

# 1st Testing

**Team 2** : "제품 관리 알림기능이 있는" 편의점 정산 프로그램

**Team 5** : 자동분류 자료관리 시스템

T2

200511318 김희재

200511326 박현진

200711436 서영주

200913987 이승효

# Contents.

## 1. Our CTIP Environment

## 2. Tools

- ◆ Sonar
- ◆ AllPairs

## 3. Testing [ Team 2 / Team 5 ]

- ◆ System Testing
- ◆ Component Testing

# Contents.

## 1. Our CTIP Environment

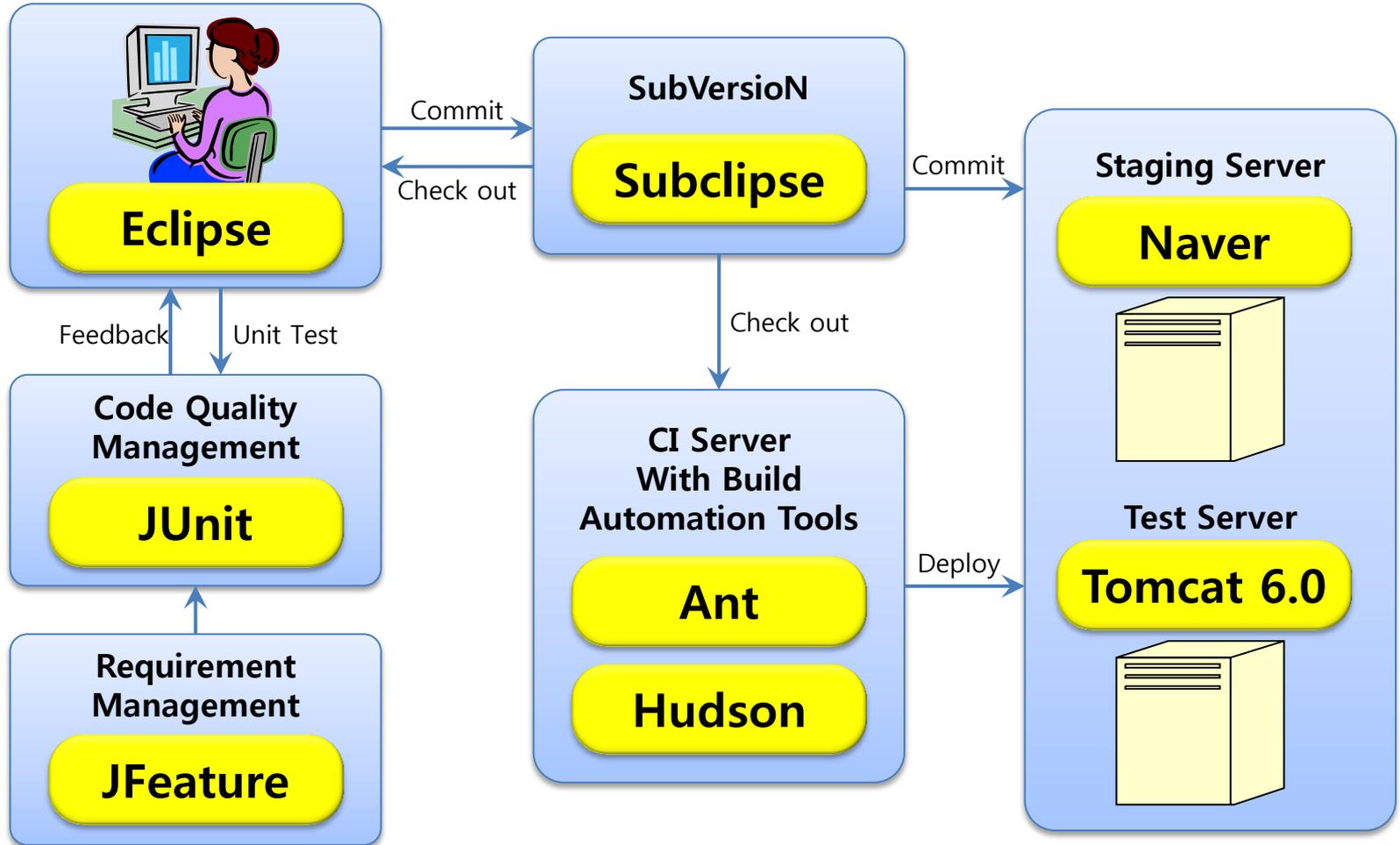
## 2. Tools

- ◆ Sonar
- ◆ AllPairs

## 3. Testing [ Team 2 / Team 5 ]

- ◆ System Testing
- ◆ Component Testing

# CTIP Environment



# Tools of CTIP



**Develop Tool**



**Unit Testing Tool**



**RE Tool**



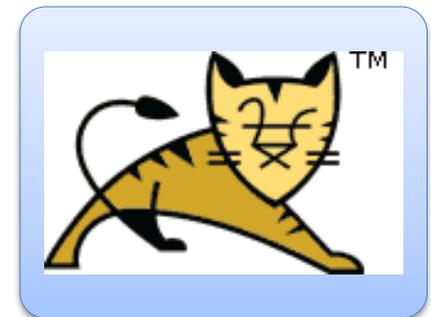
**CM Tool**



**Build Automation**



**CI Server**



**WAS**  
(Web Application Server)

# Contents.

## 1. Our CTIP Environment

## 2. Tools

- ◆ **Sonar**
- ◆ **AllPairs**

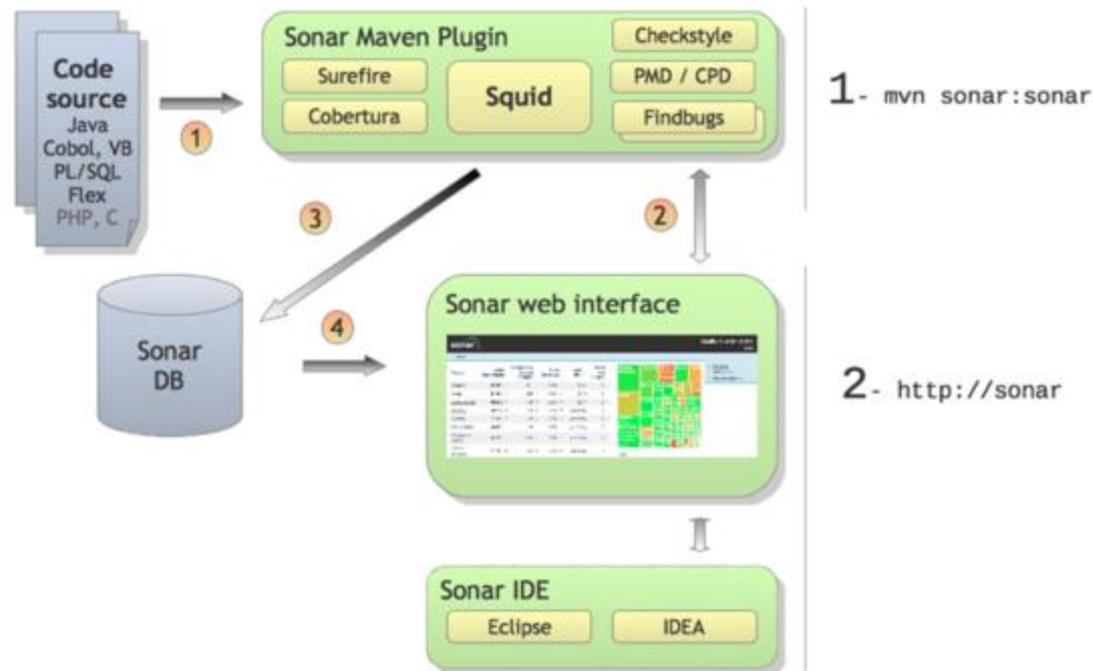
## 3. Testing [ Team 2 / Team 5 ]

- ◆ **System Testing**
- ◆ **Component Testing**

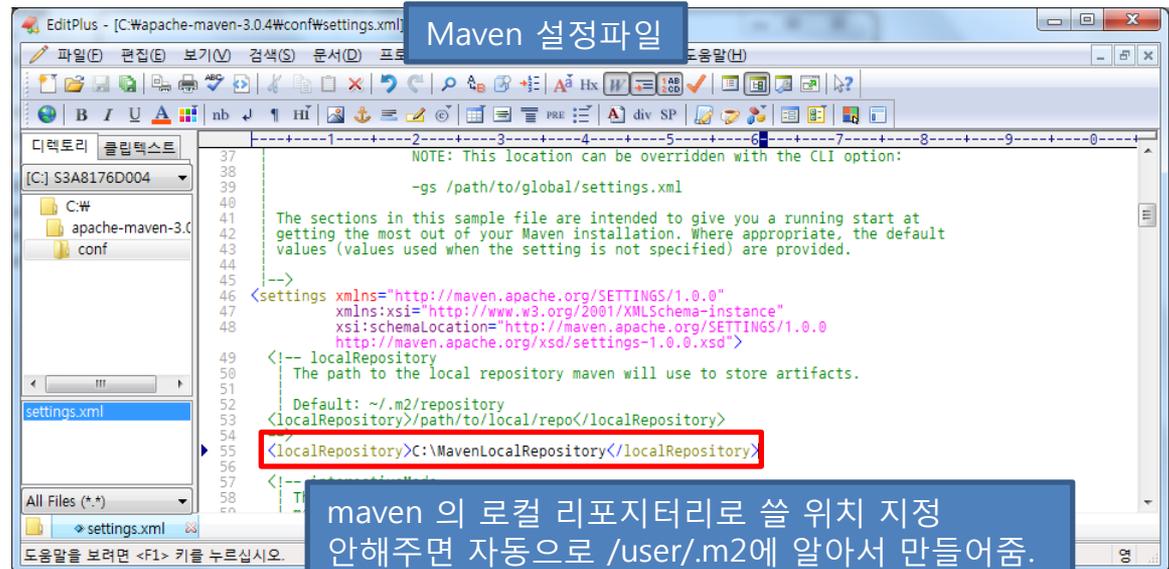
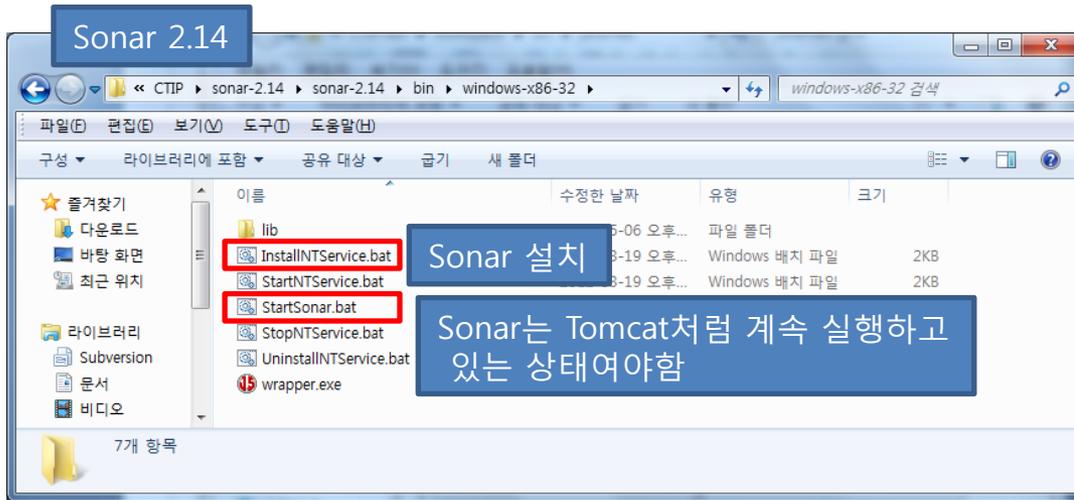
# Sonar Setting

