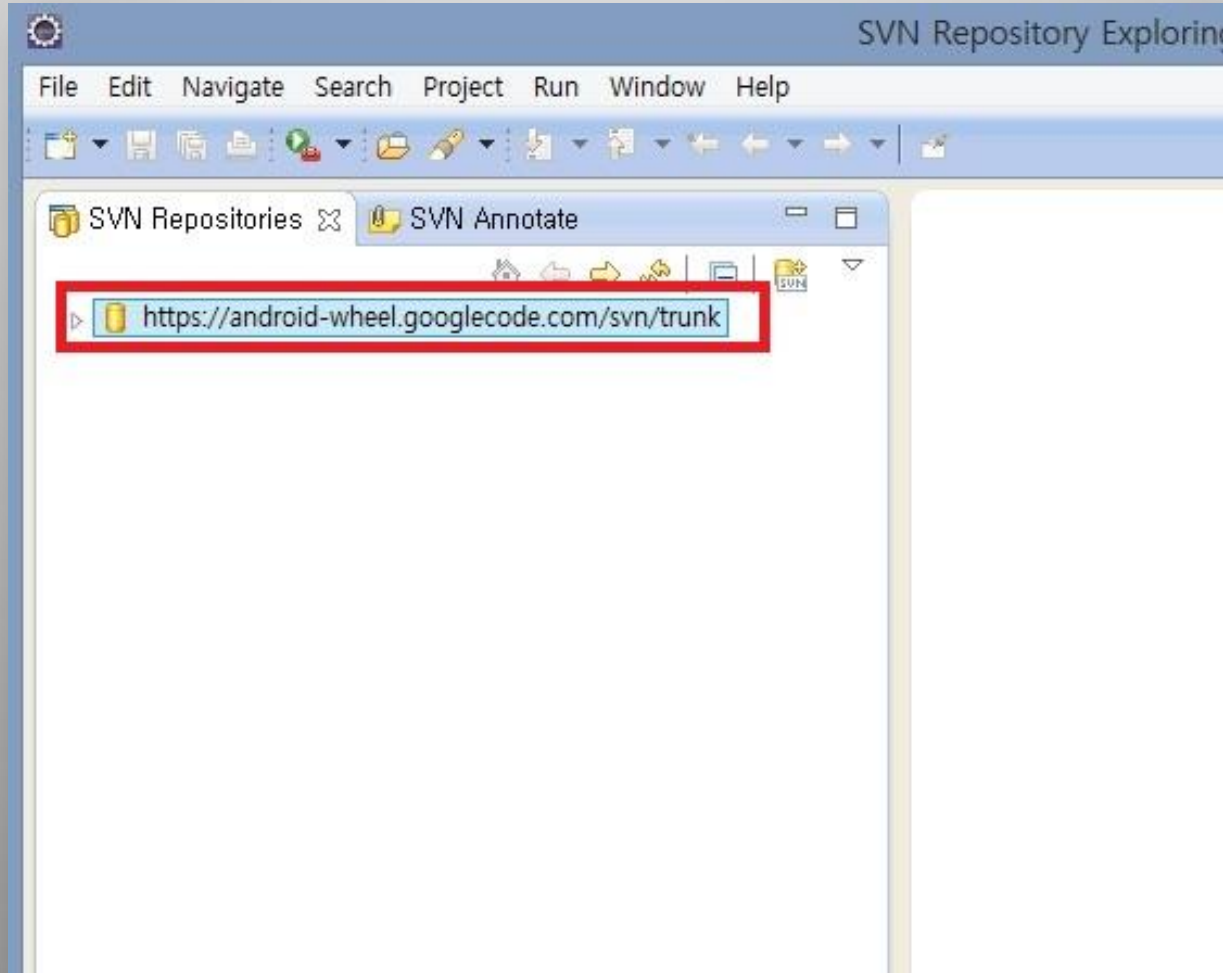


# 01

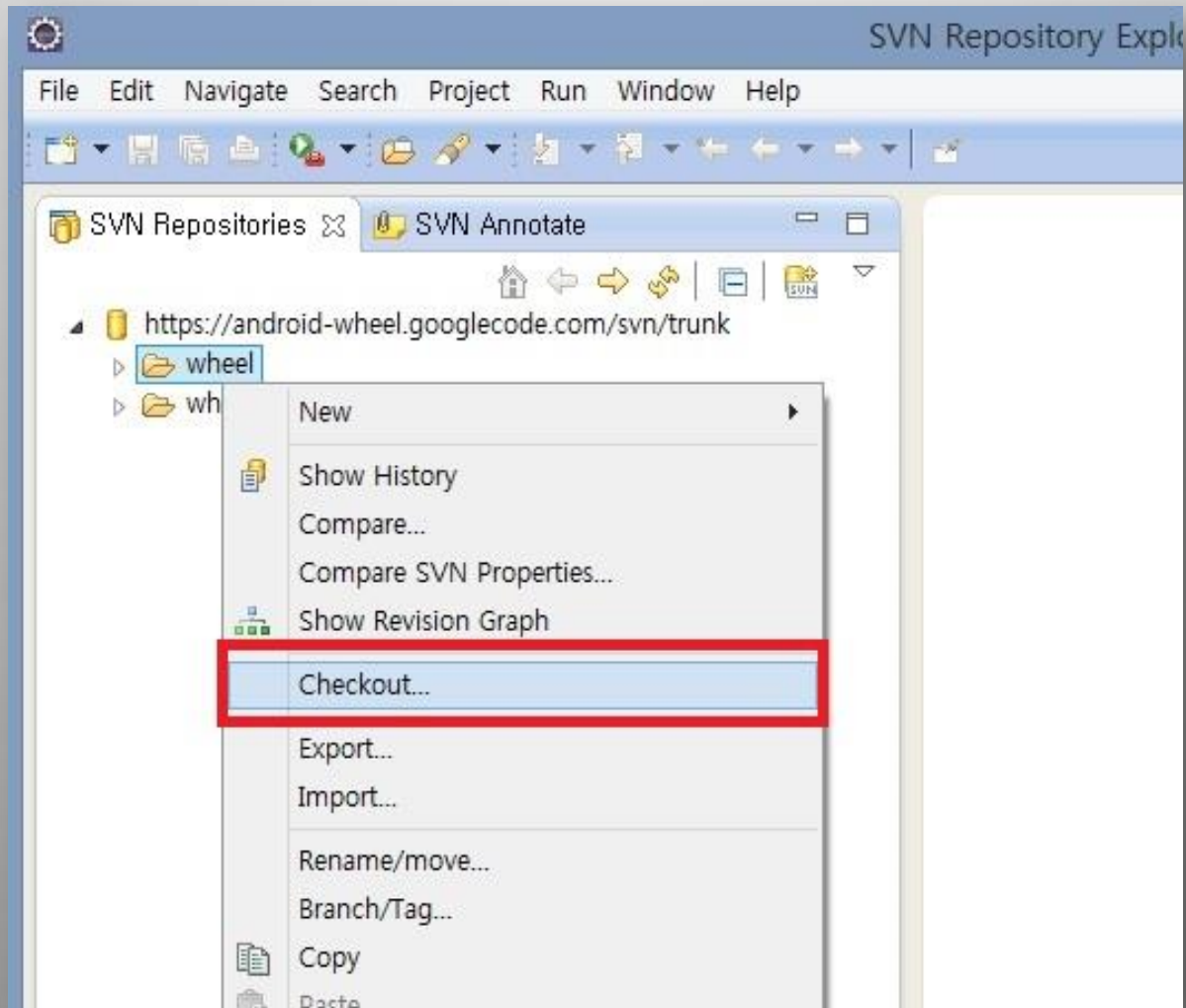
## SVN 사용법 – Subclipse



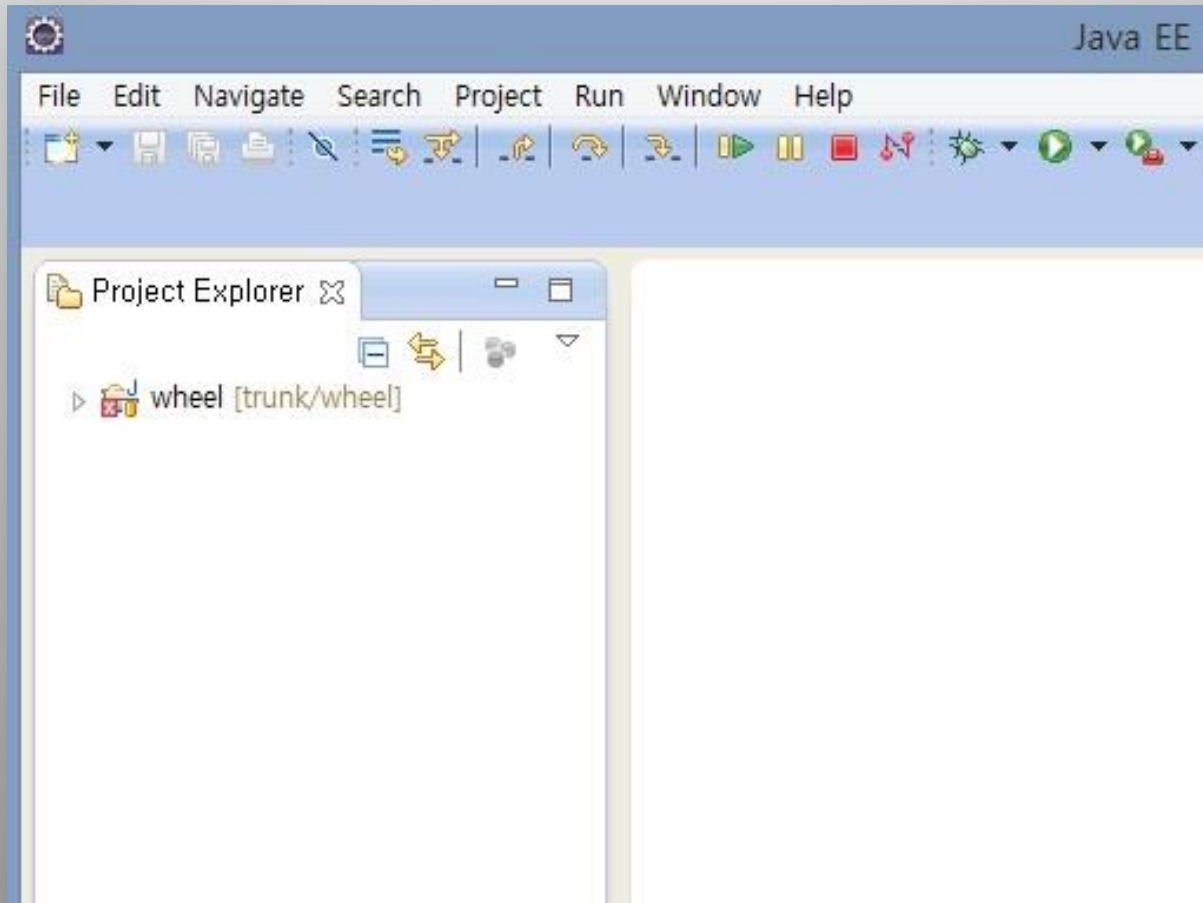
# 01 SVN 사용법 - Subclipse



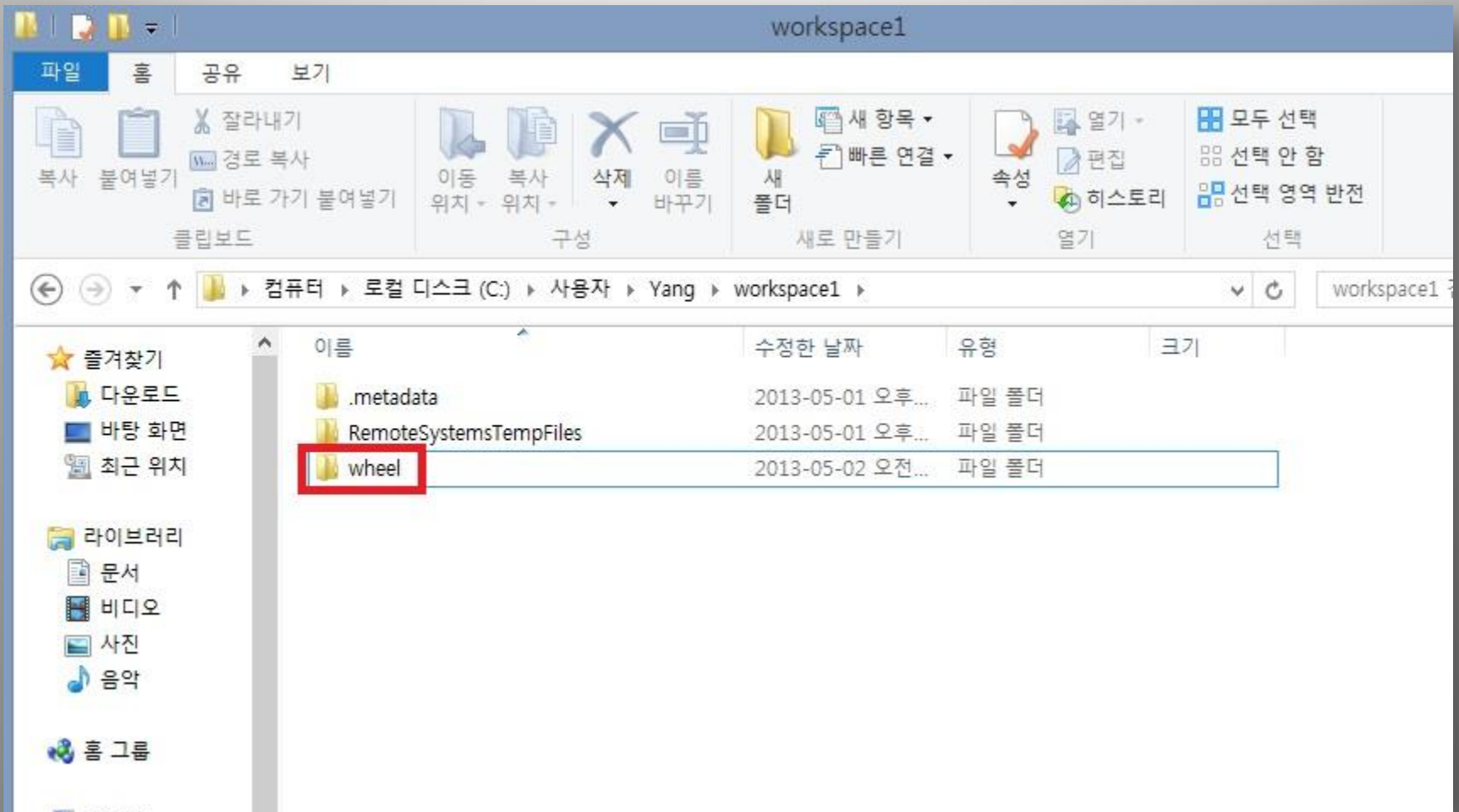
# 01 SVN 사용법 – Subclipse



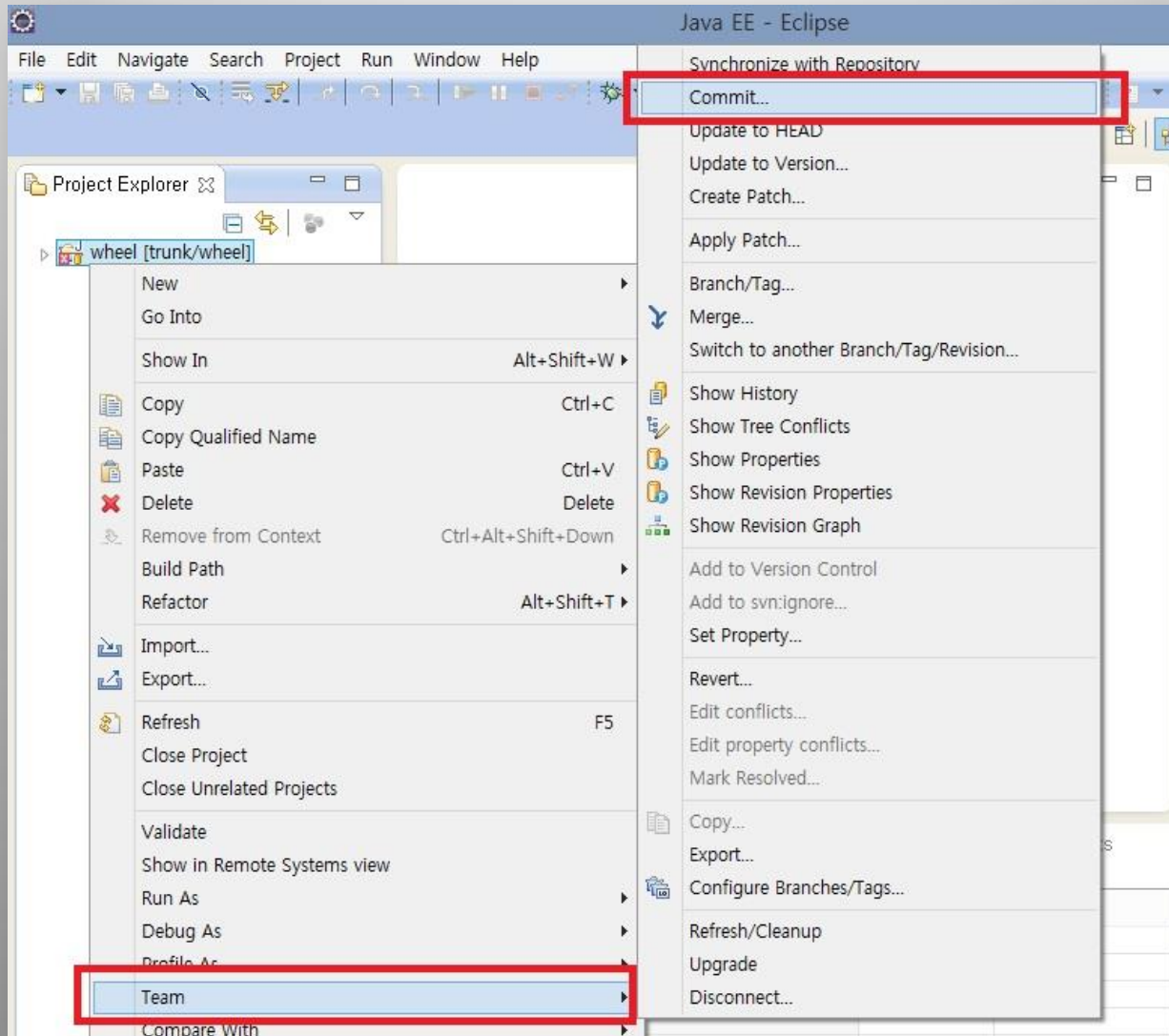
# 01 SVN 사용법 - Subclipse



# 01 SVN 사용법 - Subclipse

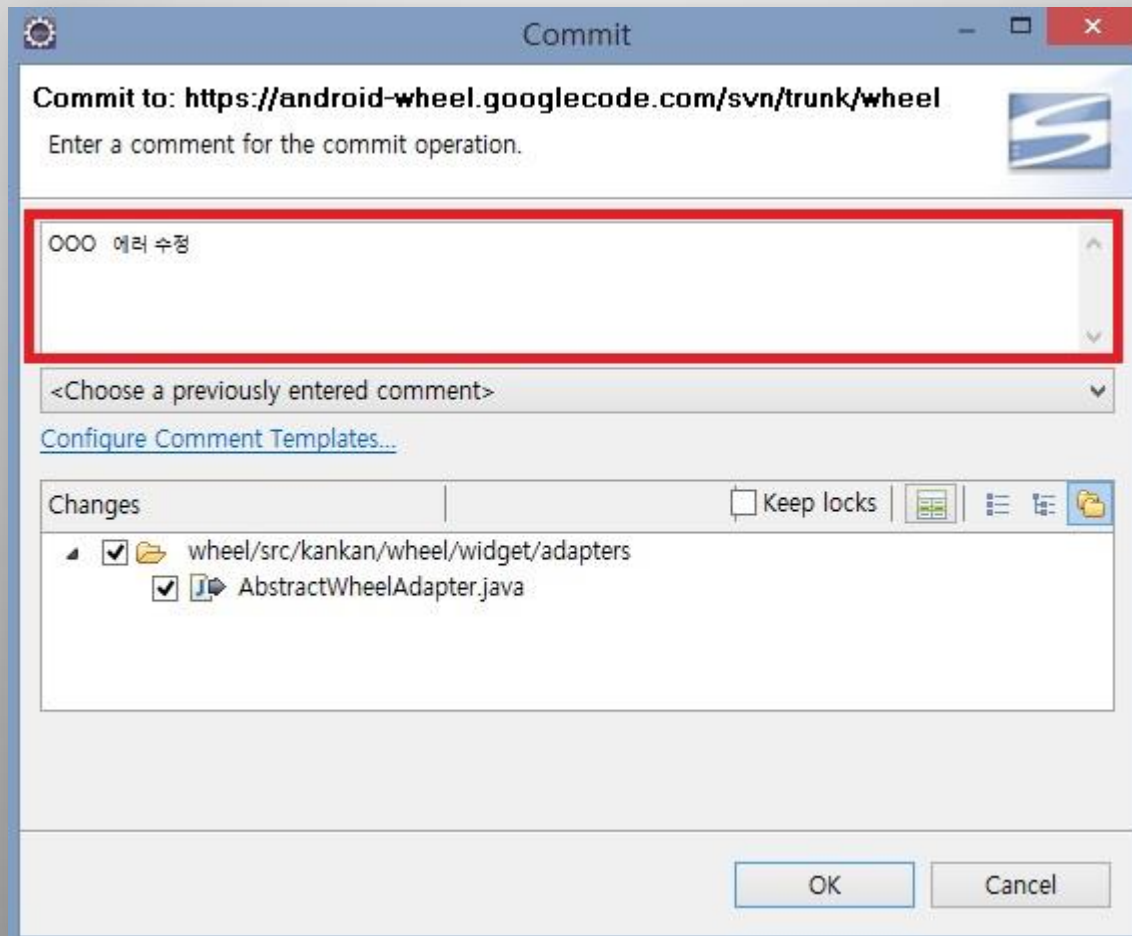


# 01 SVN 사용법 - Subclipse



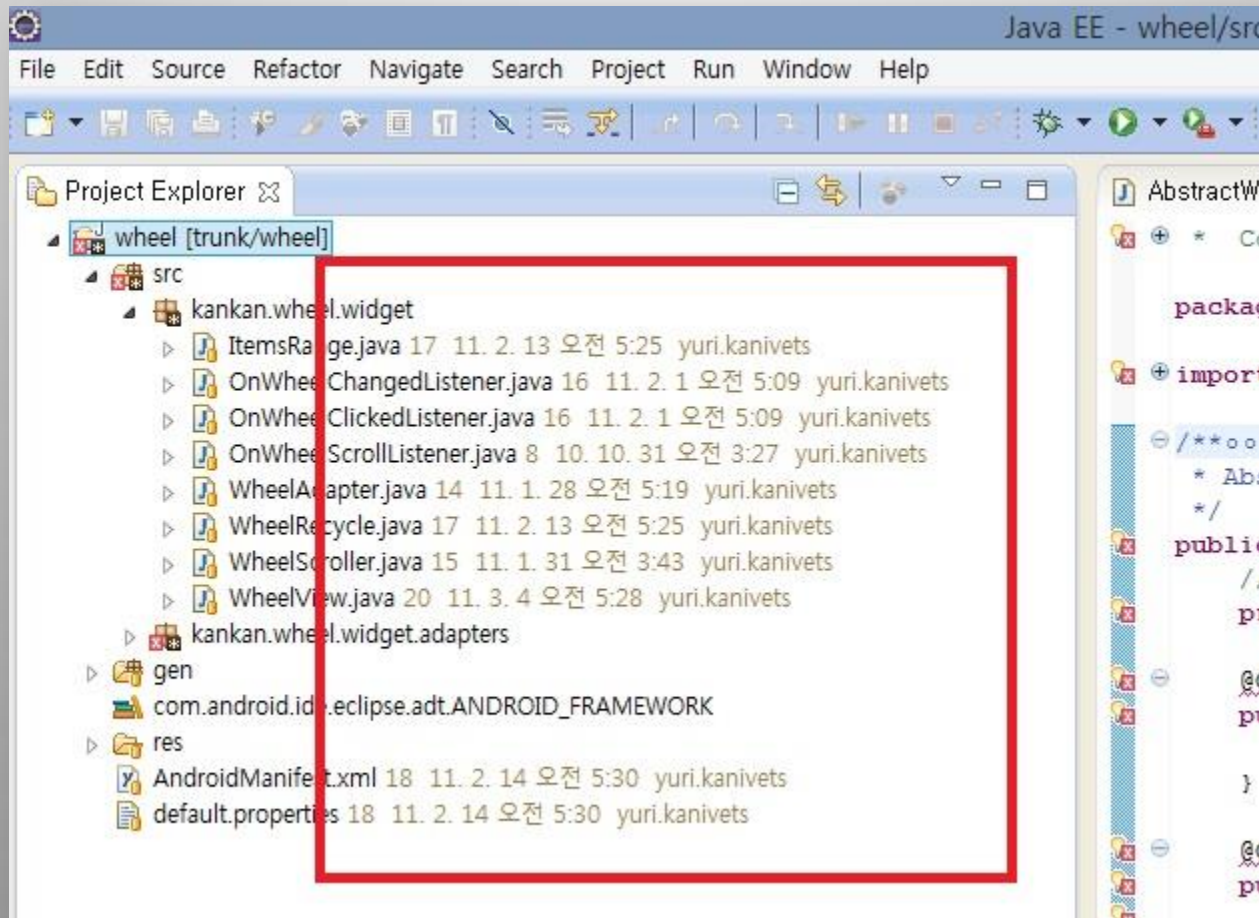
# 01

## SVN 사용법 – Subclipse



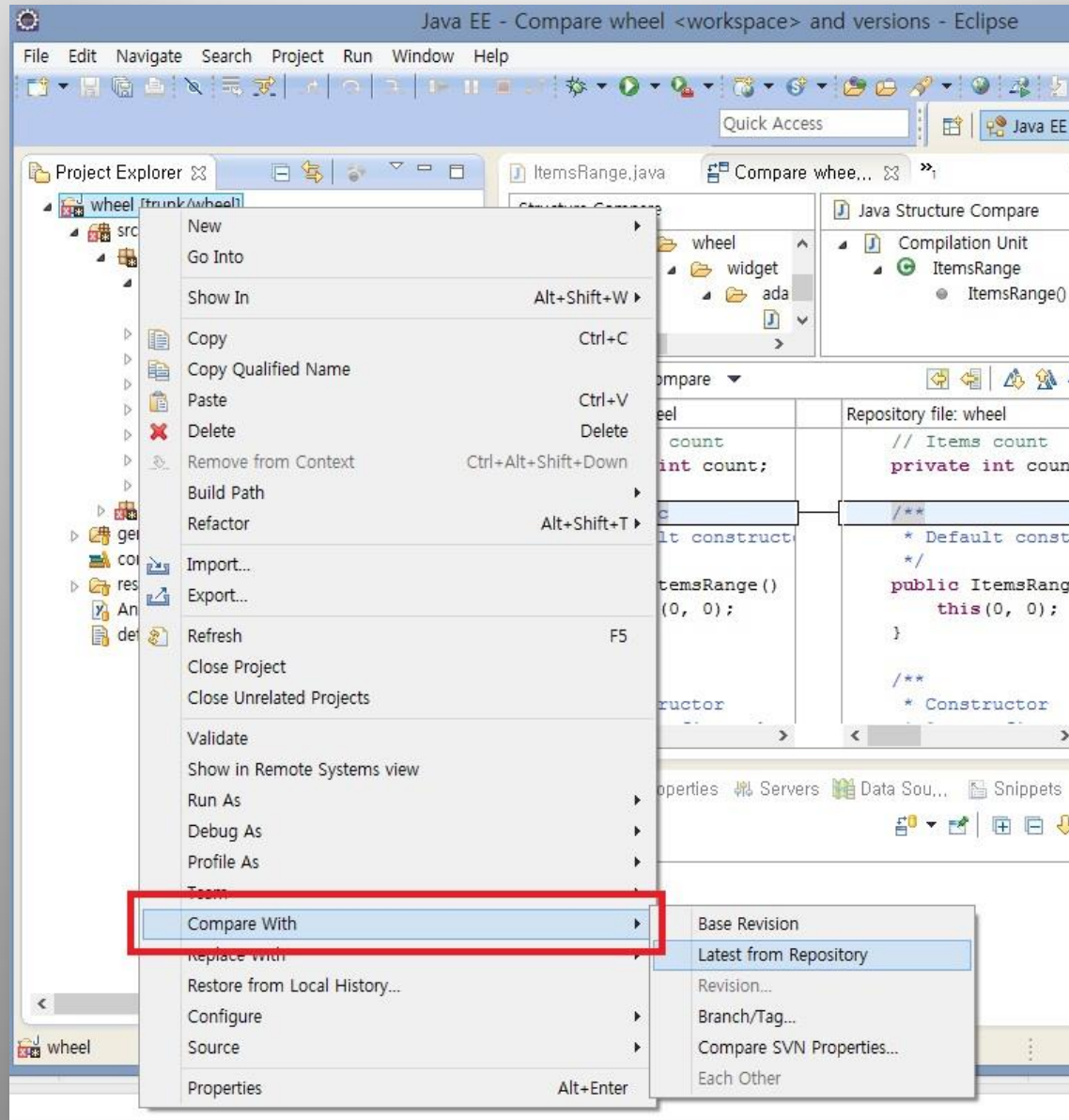


# 01 SVN 사용법 - Subclipse

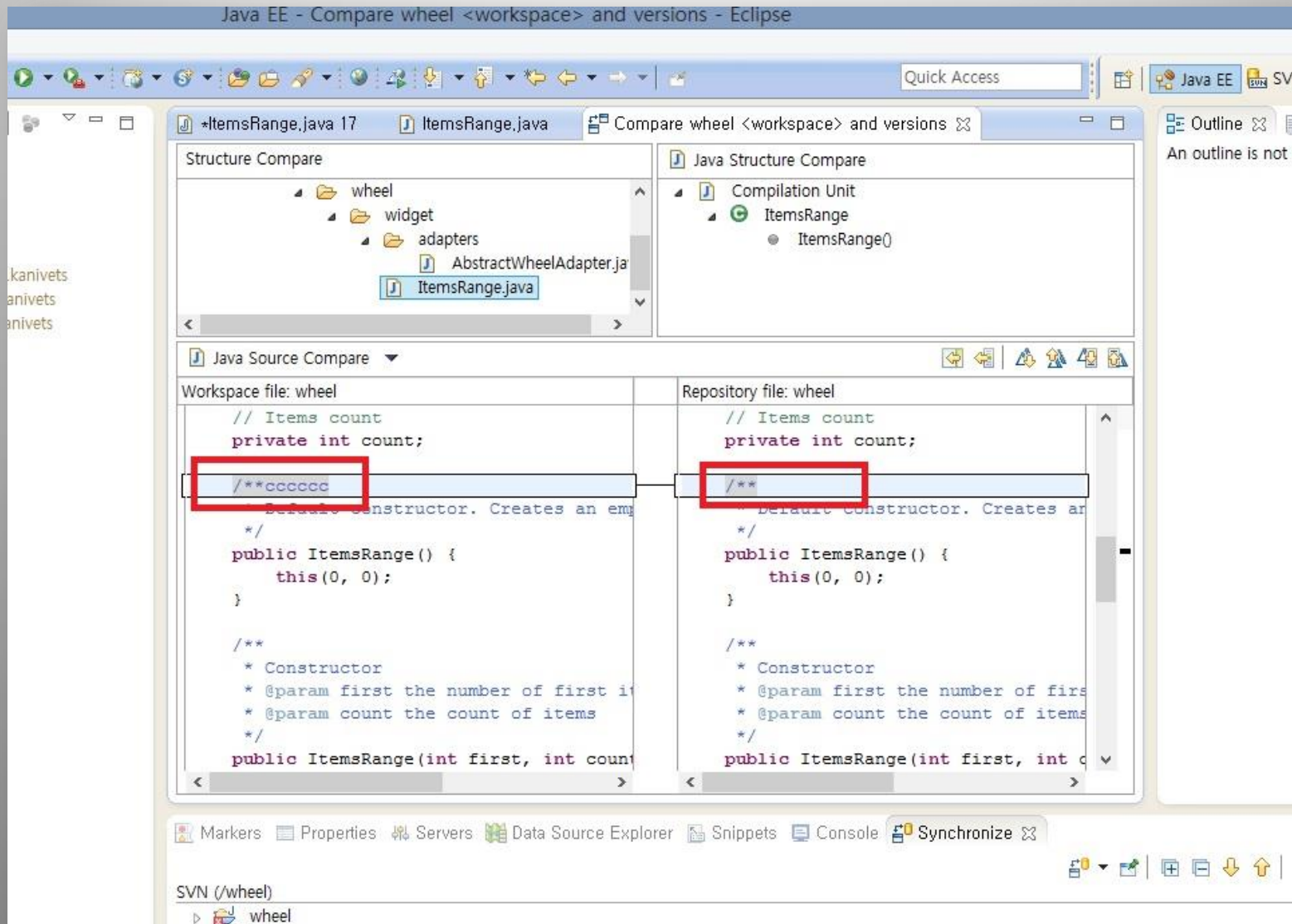


# 01

# SVN 사용법 - Subclipse



# 01 SVN 사용법 - Subclipse



---

02

---

## Mantis 사용법 - Manstis 란?

버그 추적 시스템

## 02

# Mantis 사용법

## - 버그 추적 시스템의 장점

- Web을 통해 접근이 용이하고 쉽게 사용할 수 있음