## ◆ Sonar

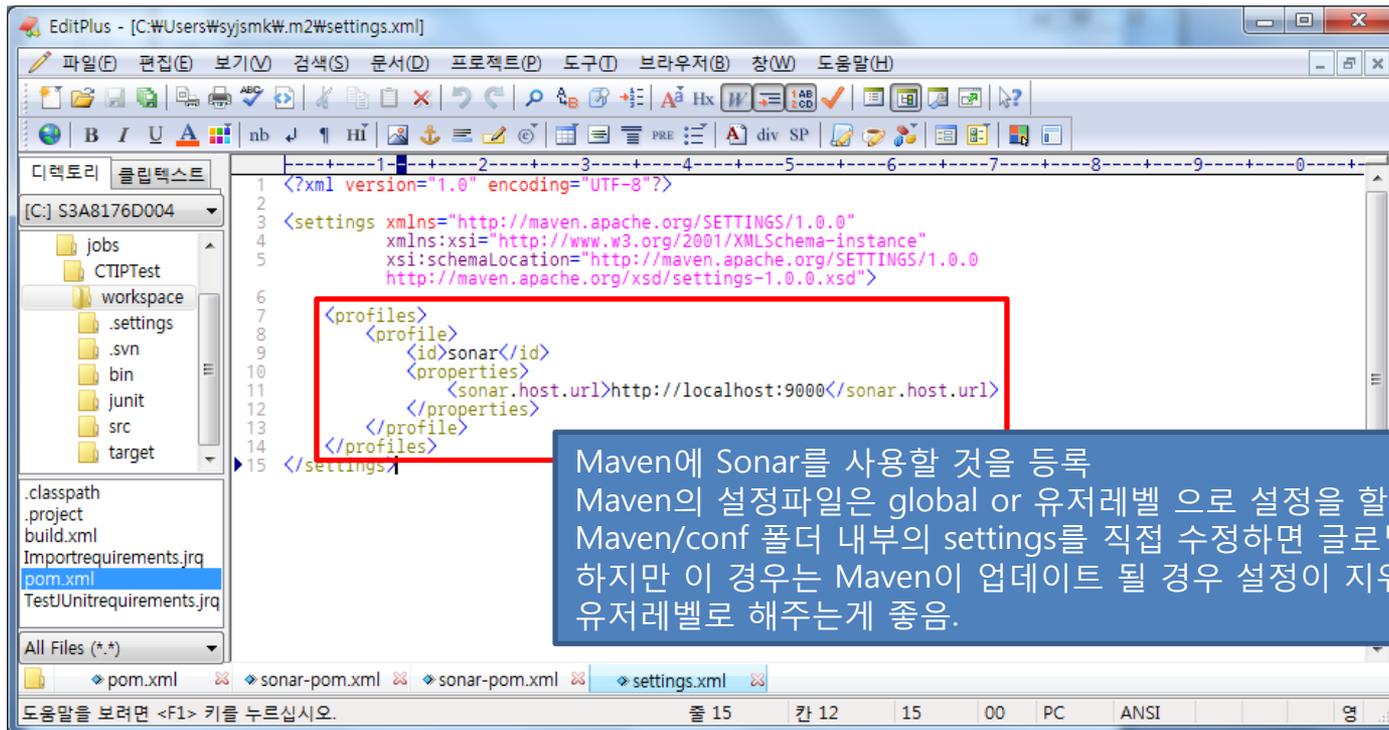
- Codehaus에서 개발한 코드품질관리 플랫폼으로 유명한 오픈소스 툴
- Sonar를 실행시키기 위해서는 maven이 필요하기 때문에, Maven도 같이 설정해줘야 한다.



# Sonar Setting



# Sonar Setting



```
<?xml version="1.0" encoding="UTF-8"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
    http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <profiles>
    <profile>
      <id>sonar</id>
      <properties>
        <sonar.host.url>http://localhost:9000</sonar.host.url>
      </properties>
    </profile>
  </profiles>
</settings>
```

Maven에 Sonar를 사용할 것을 등록  
Maven의 설정파일은 global or 유저레벨 으로 설정을 할 수도 있는데  
Maven/conf 폴더 내부의 settings를 직접 수정하면 글로벌로 적용  
하지만 이 경우는 Maven이 업데이트 될 경우 설정이 지워지기 때문에  
유저레벨로 해주는게 좋음.

# Sonar Setting

```
관리자: C:\windows\system32\cmd.exe - mvn clean install -Dtest=false -DfailIfNoTests=false
gException
[ERROR] [Help 2] http://cwiki.apache.org/confluence/display/MAVEN/ModelParseException
C:\#hudson#jobs#CTIPTest#workspace>mvn clean install -Dtest=false -DfailIfNoTests=false
[INFO] Scanning for projects...
[WARNING]
[WARNING] Some problems were encountered while building the effective model for
syjsmk:CTIPTest:jar:1.0
[WARNING] 'build.plugins.plugin.version' for org.apache.maven.plugins:maven-compiler-plugin is missing. @ line 13, column 20
[WARNING]
[WARNING] It is highly recommended to fix these problems because they threaten the stability of your build.
[WARNING]
[WARNING] For this reason, future Maven versions might no longer support building such malformed projects.
[WARNING]
[INFO]
[INFO] -----
[INFO] Building CTIPTest 1.0
[INFO] -----
Downloading: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/2.4.1/maven-clean-plugin-2.4.1.pom
Downloaded: http://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-c
```

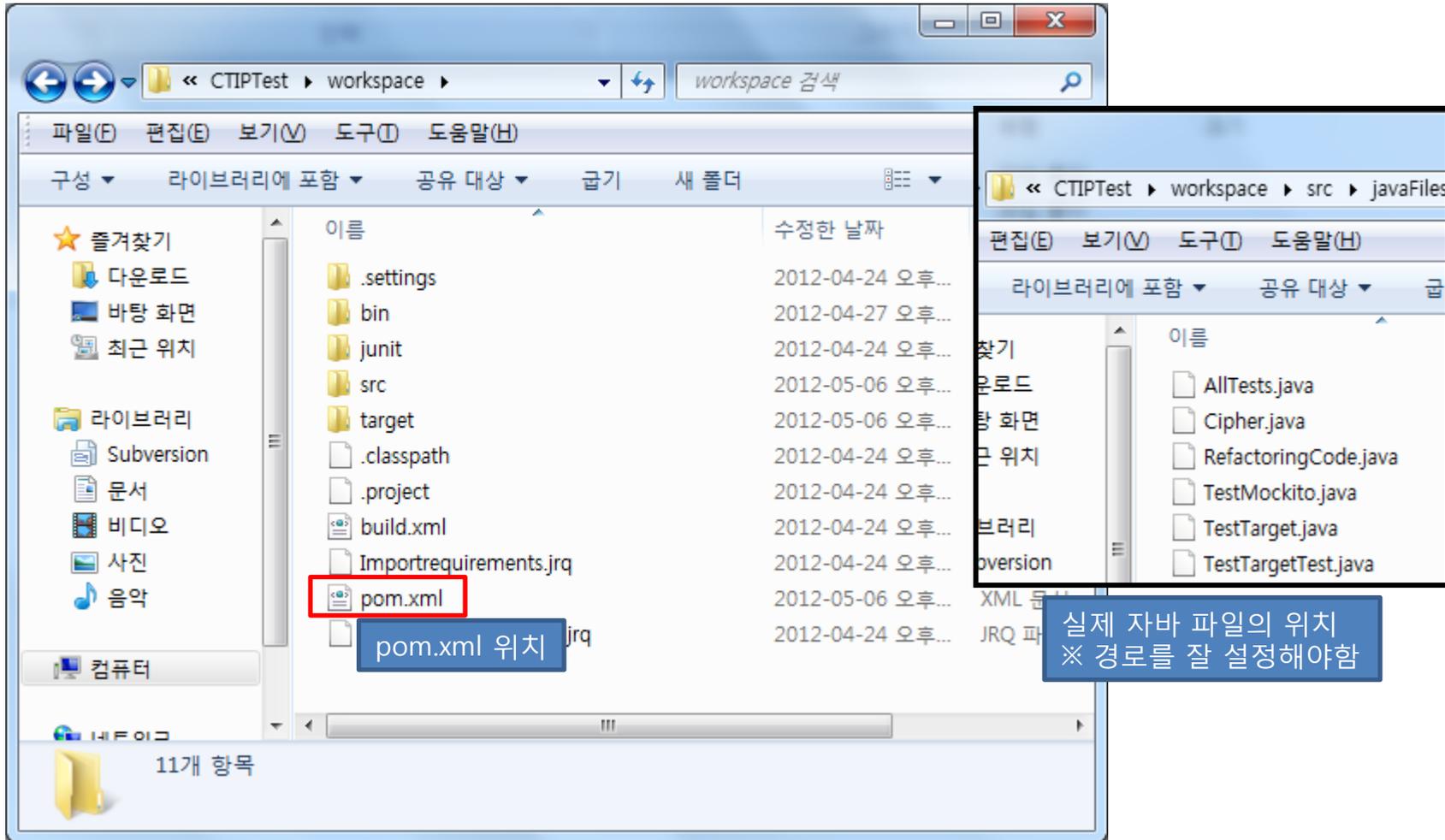
Mvn clean install -Dtest=false -DfailIfNoTests=false  
maven script를 실행하면 maven이 알아서 플러그인들을 가져오면서 스크립트를 실행

# Sonar Setting

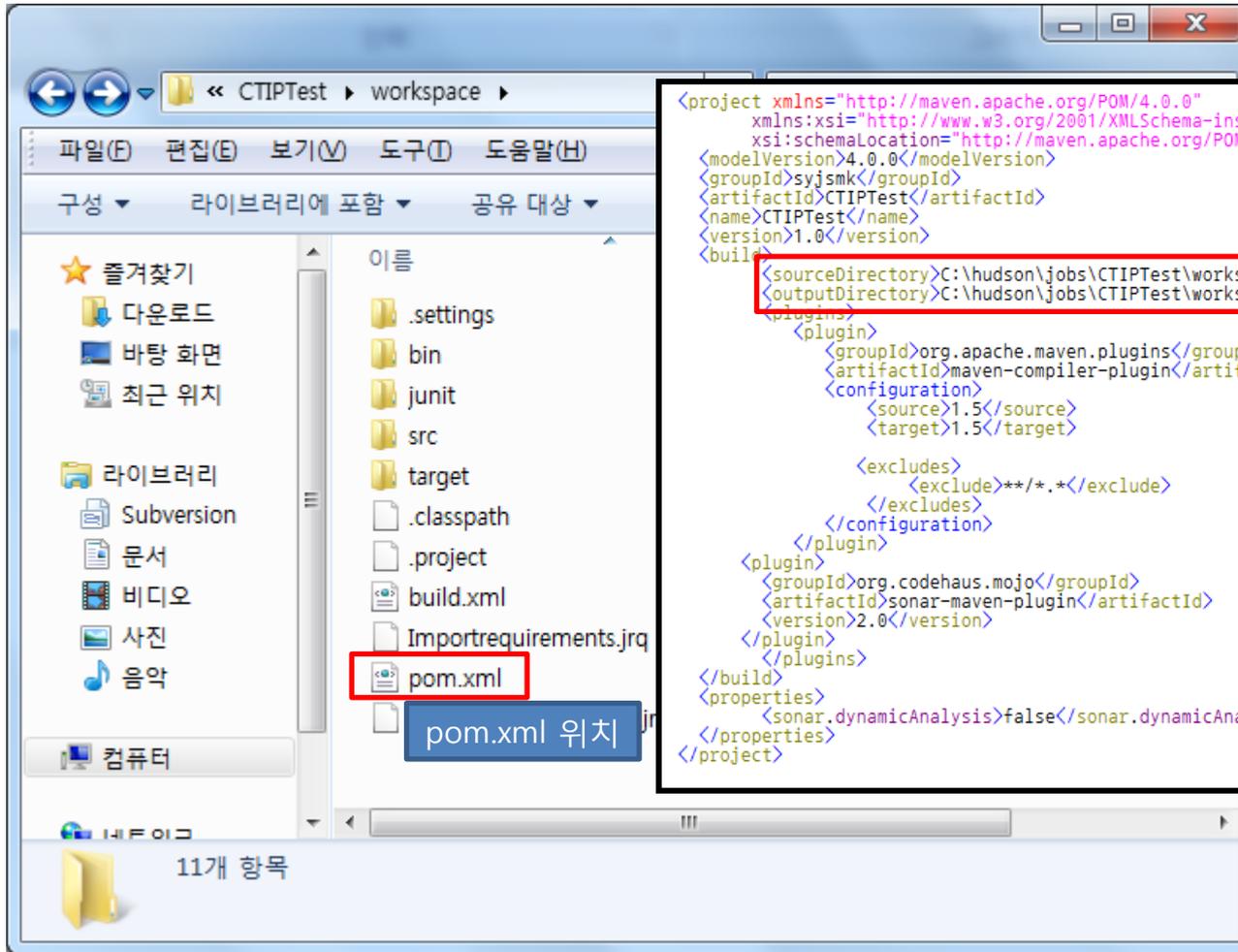
```
관리자: C:\windows\system32\cmd.exe
Downloading: http://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus/1.0.8/plexus-1.0.8.pom
Downloaded: http://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus/1.0.8/plexus-1.0.8.pom (8 KB at 11.0 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-container-default/1.0-alpha-8/plexus-container-default-1.0-alpha-8.pom
Downloaded: http://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-container-default/1.0-alpha-8/plexus-container-default-1.0-alpha-8.pom (8 KB at 11.7 KB/sec)
Downloading: http://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-digest/1.0/plexus-digest-1.0.jar
Downloaded: http://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-digest/1.0/plexus-digest-1.0.jar (12 KB at 21.8 KB/sec)
[INFO] Installing C:\hudson\jobs\CTIPTest\workspace\target\CTIPTest-1.0.jar to C:\MavenLocalRepository\syjsmk\CTIPTest\1.0\CTIPTest-1.0.jar
[INFO] Installing C:\hudson\jobs\CTIPTest\workspace\pom.xml to C:\MavenLocalRepository\syjsmk\CTIPTest\1.0\CTIPTest-1.0.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2:37.955s
[INFO] Finished at: Fri Apr 27 21:18:02 KST 2012
[INFO] Final Memory: 6M/16M
[INFO] -----
C:\hudson\jobs\CTIPTest\workspace >
```