- 모든 버그는 DB에 보관되며 추적이 가능
- 해당 버그의 담당자가 누구이며 현재 진행 업무를 파악할 수 있음
- 개발자와 테스터간의 커뮤니케이션 비용을 줄일 수 있다.

---

## 02 Mantis 사용법

---

<http://lycobs.egloos.com/2142482>

Mantis 설치법

# 02 Mantis 사용법



로그인

사용자 ID	administrator
비밀번호	••••
로그인 정보 저장	<input type="checkbox"/>

로그인

[ 사용자 신규 등록 ] [ 비밀번호 찾기 ]

**WARNING:** You should disable the default "administrator" account or change its password.

**WARNING:** Admin directory should be removed.

# 02 Mantis 사용법



로그인한 ID :: papimomi (메시지)

2013-04-18 16:08 KST

프로젝트: 모든 프로젝트 이동

[첫화면](#) | [내 페이지](#) | [이슈 보기](#) | [이슈 보고하기](#) | [변경 기록](#) | [로드맵](#) | [요약](#) | [관리](#) | [계정 관리](#) | [로그아웃](#)

이슈 번호 이동

## 할당되지 않음 [ ^ ] ( 1 - 2 / 2 )

- 0000003 [MyBuildingServer] 클라이언트에서 방 생성시 오류  
치명적 - 2013-04-17 20:18
- 0000002 [MyBuildingServer] 빌딩을 모두 파괴  
기능추가 - 2013-04-17 20:09

## 해결됨 [ ^ ] ( 0 - 0 / 0 )

## 내가 모니터링 중 [ ^ ] ( 0 - 0 / 0 )

## 내가 보고함 [ ^ ] ( 0 - 0 / 0 )

## 최근에 변경됨 [ ^ ] ( 1 - 3 / 3 )

- 0000003 [MyBuildingServer] 클라이언트에서 방 생성시 오류  
치명적 - 2013-04-17 20:18
- 0000001 [MyBuildingServer] 새로운 식사를 추가할 필요  
[모든 프로젝트] General - 2013-04-17 20:09
- 0000002 [MyBuildingServer] 빌딩을 모두 파괴  
기능추가 - 2013-04-17 20:09

신규 정보 부족 이슈 검토 확인된 이슈 할당된 이슈 해결된 이슈 닫음

Copyright © 2000 - 2013 MantisBT Team  
papimomi@brainguys.net





# 02 Mantis 사용법

Mantis - Microsoft Internet Explorer

주소 http://localhost/mantis/manage\_proj\_create\_page.php

MANTIS

로그인한 ID :: administrator (관리자) 2008-04-22 11:01 JST 프로젝트: [ 모든 프로젝트 ] 이동