**BUILD SUCCESS**  
maven이 모든 플러그인을 다운받고,  
스크립트를 실행할 수 있는 상태로 완료 함.

# Sonar Setting



# Sonar Setting

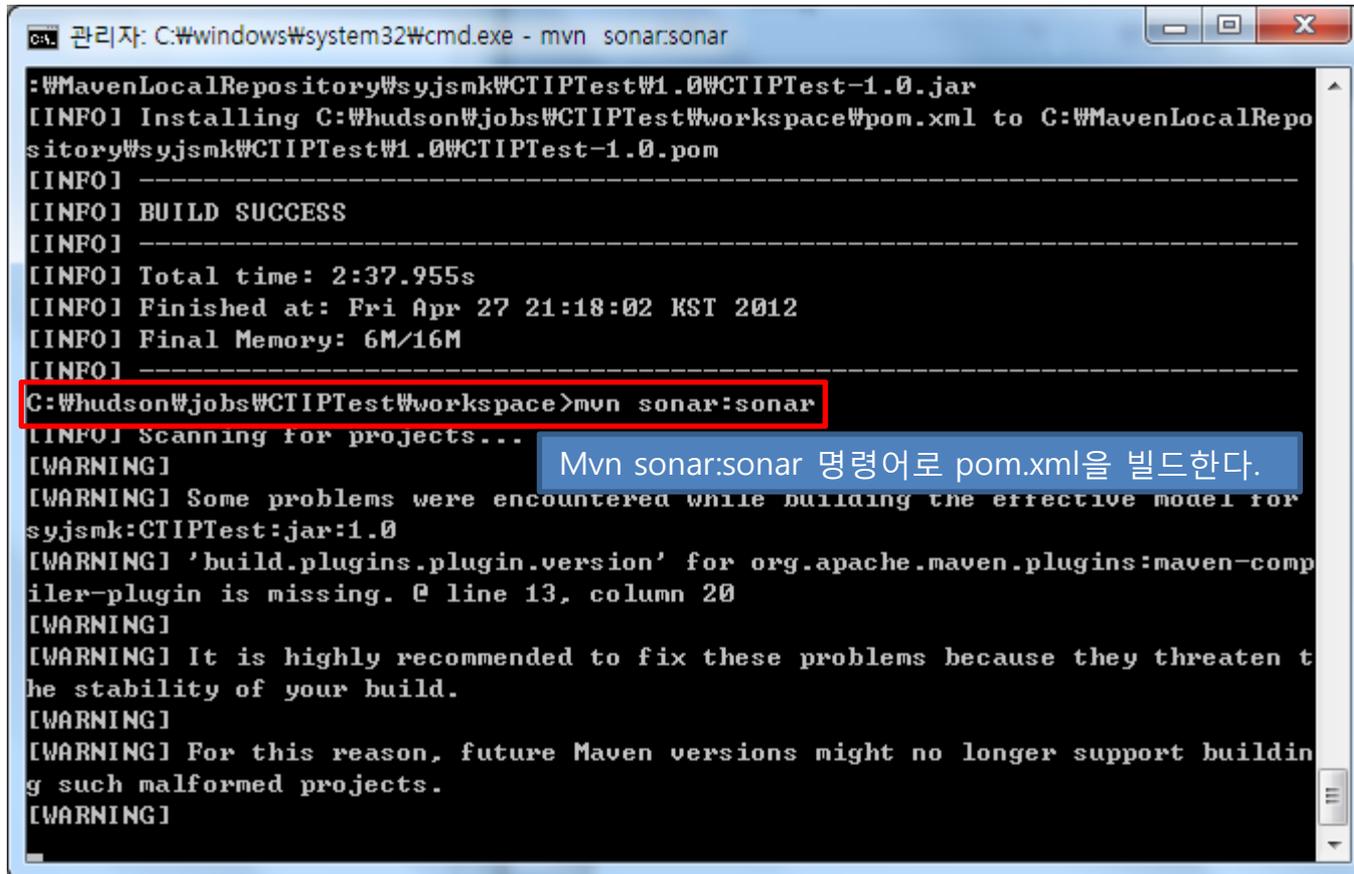


```
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>syjsmk</groupId>
  <artifactId>CTIIPTest</artifactId>
  <name>CTIIPTest</name>
  <version>1.0</version>
  <build>
    <sourceDirectory>C:\udson\jobs\CTIIPTest\workspace\src</sourceDirectory>
    <outputDirectory>C:\udson\jobs\CTIIPTest\workspace\src</outputDirectory>
  </build>
  <plugins>
    <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-compiler-plugin</artifactId>
      <configuration>
        <source>1.5</source>
        <target>1.5</target>
        <excludes>
          <exclude>/**/*.*/</exclude>
        </excludes>
      </configuration>
    </plugin>
    <plugin>
      <groupId>org.codehaus.mojo</groupId>
      <artifactId>sonar-maven-plugin</artifactId>
      <version>2.0</version>
    </plugin>
  </plugins>
  </build>
  <properties>
    <sonar.dynamicAnalysis>false</sonar.dynamicAnalysis>
  </properties>
</project>
```

소스코드 위치,  
sonar 결과파일  
출력할 위치 지정.

pom.xml 위치

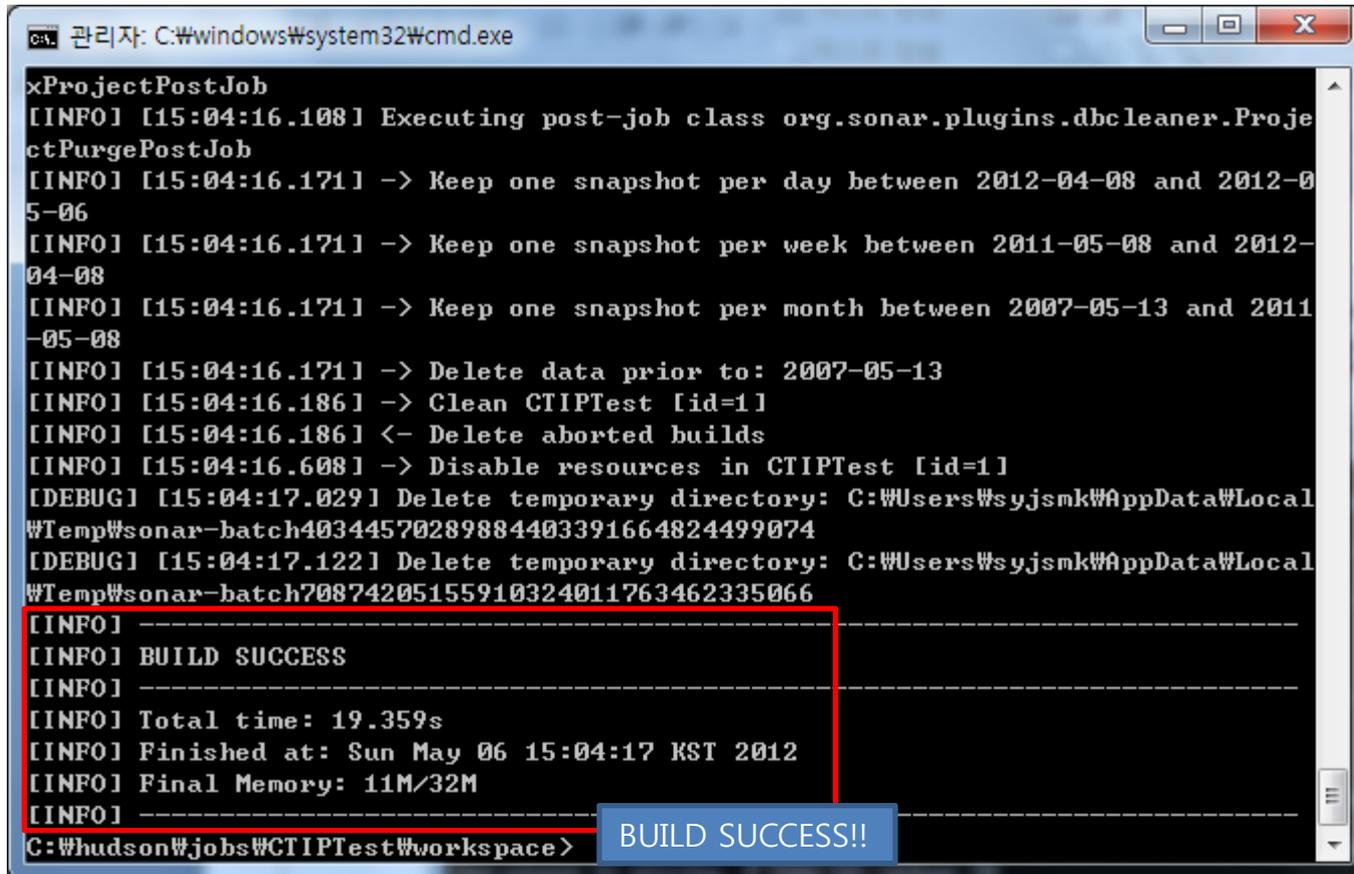
# Sonar Setting



```
관리자: C:\windows\system32\cmd.exe - mvn sonar:sonar
:~\MavenLocalRepository\syjsmk\CTIPTest\1.0\CTIPTest-1.0.jar
[INFO] Installing C:\hudson\jobs\CTIPTest\workspace\pom.xml to C:\MavenLocalRepo
sitory\syjsmk\CTIPTest\1.0\CTIPTest-1.0.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 2:37.955s
[INFO] Finished at: Fri Apr 27 21:18:02 KST 2012
[INFO] Final Memory: 6M/16M
[INFO] -----
C:\hudson\jobs\CTIPTest\workspace>mvn sonar:sonar
[INFO] Scanning for projects...
[WARNING]
[WARNING] Some problems were encountered while building the effective model for
syjsmk:CTIPTest:jar:1.0
[WARNING] 'build.plugins.plugin.version' for org.apache.maven.plugins:maven-comp
iler-plugin is missing. @ line 13, column 20
[WARNING]
[WARNING] It is highly recommended to fix these problems because they threaten t
he stability of your build.
[WARNING]
[WARNING] For this reason, future Maven versions might no longer support buildin
g such malformed projects.
[WARNING]
```