[첫화면](#) | [내 페이지](#) | [이슈 보기](#) | [이슈 보고하기](#) | [변경 기록](#) | [Roadmap](#) | [오락](#) | [문서](#) | [관리](#) | [뉴스 편집](#) | [계정 관리](#) | [로그아웃](#) Issue # 이동

[ [사용자 관리](#) ] [ [프로젝트 관리](#) ] [ [커스텀 필드 관리](#) ] [ [전역 프로파일 관리](#) ] [ [설정사항 관리](#) ]

프로젝트 추가

* 프로젝트 이름	<input type="text" value="드디어실행"/>
상태	개발버전
상태 보기	공개
업로드 파일 경로	<input type="text"/>
설명	<input type="text" value="머림게 실행했습니다."/>

프로젝트 추가

Mantis 1.1.1[<sup>^</sup>]  
Copyright © 2000 - 2008 Mantis Group  
webmaster@example.com  
16 total queries executed.  
14 unique queries executed.

로컬 인트라넷

# 02 Mantis 사용법

로그인한 ID :: administrator (관리자) 2008-04-22 11:03 JST 프로젝트: 드디어실형

첫화면 | 내 페이지 | 이슈 보기 | 이슈 보고하기 | 변경 기록 | Roadmap | 요약 | 문서 | 관리 | 뉴스 편집 | 계정 관리 | 로그아웃

상세 보고 내역을 입력하세요 [ 상세 보고 ]

\*분류 (select)

재발생 가능성 항상발생

중요도(심각성) 보통

우선 순위 낮음

\*요약 실행은 아무나 하나.