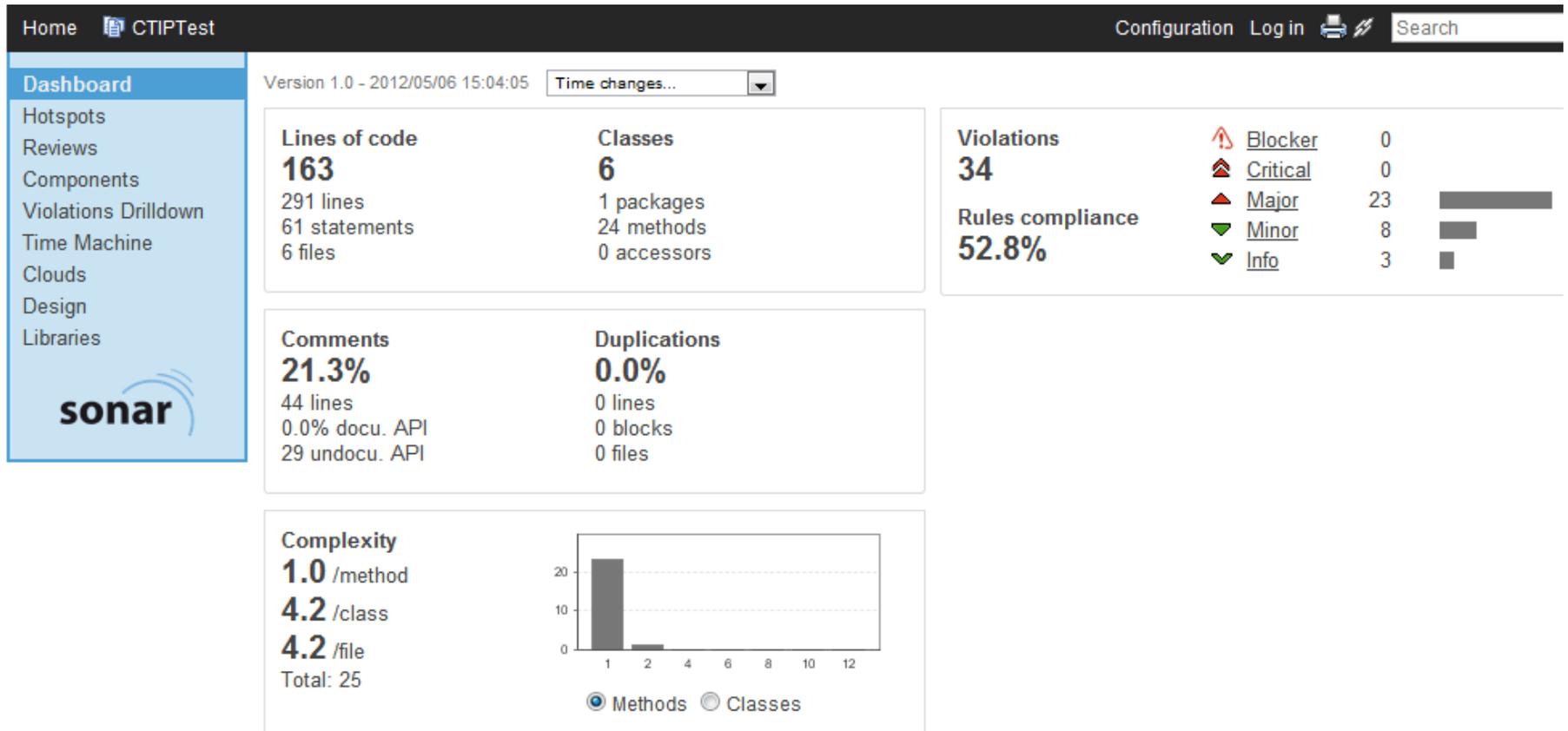
Mvn sonar:sonar 명령어로 pom.xml을 빌드한다.

# Sonar Setting



```
관리자: C:\windows\system32\cmd.exe
xProjectPostJob
[INFO] [15:04:16.108] Executing post-job class org.sonar.plugins.dbcleaner.Proje
ctPurgePostJob
[INFO] [15:04:16.171] -> Keep one snapshot per day between 2012-04-08 and 2012-0
5-06
[INFO] [15:04:16.171] -> Keep one snapshot per week between 2011-05-08 and 2012-
04-08
[INFO] [15:04:16.171] -> Keep one snapshot per month between 2007-05-13 and 2011
-05-08
[INFO] [15:04:16.171] -> Delete data prior to: 2007-05-13
[INFO] [15:04:16.186] -> Clean CTIPTest [id=1]
[INFO] [15:04:16.186] <- Delete aborted builds
[INFO] [15:04:16.608] -> Disable resources in CTIPTest [id=1]
[DEBUG] [15:04:17.029] Delete temporary directory: C:\Users\syjsmk\AppData\Local
\Temp\sonar-batch40344570289884403391664824499074
[DEBUG] [15:04:17.122] Delete temporary directory: C:\Users\syjsmk\AppData\Local
\Temp\sonar-batch70874205155910324011763462335066
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 19.359s
[INFO] Finished at: Sun May 06 15:04:17 KST 2012
[INFO] Final Memory: 11M/32M
[INFO] -----
C:\hudson\jobs\CTIPTest\workspace> BUILD SUCCESS!
```

# Sonar Setting



# Pairwise Testing

## ◆ 필요성

- 많은 결함의 경우 대부분 2개 이상의 파라미터 간의 상호작용에 의해 발생
  - 파라미터 전체의 상호작용을 테스트 범위에 포함시키는 것은 비현실적
- 테스트 노력과 테스트 범위(coverage) 사이에 적절한 Trade-off가 필요

## ◆ *Pairwise Combination Testing*

- 효율적으로 커버할 수 있는 모든 쌍을 생성
- 대부분의 Failure가 2개 요소의 상호작용에 기인
- 모든 조합을 고려해 테스트했을 때 발견할 수 있는 Failure를 모두 발견할 수 있는 것은 아님

# Pairwise Testing – All Pairs

www.satisfice.com/tools.shtml

이 페이지는 영어로 되어 있습니다. 번역 번역 안함 영어 번역 안함 옵션

**SATISFICE, INC.**  
"Epistemology for the rest of us"

## Test Tools

This is a repository of some test tools written, in whole or part, by James Bach. These tools are free and open source. They are offered under the GPL 2.0 license.

**ALLPAIRS Test Case Generation Tool (Version 1.2.1)**  
Download ALLPAIRS (ZIP format; includes a little tutorial and example)

Let's say you must test something that involves a lot of different variables. Computers, different print options, different documents. Who knows what combinations of all variables together, because that would be too many tests. Each value of each of the variables with each value of each other variable at least find a reasonably small set of test cases to satisfy that coverage standard. For variables with ten values each would require 10,000,000,000 test cases. All command-line executable based on a Perl script. Source is included.

### Perlclip

Download PerlClip (ZIP format)

Sometimes you need to test a text field or document with different kinds of strings of the text data. PerlClip is a tool that helps you do that. PerlClip places prepared paste it wherever you need it.

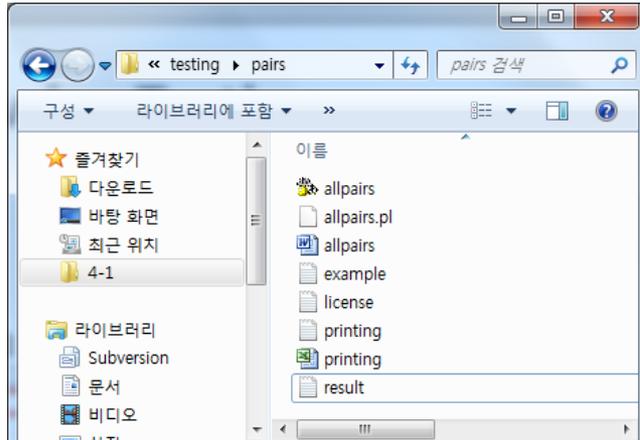
You can run the Perl script, or click on the EXE version (a DOS console window pattern you want to produce. You can enter the following things:

- Any Perl code, such as the following:

```
"james" produces james  
"james" x 5 produces jamesjamesjamesjamesjames
```

www.satisfice.com/tool.shtml 에서 다운로드

# Pairwise Testing – All Pairs



Allpairs.zip 을 사용자가 원하는 위치에 압축 해제

## ① 구하려고 하는 파라미터와 값들을 엑셀에 기입

	A	B	C
1	재생	볼륨	이퀄라이저
2	play	아주크게	Classic
3	stop	크게	Rock
4	순차반복	보통	Hiphop
5	1곡반복	작게	Balad
6	무한반복	아주작게	Dance

Test Case가  $5 * 5 * 5 = 125$ 가지가 나온다.  
이를 Pairwise 조합 기법을 이용해 조합의 수를 줄인다.

## ② 데이터를 메모장에 복사하여 example.txt로 Allpairs틀이 있는 곳에 저장

example - 메모장				
파일(E)	편집(E)	서식(O)	보기(V)	도움말(H)
play		아주크게		Classic
stop		크게		Rock
순차반복		보통		Hiphop
1곡반복		작게		Balad
무한반복		아주작게		Dance

# Pairwise Testing – All Pairs

```

관리자: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Owner>cd C:\Users\Owner\Desktop\4-1\5_SU\testing\pairs
C:\Users\Owner\Desktop\4-1\5_SU\testing\pairs>allpairs example.txt>testcase.txt
C:\Users\Owner\Desktop\4-1\5_SU\testing\pairs>_
    
```

Allpairs가 저장된 폴더로 이동

위와 같은 명령어로 testcase.txt에 페어를 산출한다.

TEST CASES	case	재생	볼륨	이퀄라이저	pairings
1	1	play	아주 크게	Classic	3
2	2	play	크게	Rock	3
3	3	play	보통	Hiphop	3
4	4	play	작게	Balad	3
5	5	play	아주작게	Dance	3
6	6	stop	아주 크게	Rock	3
7	7	stop	크게	Classic	3
8	8	stop	보통	Balad	3
9	9	stop	작게	Hiphop	3
10	10	stop	아주작게	Classic	2
11	11	순차 반복	아주 크게	Hiphop	3
12	12	순차 반복	크게	Balad	3
13	13	순차 반복	보통	Classic	3
14	14	순차 반복	작게	Rock	3
15	15	순차 반복	아주작게	Rock	2
16	16	1곡 반복	아주 크게	Balad	3
17	17	1곡 반복	크게	Hiphop	3
18	18	1곡 반복	보통	Rock	3
19	19	1곡 반복	작게	Classic	3
20	20	1곡 반복	아주작게	Hiphop	2
21	21	무한반복	아주 크게	Dance	3
22	22	무한반복	크게	Classic	2
23	23	무한반복	보통	Rock	2
24	24	무한반복	작게	Hiphop	2
25	25	무한반복	아주작게	Balad	3
26	26	stop	크게	Dance	2
27	27	순차 반복	보통	Dance	2
28	28	1곡 반복	작게	Dance	2