\*설명 초보 개발자의 좌충우돌 mantis 실행기

추가 정보 사항

파일 업로드 (최대 크기: 5,000k) 찾아보기...

상태 보기  공개  비공개

계속 보고  (이슈 보고를 계속 하려면 선택)

\* 필수 보고서 제출

Mantis 1.1.1[<sup>^</sup>]  
Copyright © 2000 - 2008 Mantis Group  
webmaster@example.com  
18 total queries executed.  
16 unique queries executed.

MANTIS  
BUG TRACKING SYSTEM

로컬 인터넷

# 02 Mantis 사용법

파일(F) 편집(E) 보기(V) 즐겨찾기(A) 도구(T) 도움말(H)

주소(D) http://localhost/mantis/view\_all\_bug\_page.php 이동 연결 >>

MANTIS

로그인한 ID :: administrator (관리자) 2008-04-22 11:04 JST 프로젝트: 드디어실함 이동

[첫화면](#) | [내 페이지](#) | [이슈 보기](#) | [이슈 보고하기](#) | [변경 기록](#) | [Roadmap](#) | [요약](#) | [문서](#) | [관리](#) | [뉴스 편집](#) | [계정 관리](#) | [로그아웃](#) Issue # 이동

@ 검색: 필터 [ 고급 필터 ] [ Create Permalink ] 필터 초기화 현재 필터 저장

이슈 보기 (1 - 1 / 1) [ 보고서 출력 ] [ CSV 내보내기 ]

	P	이슈 번호	#	종류	중요도(심각성)	상태	최종 갱신	요약
<input type="checkbox"/>		<a href="#">0000001</a>			보통	새로운 이슈	2008-04-22	실행은 아무나 하나.

전부 선택 | 이동

새로운 이슈 정보 부족 이슈 검토 확인된 이슈 할당된 이슈 해결된 이슈 폐쇄된 이슈

Mantis 1.1.1 [^]  
Copyright © 2000 - 2008 Mantis Group  
[webmaster@example.com](mailto:webmaster@example.com)  
27 total queries executed.  
21 unique queries executed.

# 02 Mantis 사용법

이슈 상세 정보 보기 [ [메모로 이동](#) ] [ [알림 메시지 보내기](#) ] [ >> ] [ [이슈 역사](#) ] [ [인쇄](#) ]

ID	프로젝트	분류	상태 보기	보고 날짜	최종 갱신
0000005	MyBuildingServer	기능추가	공개	2013-04-18 16:41	2013-04-18 16:41
보고자	papimomi				
이슈 할당	AuraWalker				
우선 순위	보통	중요도(심각성)	기능 개선	재발생 가능성	N/A
상태	할당된 이슈	해결 상태	개설		
플랫폼		운영체제		버전	
제품 번호	1.0.x				
목표 버전	1.1.x	해결된 버전			
요약	0000005: 건물 스프라이트 변경				
설명	건물에 따라서 이미지를 만들어가지고 랜덤으로 보여질 수 있도록 기능을 추가				
태그	붙인 태그가 없습니다.				
태그 붙이기	("," 사용자 분리) <input type="text"/> <input type="button" value="기본 태그"/> <input type="button" value="붙이기"/>				
첨부 파일					

# 02

## Mantis 사용법

### 내가 모니터링 중 [ ^ ] ( 1 - 2 / 2 )

0000001



새로운 식사를 추가할 필요

[모든 프로젝트] General - **2013-04-18 16:45**

0000003



클라이언트에서 방 생성시 오류

치명적 - **2013-04-18 16:45**

# 02

## Mantis 사용법

이슈 보기 (1 - 2 / 2) [ 보고서 출력 ] [ CSV 내보내기 ] [ Excel 내보내기 ]

		P	ID	#	📧	분류	중요도(심각성)	상태	업데이트됨▼	요약
<input type="checkbox"/>		^	<a href="#">0000004</a>			매우 급함	장애	할당된 이슈 (papimomi)	2013-04-18	건물이 세워지지 않는 오류
<input type="checkbox"/>		-	<a href="#">0000002</a>			기능추가	보통	할당된 이슈 (papimomi)	2013-04-18	빌딩을 모두 파괴

모두 선택    이동

# 02

## Mantis 사용법 - SVN과 연동

☐ 메모

([0000001](#))

**papimomi** (매니저) [ 비공개 ]

2013-04-18 17:39

편집

삭제

공개로 하기

Test fixed issue [0000004](#) by papimomi

([0000002](#))