testcase 수정한 날짜: 2012-05-10 오전 1:06  
텍스트 문서 크기: 270바이트

28가지 페어와이즈 조합 생성

125가지 조합  
↓  
28가지 조합

# Contents.

## 1. Our CTIP Environment

## 2. Tools

- ◆ Sonar
- ◆ AllPairs

## 3. Testing [ Team 2 / Team 5 ]

- ◆ System Testing
- ◆ Component Testing

# Testing [Team 2]

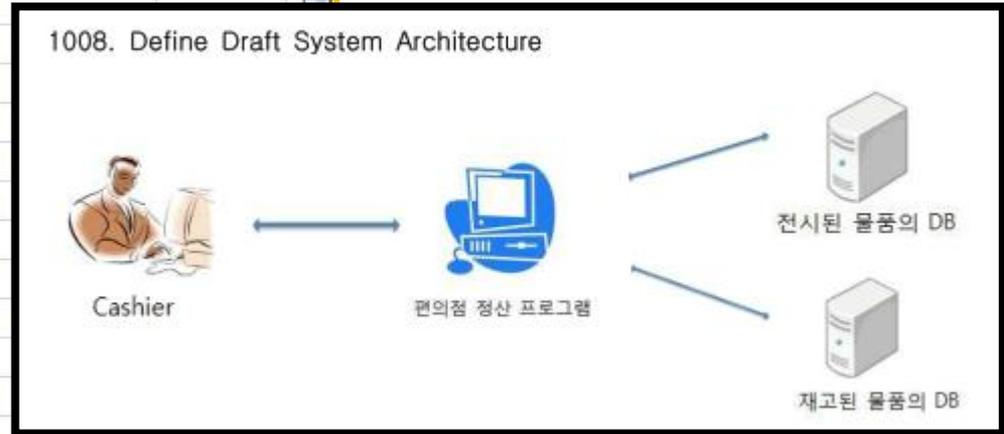
"제품 관리 알림기능이 있는" 편의점 정산 프로그램

# 시나리오 선정

1. 물품관리
2. 장바구니
3. 매출현황

# 물품 정보 입력

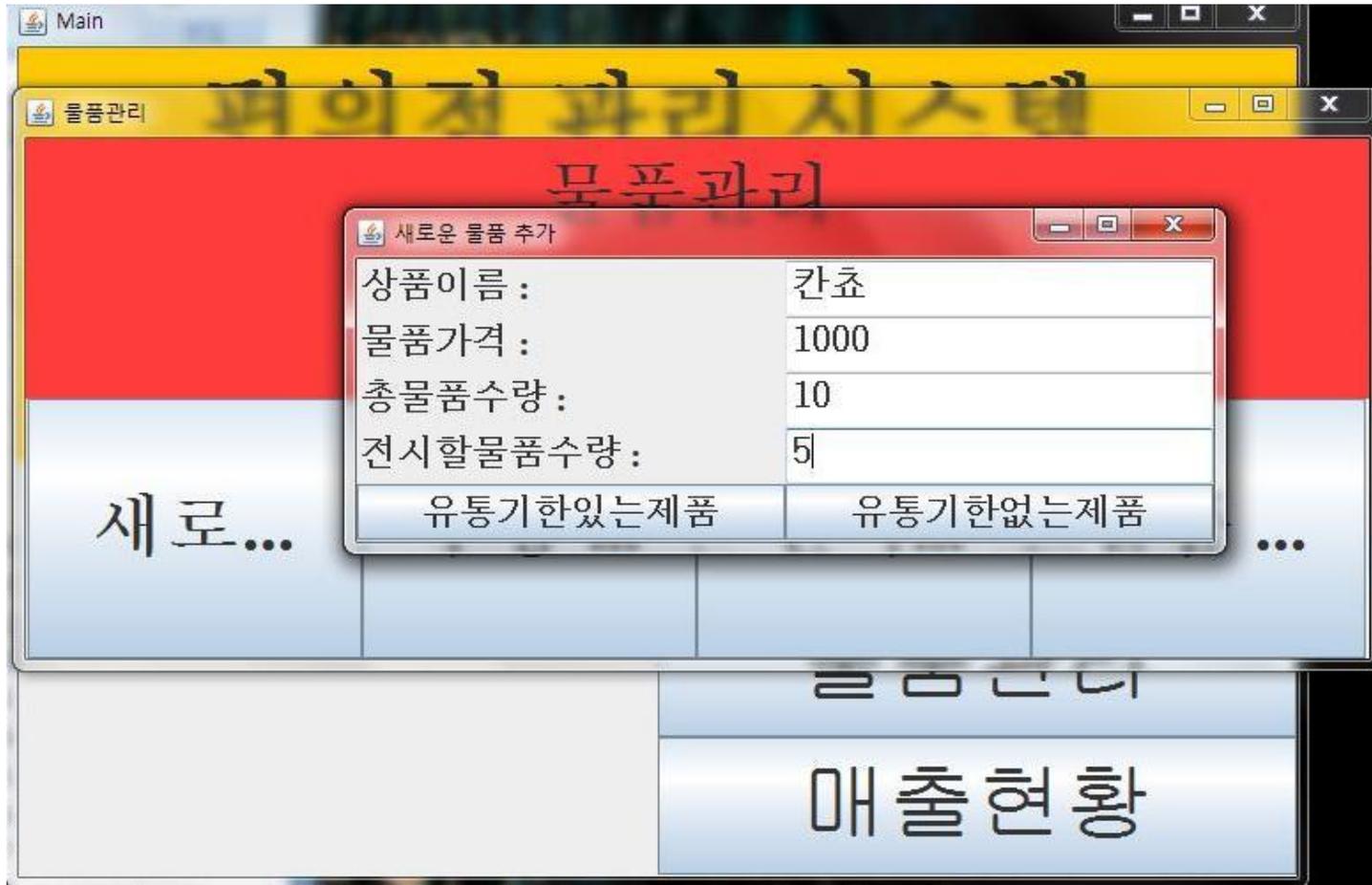
	B	C	D	E	F	G	H
1	물품명	수량	전시	가격	유통기한		
2	칸초	10	5	1000	N/A		
3	오레오	5	2	2000	N/A		
4	비타민워터	6	3	2000	N/A		
5	더위사냥	7	4	1000	N/A		
6	센스민트	4	2	500	N/A		
7	케로로선풍기	3	2	6000	N/A		
8	참봉어빵	5	3	2500	N/A		
9	츄파춥스	1	1	200	N/A		
10	삼다수	20	10	800	N/A		
11	전주비빔밥	8	8	1000	2012050915		
12	참치마요	5	5	1000	2012051015		
13	한줄김밥	10	10	1000	2012051115		
14	부산오뎅	6	6	5000	2012051215		
15	산동만두	4	4	2000	2012051315		
16	김치김밥	5	5	1000	2012051415		
17	오렌지	10	10	3000	2012051515		
18	키위	5	5	2000	2012051615		
19	자몽	20	20	1500	2012051715		
20	딸기	10	10	2000	2012051815		
21	불가리스	5	5	6000	2012051915		



DB가 존재하지 않음.

# ① 물품관리

## ◆ 물품정보 입력



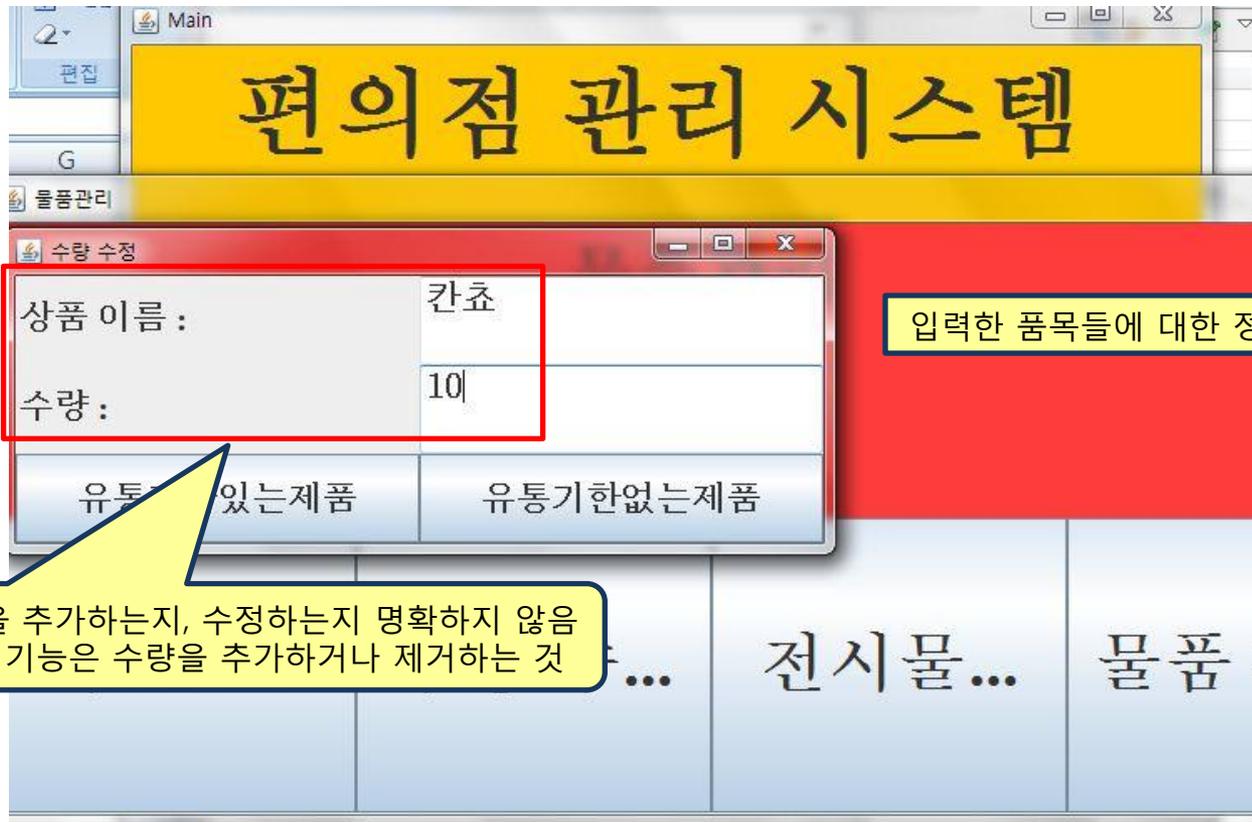
# ① 물품관리

## ◆ 유통기한 입력

The screenshot displays a software interface for inventory management. At the top, a yellow banner reads '편의점 관리 시스템' (Convenience Store Management System). Below it, a red banner reads '물품관리' (Inventory Management). A dialog box titled '유통기한 입력' (Expiration Date Input) is open, showing a text field with '유통기한: 2012050915' and a '확인' (Confirm) button. A yellow callout box points to this dialog with the text '유통기한이 이미 지난 물품도 입력 가능' (Expiration date input possible even for expired items). In the background, a '새로운 물품 추가' (Add New Item) form is visible. It contains fields for '상품명:' (Product Name) with '전주비빔밥', '물품가격:' (Item Price) with '1000', '총물품수량:' (Total Item Quantity) with '5', and '전시할물품수량:' (Quantity to Display) with '5'. A red box highlights the '총물품수량' and '전시할물품수량' fields. A yellow callout box explains: '유통기한이 있는 제품은 총 물품 수량과 전시할 물품 수량이 같으므로 따로 입력 받을 필요가 없다' (For items with an expiration date, the total item quantity and the quantity to display are the same, so there is no need to input them separately). A '물품 ...' (Item ...) button is also visible.

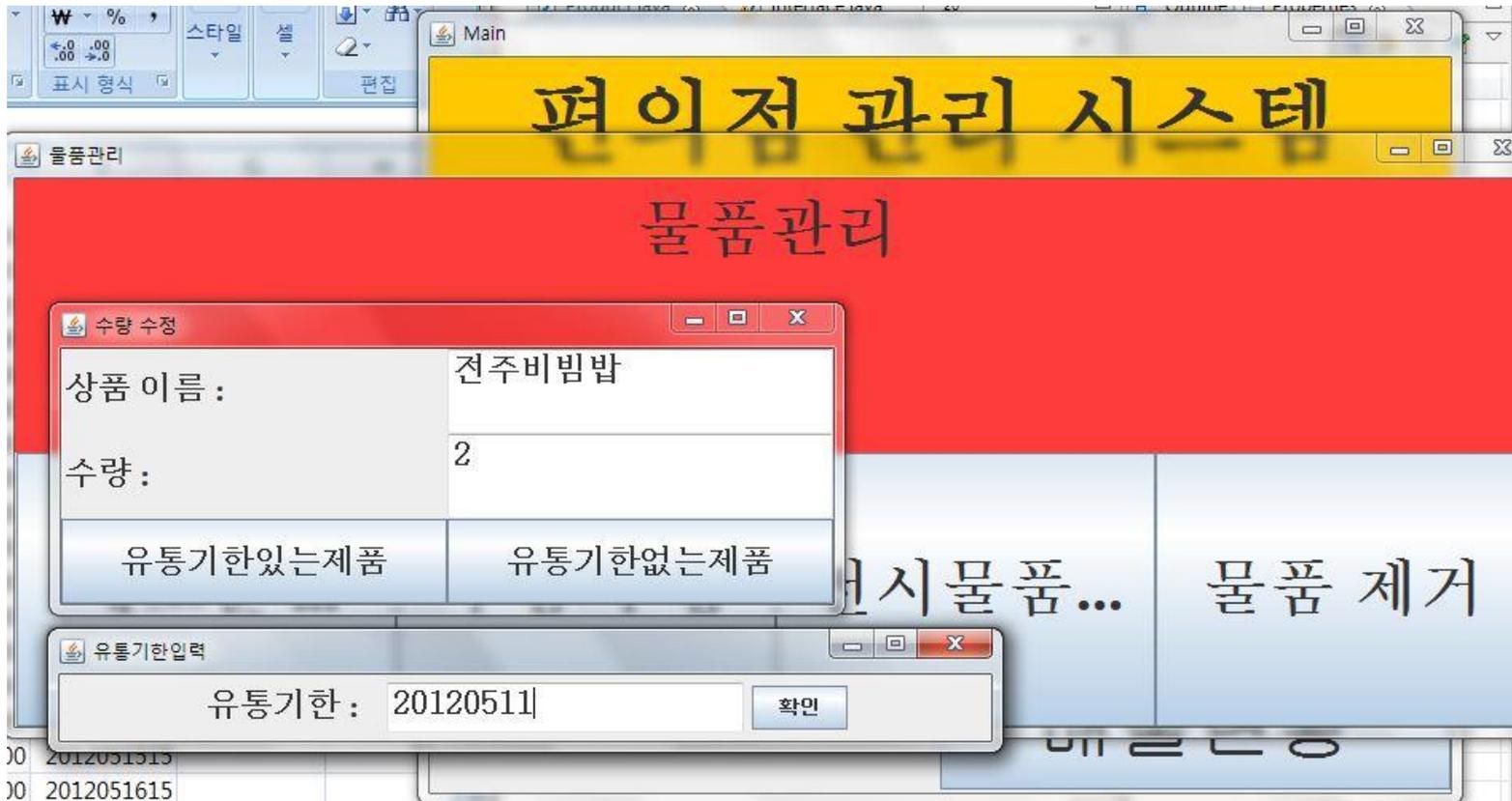
# ① 물품관리

## ◆ 수량 수정 - 유통기한 없는 제품



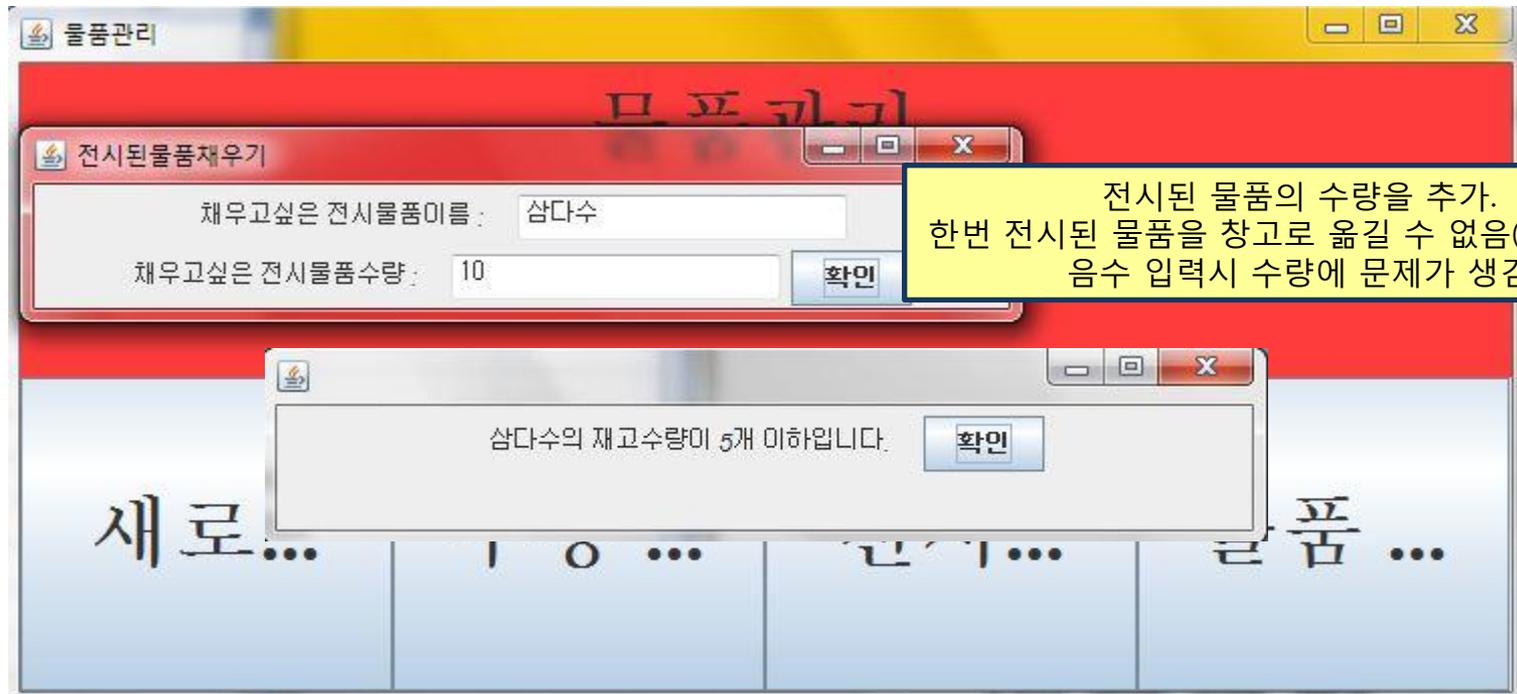
# ① 물품관리

## ◆ 수량 수정 - 유통기한 있는 제품



# ① 물품관리

## ◆ 전시된 물품 수량 조정



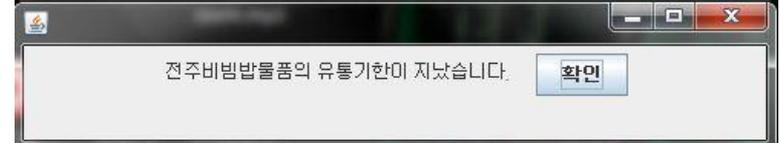
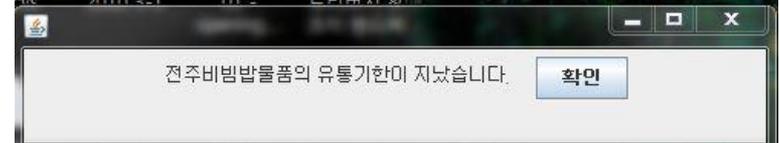
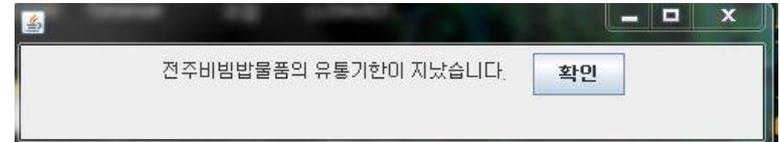
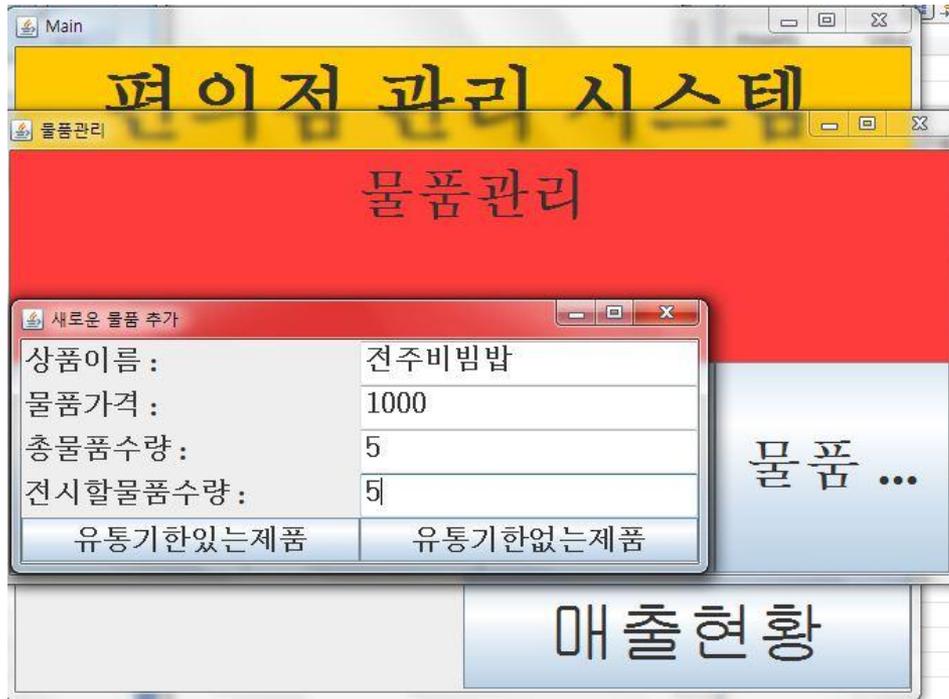
# ① 물품관리