**papimomi** (매니저) [ 비공개 ]

2013-04-18 17:45

편집

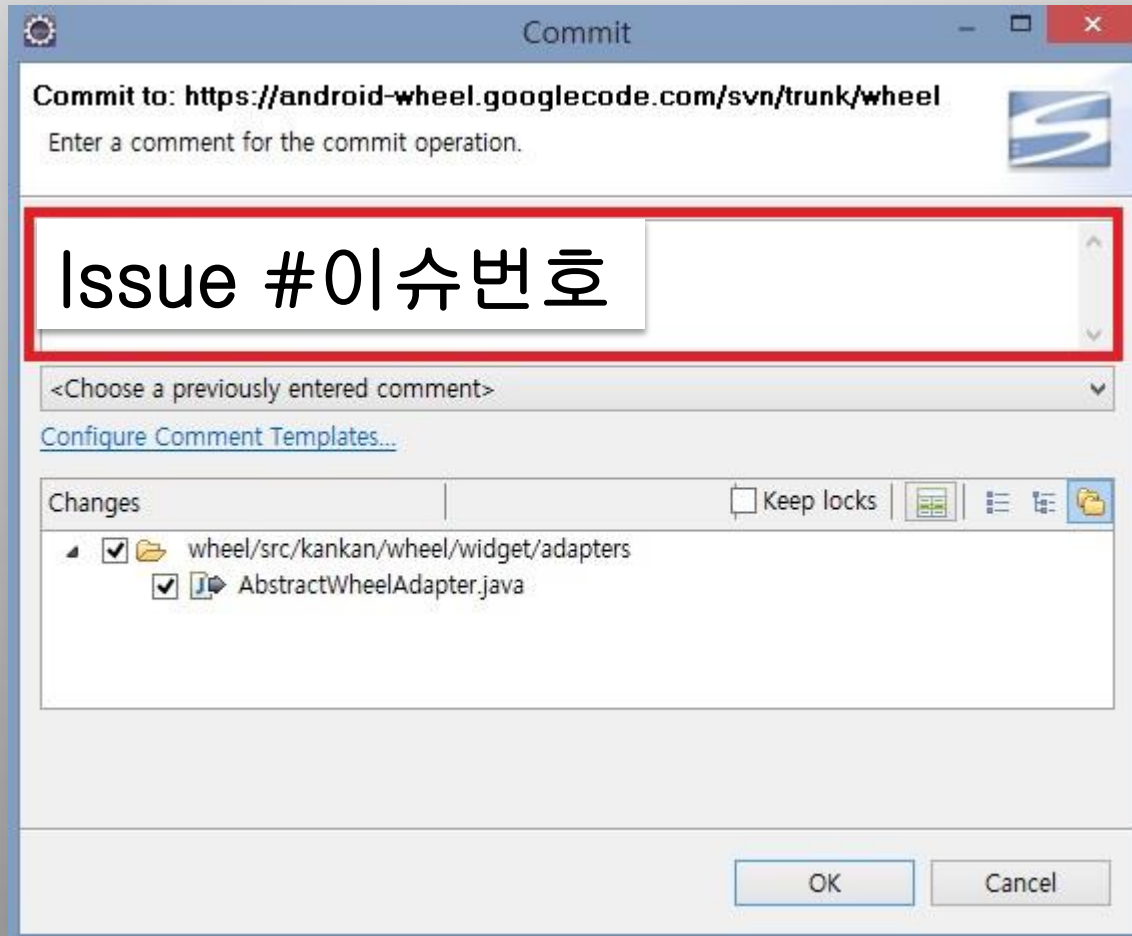
삭제

공개로 하기

Test issue [0000004](#) by papimomi

# 02

## Mantis 사용법 - SVN과 연동





---

# 03

---

## JUnit 사용법 - JUnit 이란?

Unit테스트를 쉽게 해주는  
프레임 워크, 테스트 툴

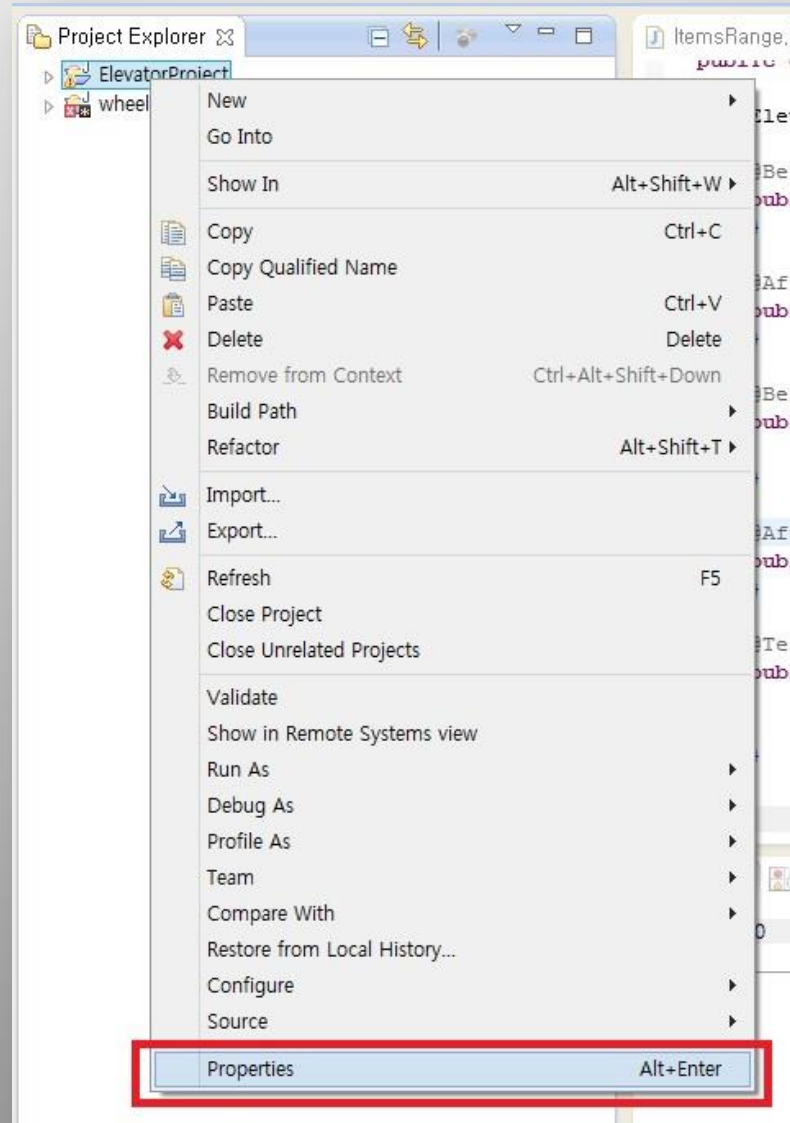
# 03

## JUnit 사용법

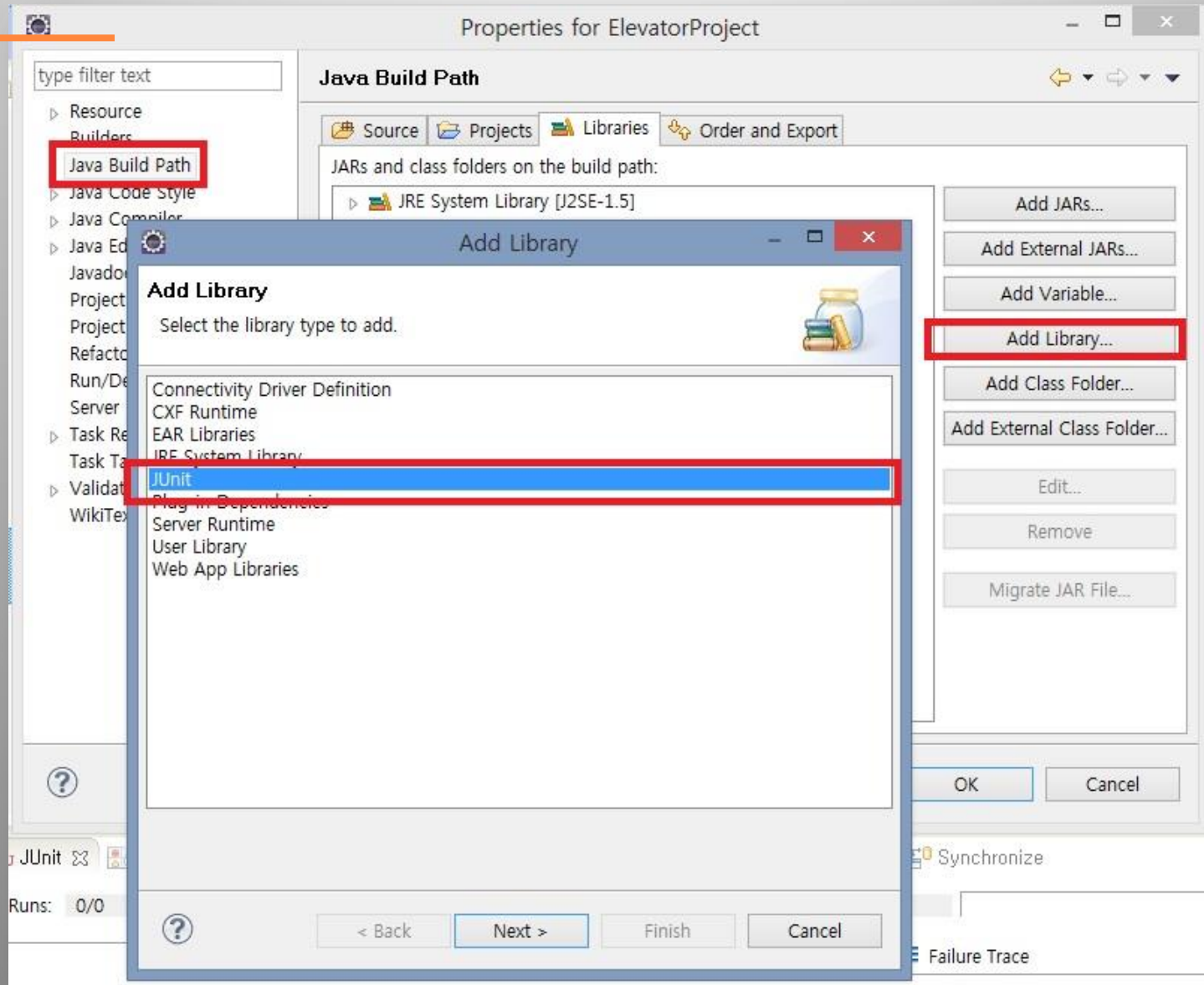
### - TestCase의 assert메소드

assertEquals()	두 객체 또는 값이 동일한지 판단
assertFalse()	false인지 판단
assertNotNull()	해당 객체가 null이 아닌지 판단
assertNull()	해당 객체가 null인지 판단
assertNotSame()	두 객체가 동일하지 않은지 판단
assertSame()	두 객체가 동일한 객체인지 판단
assertTrue()	true인지 판단
fail()	테스트 실패처리