## ◆ 물품 제거



# ① 물품관리

## ◆ 유통기한 알림 기능

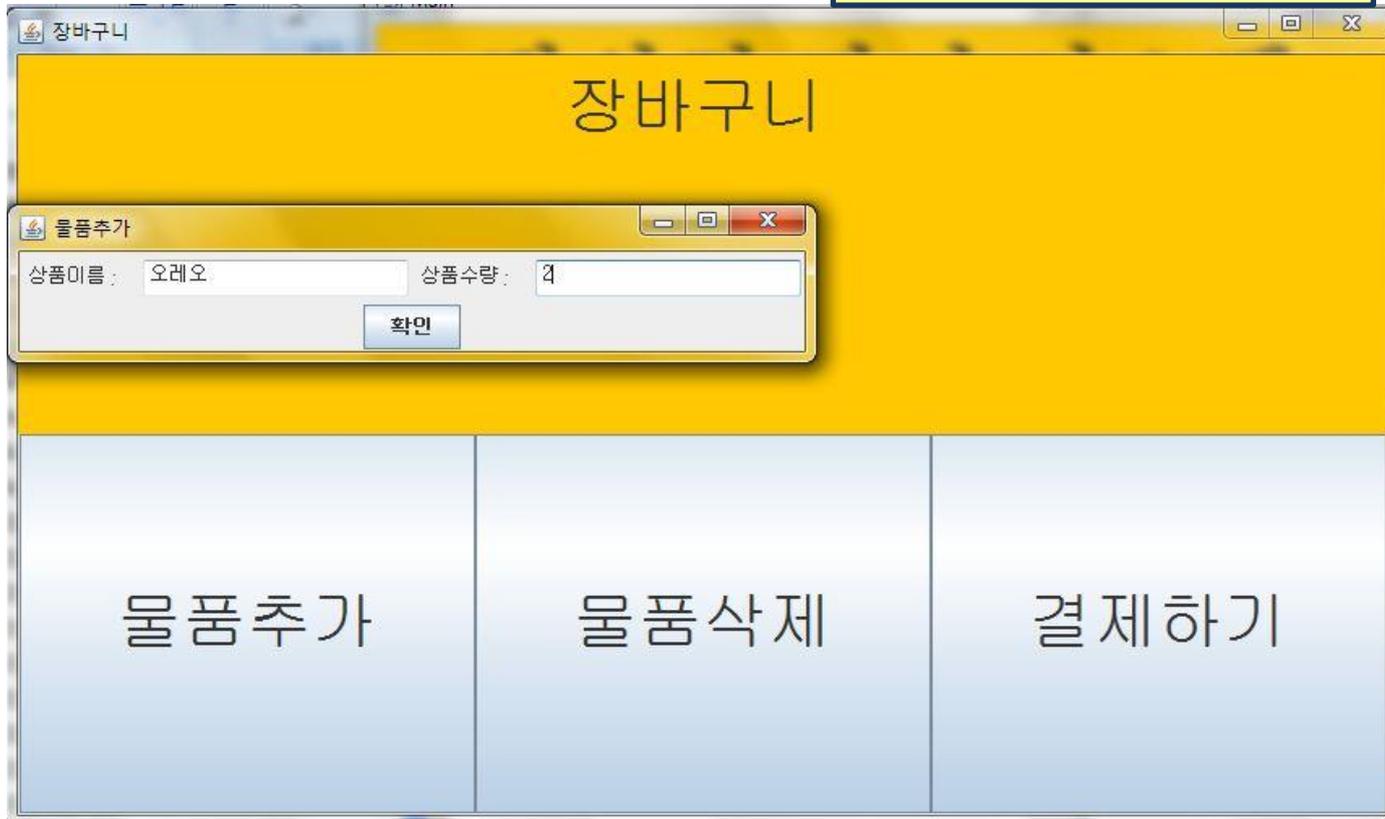


유통기한이 지난 품목 전체를 제거하는게 아니라 일정 시간 단위로 한 품목에 하나씩 제거됨.  
알림 창은 3개까지만 뜬다

# ② 장바구니

## ◆ 물품 추가

장바구니에 뭐가 있는지 알수 없음



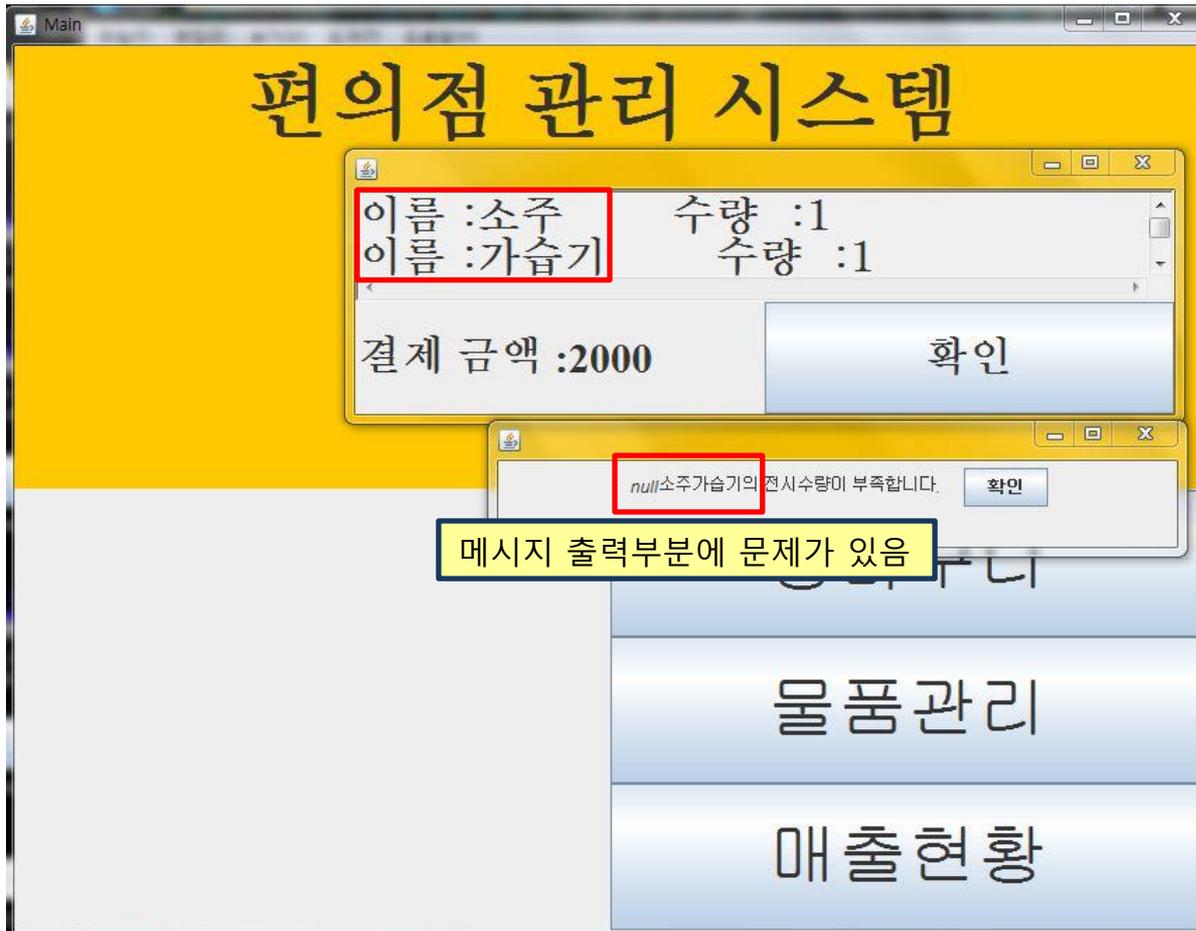
# ② 장바구니

## ◆ 물품 삭제



# ② 장바구니

## ◆ 물품 결제



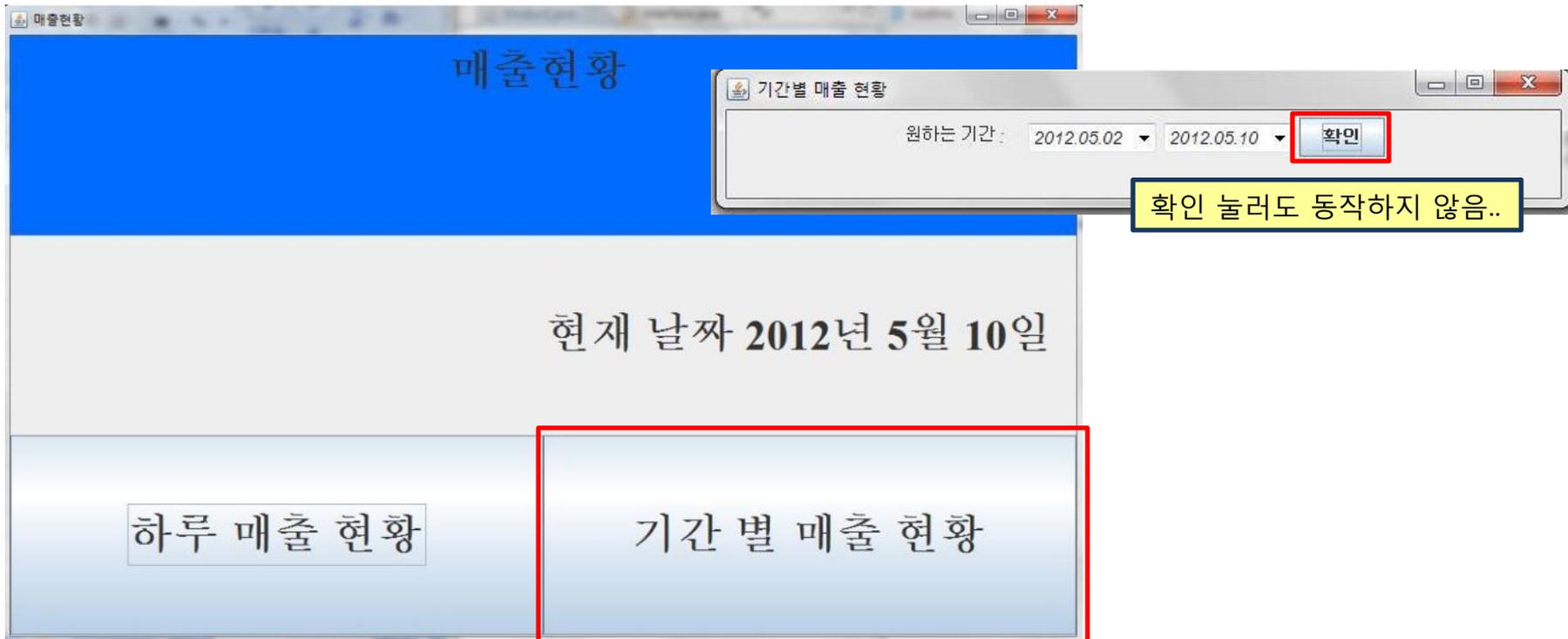
# ③ 매출현황

## ◆ 하루 매출 현황

The screenshot displays a web application interface for '매출현황' (Sales Status). The main content area shows the current date as '현재 날짜 2012년 5월 10일'. At the bottom, there are two buttons: '하루 매출 현황' (Daily Sales Status), which is highlighted with a red border, and '기간 별 매출 현황' (Sales Status by Period). A modal dialog box is open, showing a search field with the text '이름 :더위사냥' and a result area displaying '오늘 하루 총매출 :4000'. Below the result is an '확인' (Confirm) button. A yellow callout box on the right contains the text: '장바구니 구매가 2번 이상 일어나면, 양 매출 현황 조회가 정상적으로 이루어지지 않음' (If the shopping cart purchase occurs more than 2 times, the sales status query will not be performed normally).

# ③ 매출 현황

## ◆ 기간별 매출 현황



# (번외) 프로그램 상태



Windows 작업 관리자

파일(F) 옵션(O) 보기(V) 도움말(H)

응용 프로그램 프로세스 서비스 성능 네트워킹 사용자

이미지 이름	사...	CPU	메모리(개인 ...	설명
eclipse.exe	Hy...	00	146,524 KB	eclipse.exe
iexplore.exe	Hy...	01	83,412 KB	Internet Explorer
javaw.exe	Hy...	00	50,464 KB	Java(TM) Platform SE binary
javaw.exe	Hy...	00	50,188 KB	Java(TM) Platform SE binary
javaw.exe	Hy...	00	42,848 KB	Java(TM) Platform SE binary
javaw.exe	Hy...	00	41,560 KB	Java(TM) Platform SE binary
javaw.exe	Hy...	00	41,348 KB	Java(TM) Platform SE binary

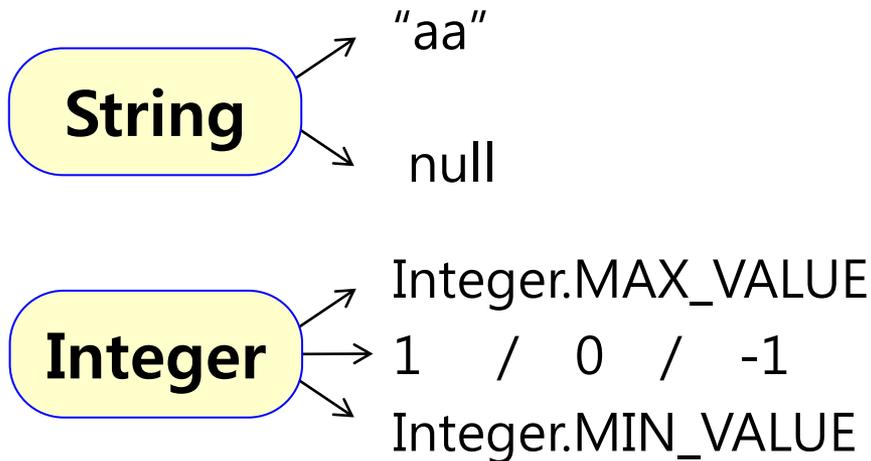
프로그램을 종료 시에도 유통기한이 지났음을 알리는 알림창이 뜨는 것으로 보아, 비정상적으로 종료 되었음을 알 수 있음

모든 사용자의 프로세스 표시(S) 프로세스 끝내기(E)

프로세스: 85 CPU 사용: 13% 실제 메모리: 58%

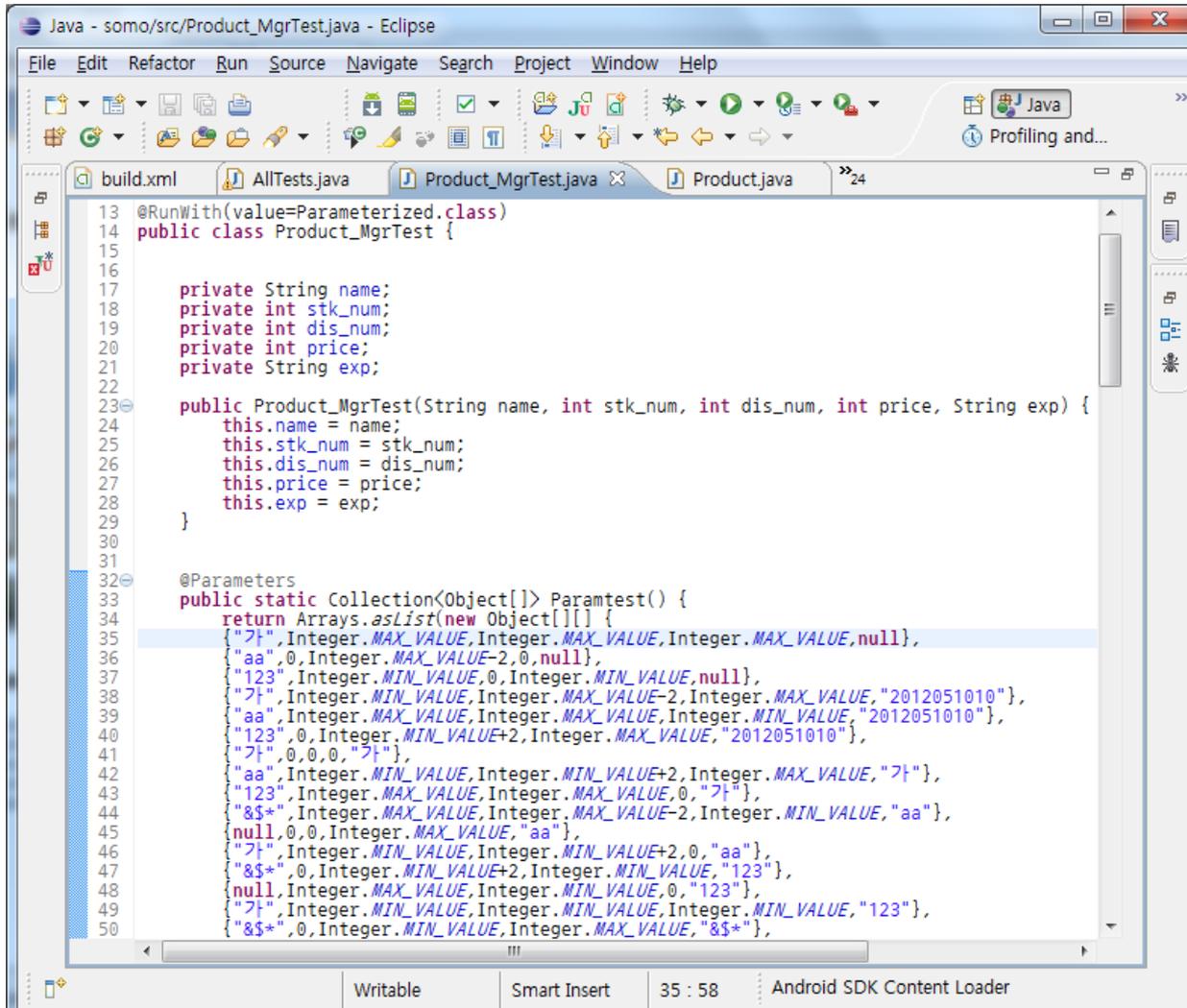
# Representative values

- Representative value는 문서를 기준으로 선정.
- 문서에 명시되어있지 않은 경우에는 각 변수 타입들의 경계값들을 Representative value로 설정