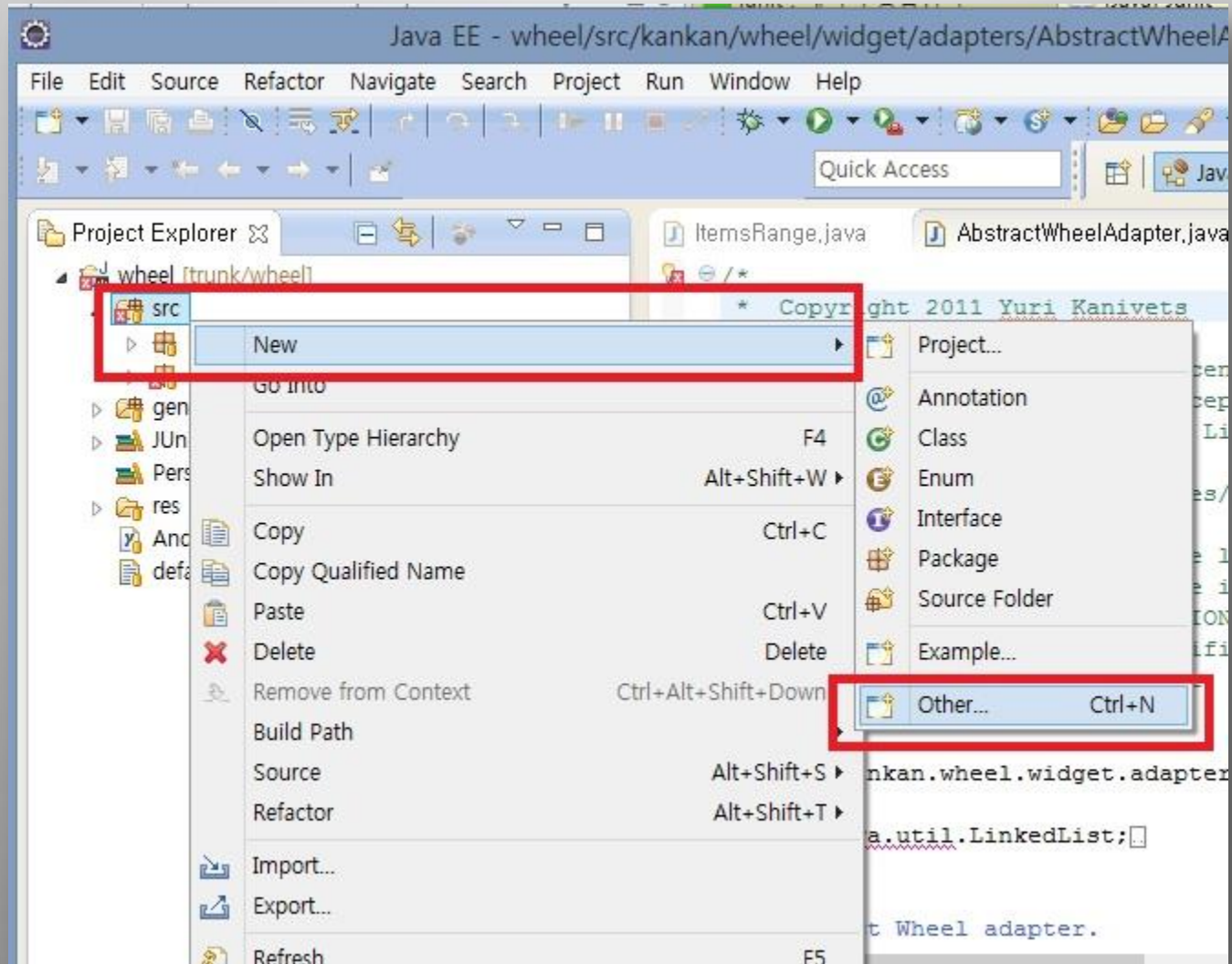
# 03 JUnit 사용법



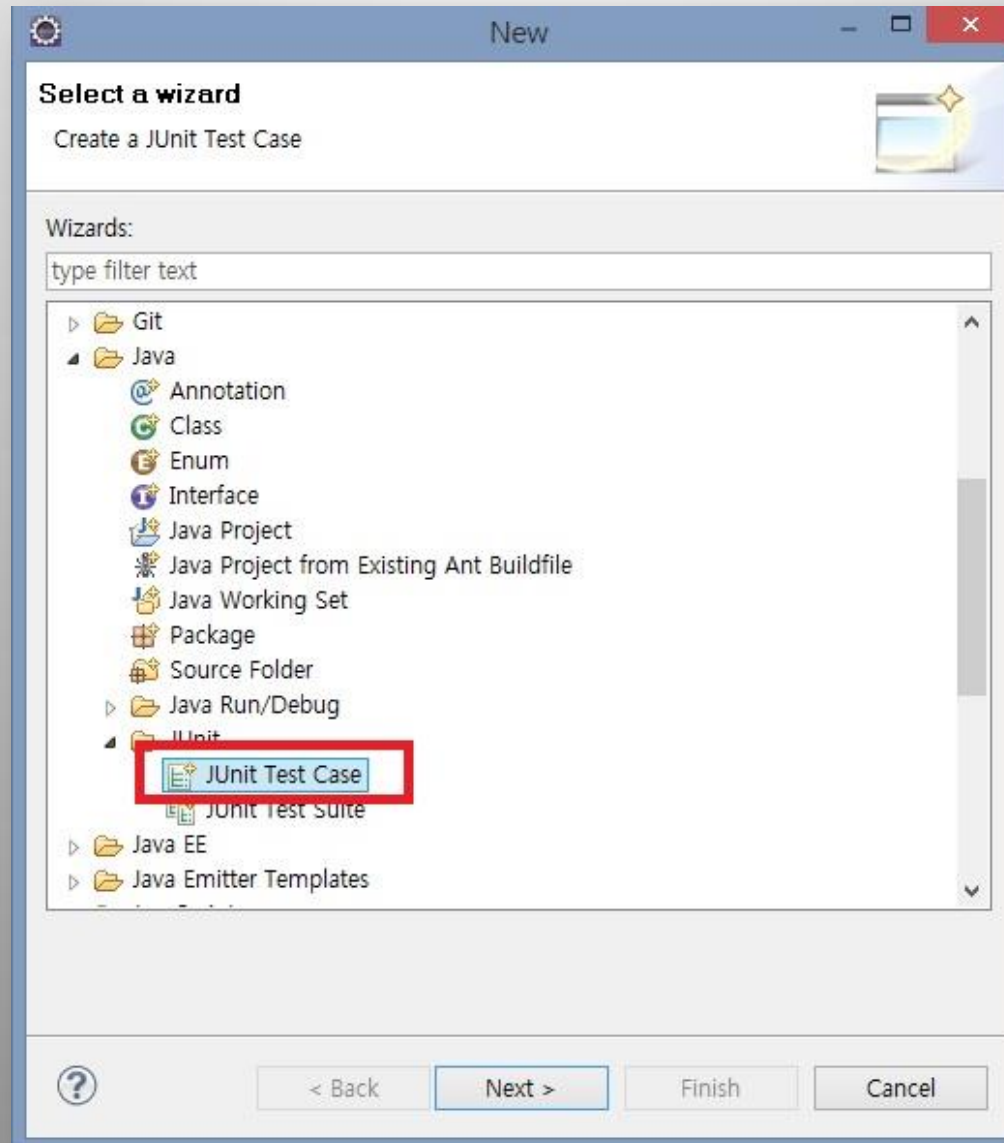
# 03 JUnit 사용법



# 03 JUnit 사용법



# 03 JUnit 사용법



# 03 JUnit 사용법

**JUnit Test Case**

⚠ The use of the default package is discouraged.

New JUnit 3 test  New JUnit 4 test

Source folder: ElevatorProject/src

Package:  (default)

Name:

Superclass: junit.framework.TestCase

Which method stubs would you like to create?

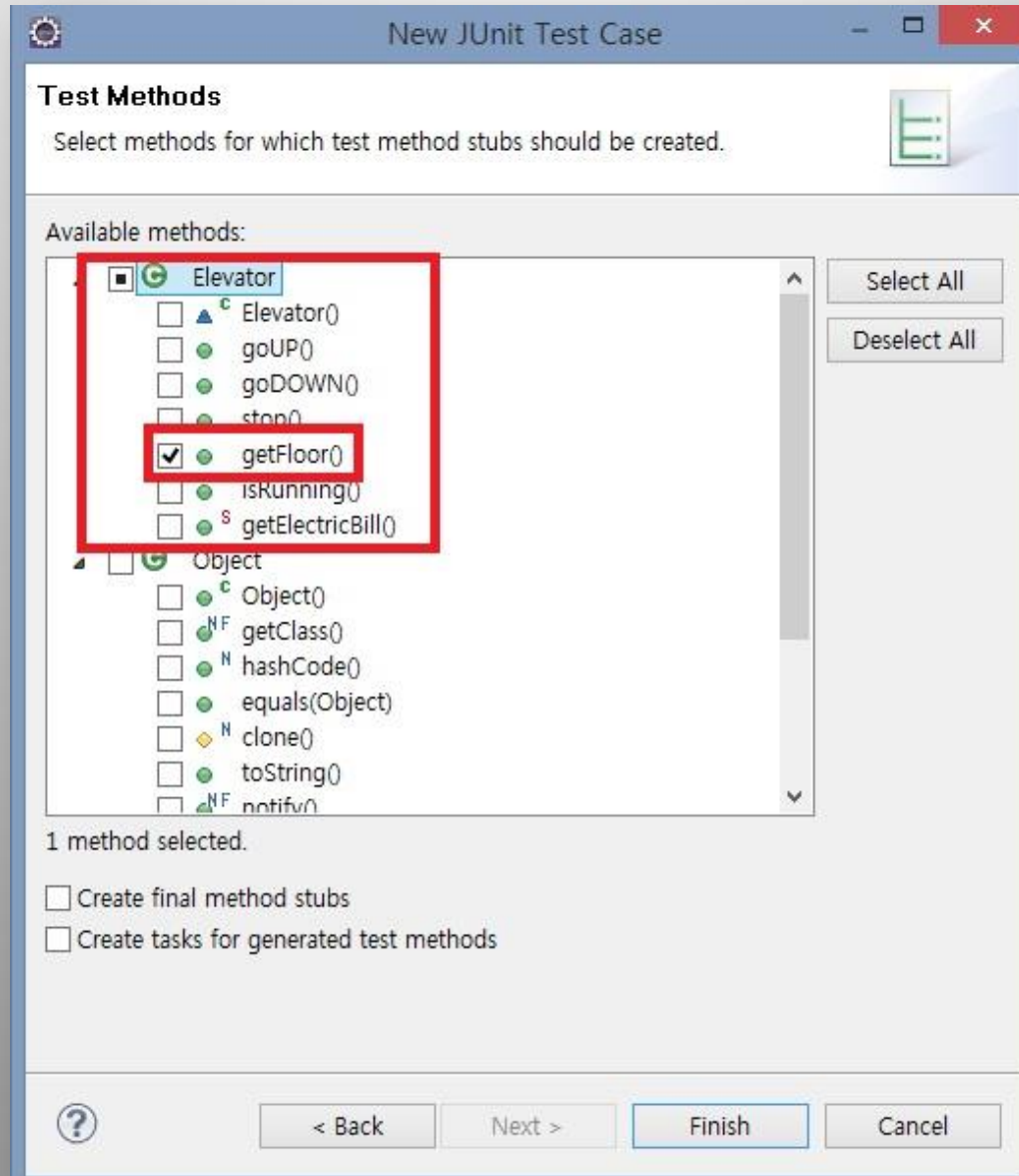
setUpBeforeClass()  tearDownAfterClass()  
 setUp()  tearDown()  
 constructor

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

Generate comments

Class under test:

# 03 JUnit 사용법





# 03 JUnit 사용법

```
import static org.junit.Assert.*;

public class Testing {

    @BeforeClass
    public static void setUpBeforeClass() throws Exception {
    }

    @AfterClass
    public static void tearDownAfterClass() throws Exception {
    }

    @Before
    public void setUp() throws Exception {
    }

    @After
    public void tearDown() throws Exception {
    }

    @Test
    public void testGetFloor() {
        fail("Not yet implemented");
    }
}
```

# 03 JUnit 사용법

The screenshot illustrates the process of running a JUnit test in an IDE. The Project Explorer on the left shows a package structure with a class named 'Testing'. A context menu is open over this class, with 'Run As' selected. The sub-menu for 'Run As' is visible, showing 'JUnit Test' as the chosen option. The background editor shows the source code of 'Testing.java' with JUnit annotations: `@BeforeClass`, `@AfterClass`, `@Before`, `@After`, and `@Test`. The test runner output at the bottom indicates the test 'testGetFloor' passed successfully.

```
import static org.junit.Assert.*;

public class Testing {

    @BeforeClass
    public static void setUpClass() {
    }

    @AfterClass
    public static void tearDownClass() {
    }

    @Before
    public void setUp() throws Exception {
    }

    @After
    public void tearDown() throws Exception {
    }

    @Test
    public void testGetFloor() {
        fail("Not yet implemented");
    }
}
```

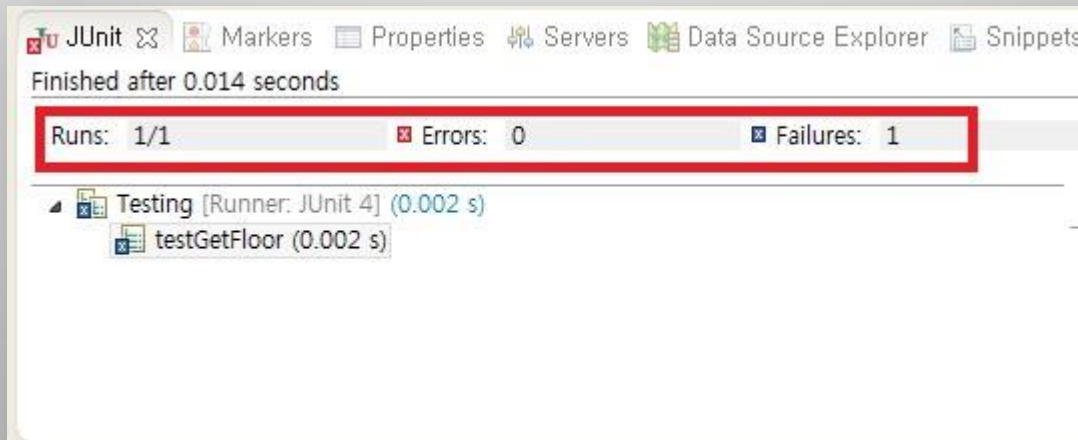
Run As

- 1 Run on Server Alt+Shift+X, R
- 2 JUnit Test Alt+Shift+X, T
- Run Configurations...

Testing [Runner: JUnit 4] (0.002 s)

- testGetFloor (0.002 s)

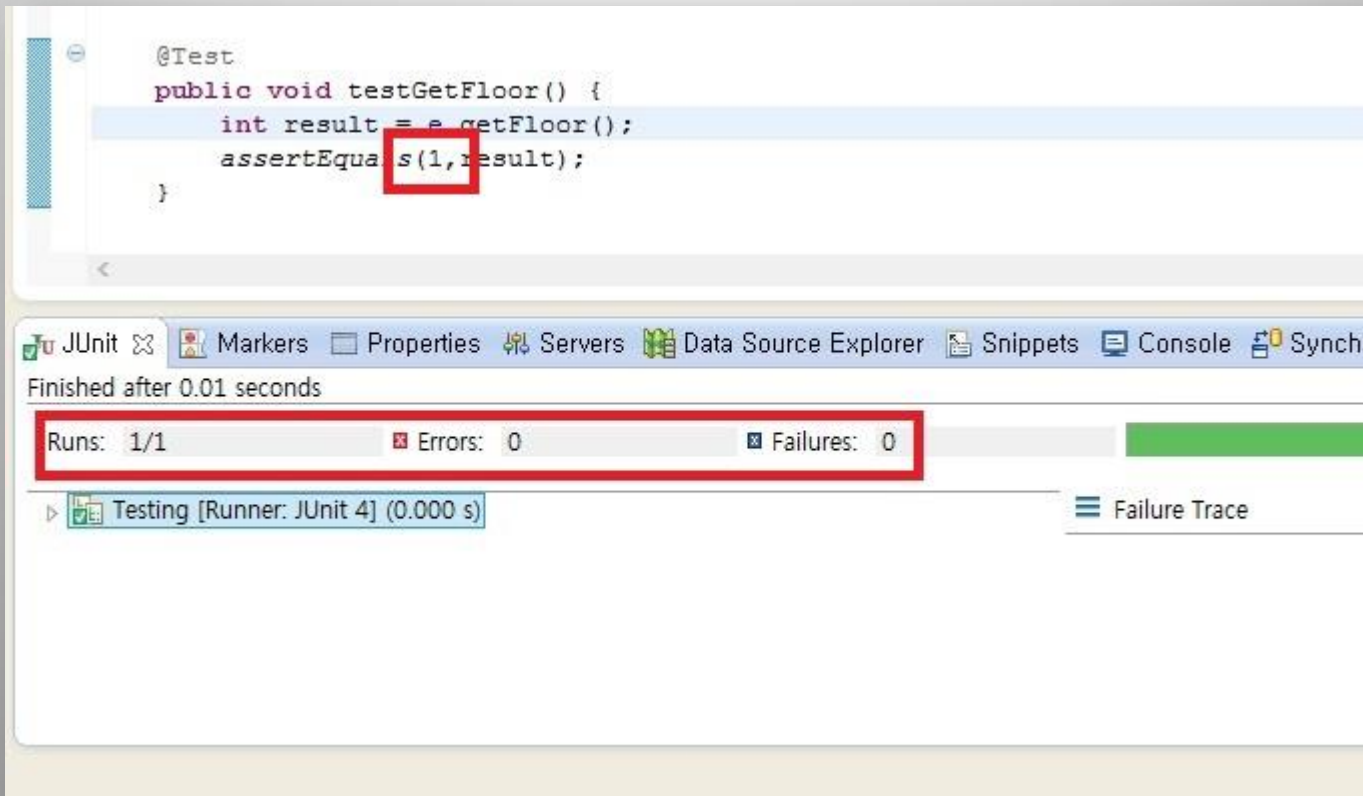
# 03 JUnit 사용법



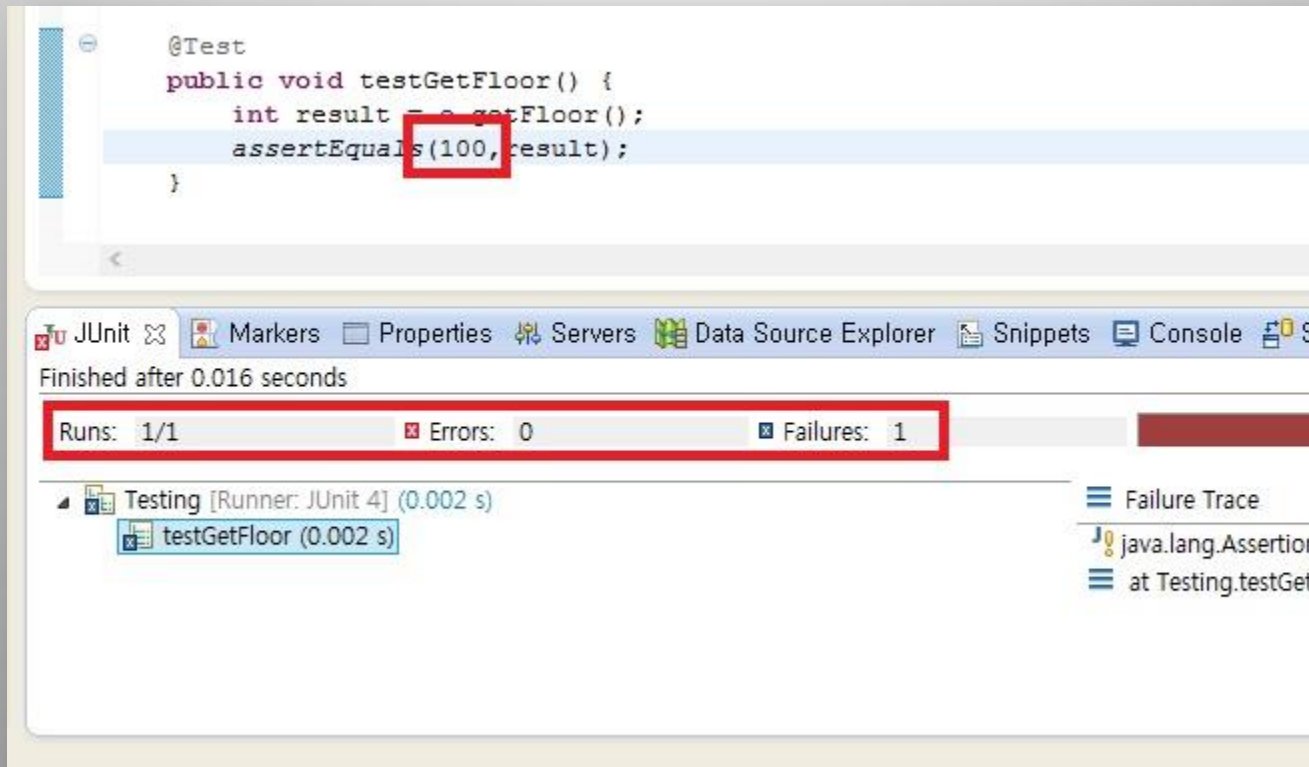
# 03 JUnit 사용법

```
public class Testing {  
  
    Elevator e;  
  
    @BeforeClass  
    public static void setUpBeforeClass() throws Exception {  
    }  
  
    @AfterClass  
    public static void tearDownAfterClass() throws Exception {  
    }  
  
    @Before  
    public void setUp() throws Exception {  
        e = new Elevator();  
    }  
  
    @After  
    public void tearDown() throws Exception {  
    }  
  
    @Test  
    public void testGetFloor() {  
        int result = e.getFloor();  
        assertEquals(1, result);  
    }  
}
```

# 03 JUnit 사용법



# 03 JUnit 사용법



# 03 JUnit 사용법

```
@Test
public void testGetFloor() {
    int result = e.getFloor();
    assertEquals(100, result);
}
```

JUnit 4 | Markers | Properties | Servers | Data Source Explorer | Snippets | Console | Synchronize

Finished after 0.016 seconds

Runs: 1/1 | Errors: 0 | Failures: 1

Testing [Runner: JUnit 4] (0.002 s)

- testGetFloor (0.002 s)

Failure Trace

```
java.lang.AssertionError: expected:<100> but was:<1>
    at Testing.testGetFloor(Testing.java:34)
```

Thank you!