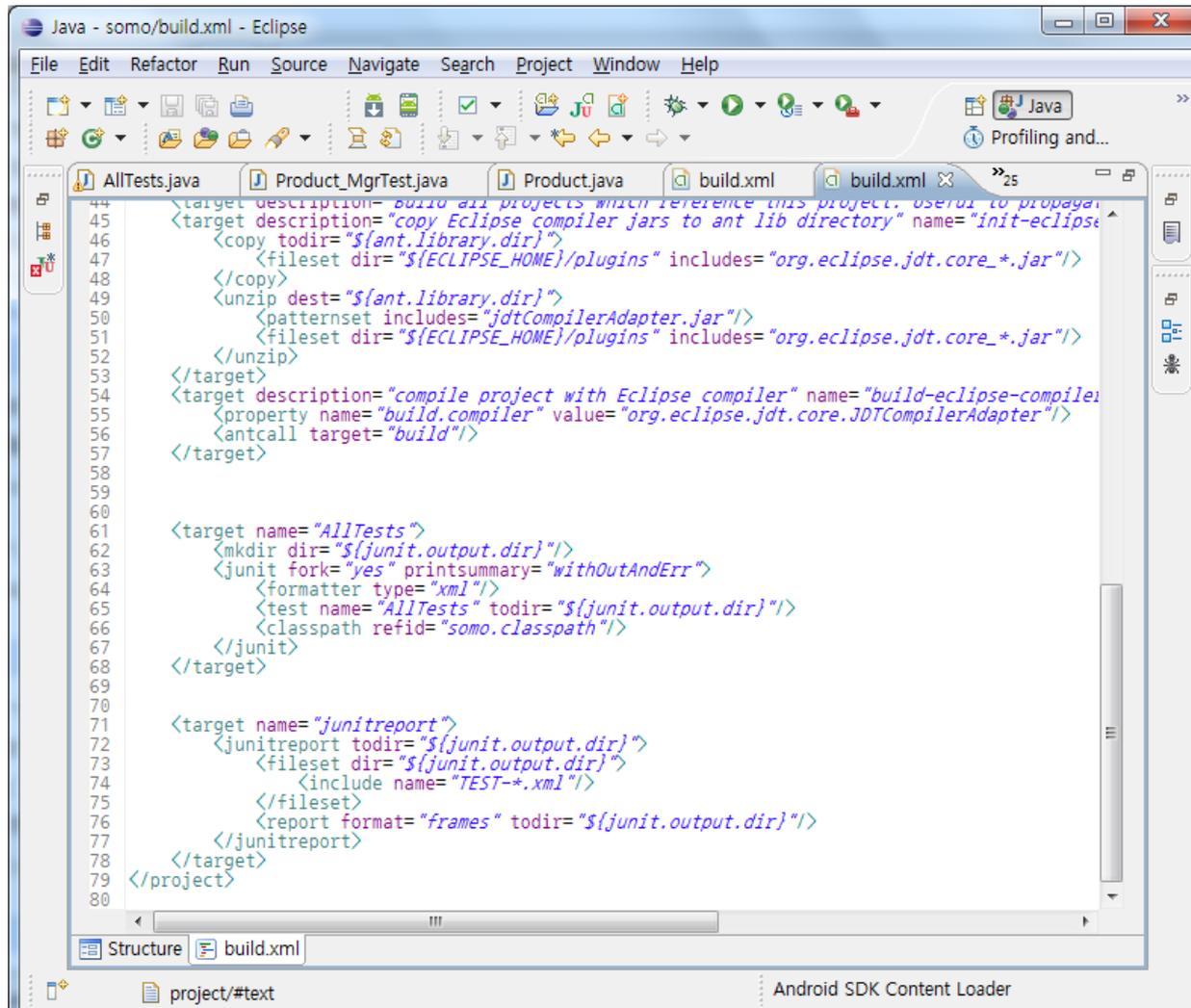
```
@Parameters
public static Collection<Object[]> Paramtest() {
    return Arrays.asList(new Object[][] {
        {"가", Integer.MAX_VALUE, Integer.MAX_VALUE, Integer.MAX_VALUE, null},
        {"aa", 0, Integer.MAX_VALUE-2, 0, null},
        {"123", Integer.MIN_VALUE, 0, Integer.MIN_VALUE, null},
        {"가", Integer.MIN_VALUE, Integer.MAX_VALUE-2, Integer.MAX_VALUE, "2012051010"},
        {"aa", Integer.MAX_VALUE, Integer.MAX_VALUE, Integer.MIN_VALUE, "2012051010"},
        {"123", 0, Integer.MIN_VALUE+2, Integer.MAX_VALUE, "2012051010"},
        {"가", 0, 0, 0, "가"},
        {"aa", Integer.MIN_VALUE, Integer.MIN_VALUE+2, Integer.MAX_VALUE, "가"},
        {"123", Integer.MAX_VALUE, Integer.MAX_VALUE, 0, "가"},
        {"&$*", Integer.MAX_VALUE, Integer.MAX_VALUE-2, Integer.MIN_VALUE, "aa"},
        {"null", 0, 0, Integer.MAX_VALUE, "aa"},
        {"가", Integer.MIN_VALUE, Integer.MIN_VALUE+2, 0, "aa"},
        {"&$*", 0, Integer.MIN_VALUE+2, Integer.MIN_VALUE, "123"},
        {"null", Integer.MAX_VALUE, Integer.MIN_VALUE, 0, "123"},
        {"가", Integer.MIN_VALUE, Integer.MIN_VALUE, Integer.MIN_VALUE, "123"},
        {"&$*", 0, Integer.MIN_VALUE, Integer.MAX_VALUE, "&$*"},
        {"null", Integer.MIN_VALUE, Integer.MAX_VALUE, Integer.MIN_VALUE, "&$*"},
        {"&$*", Integer.MAX_VALUE, 0, 0, "&$*"},
        {"&$*", Integer.MIN_VALUE, Integer.MAX_VALUE, 0, "2012051010"},
        {"null", Integer.MAX_VALUE, Integer.MAX_VALUE-2, Integer.MIN_VALUE, "가"},
        {"aa", 0, Integer.MAX_VALUE, Integer.MAX_VALUE, "123"},
        {"null", Integer.MAX_VALUE, Integer.MIN_VALUE+2, Integer.MAX_VALUE, null},
        {"aa", 0, Integer.MIN_VALUE, Integer.MIN_VALUE, null},
        {"aa", Integer.MAX_VALUE, 0, 0, "2012051010"},
        {"123", Integer.MAX_VALUE, Integer.MIN_VALUE, Integer.MAX_VALUE, "2012051010"},
        {"123", 0, Integer.MAX_VALUE-2, 0, "가"},
        {"&$*", Integer.MIN_VALUE, Integer.MIN_VALUE, Integer.MAX_VALUE, "가"},
        {"123", 0, Integer.MAX_VALUE, Integer.MIN_VALUE, "aa"},
        {"aa", Integer.MIN_VALUE, Integer.MIN_VALUE, 0, "aa"},
        {"123", Integer.MIN_VALUE, Integer.MAX_VALUE-2, Integer.MAX_VALUE, "123"},
        {"가", Integer.MIN_VALUE, Integer.MIN_VALUE+2, Integer.MIN_VALUE, "&$*"},
        {"aa", Integer.MIN_VALUE, Integer.MAX_VALUE-2, Integer.MAX_VALUE, "&$*"},
        {"&$*", Integer.MIN_VALUE, 0, Integer.MIN_VALUE, null},
        {"null", 0, 0, Integer.MIN_VALUE, "2012051010"},
        {"null", Integer.MIN_VALUE, 0, 0, "123"}
    });
}
```

# Junit Code 작성



```
13 @RunWith(value=Parameterized.class)
14 public class Product_MgrTest {
15
16
17     private String name;
18     private int stk_num;
19     private int dis_num;
20     private int price;
21     private String exp;
22
23     public Product_MgrTest(String name, int stk_num, int dis_num, int price, String exp) {
24         this.name = name;
25         this.stk_num = stk_num;
26         this.dis_num = dis_num;
27         this.price = price;
28         this.exp = exp;
29     }
30
31
32     @Parameters
33     public static Collection<Object[]> Paramtest() {
34         return Arrays.asList(new Object[][] {
35             {"가", Integer.MAX_VALUE, Integer.MAX_VALUE, Integer.MAX_VALUE, null},
36             {"aa", 0, Integer.MAX_VALUE-2, 0, null},
37             {"123", Integer.MIN_VALUE, 0, Integer.MIN_VALUE, null},
38             {"가", Integer.MIN_VALUE, Integer.MAX_VALUE-2, Integer.MAX_VALUE, "2012051010"},
39             {"aa", Integer.MAX_VALUE, Integer.MAX_VALUE, Integer.MIN_VALUE, "2012051010"},
40             {"123", 0, Integer.MIN_VALUE+2, Integer.MAX_VALUE, "2012051010"},
41             {"가", 0, 0, 0, "가"},
42             {"aa", Integer.MIN_VALUE, Integer.MIN_VALUE+2, Integer.MAX_VALUE, "가"},
43             {"123", Integer.MAX_VALUE, Integer.MAX_VALUE, 0, "가"},
44             {"&$*", Integer.MAX_VALUE, Integer.MAX_VALUE-2, Integer.MIN_VALUE, "aa"},
45             {null, 0, 0, Integer.MAX_VALUE, "aa"},
46             {"가", Integer.MIN_VALUE, Integer.MIN_VALUE+2, 0, "aa"},
47             {"&$*", 0, Integer.MIN_VALUE+2, Integer.MIN_VALUE, "123"},
48             {null, Integer.MAX_VALUE, Integer.MIN_VALUE, 0, "123"},
49             {"가", Integer.MIN_VALUE, Integer.MIN_VALUE, Integer.MIN_VALUE, "123"},
50             {"&$*", 0, Integer.MIN_VALUE, Integer.MAX_VALUE, "&$*"}
51         });
52     }
53 }
```

# Build 파일 생성



```
44 <target description="build all projects which reference this project, useful to propagate
45 <target description="copy Eclipse compiler jars to ant lib directory" name="init-eclipse
46 <copy todir="${ant.library.dir}"
47 <fileset dir="${ECLIPSE_HOME}/plugins" includes="org.eclipse.jdt.core_*.jar"/>
48 </copy>
49 <unzip dest="${ant.library.dir}"
50 <patternset includes="jdtCompilerAdapter.jar"/>
51 <fileset dir="${ECLIPSE_HOME}/plugins" includes="org.eclipse.jdt.core_*.jar"/>
52 </unzip>
53 </target>
54 <target description="compile project with Eclipse compiler" name="build-eclipse-compile
55 <property name="build.compiler" value="org.eclipse.jdt.core.JDTCompilerAdapter"/>
56 <antcall target="build"/>
57 </target>
58
59
60
61 <target name="AllTests">
62 <mkdir dir="${junit.output.dir}"/>
63 <junit fork="yes" printsummary="withOutAndErr">
64 <formatter type="xml"/>
65 <test name="AllTests" todir="${junit.output.dir}"/>
66 <classpath refid="somo.classpath"/>
67 </junit>
68 </target>
69
70
71 <target name="junitreport">
72 <junitreport todir="${junit.output.dir}">
73 <fileset dir="${junit.output.dir}">
74 <include name="TEST-*.xml"/>
75 </fileset>
76 <report format="frames" todir="${junit.output.dir}"/>
77 </junitreport>
78 </target>
79 </project>
80
```

# Hudson에서 Build

**Hudson**  [?](#)

Hudson > [Team2Test](#) 자음 재실행 켜기

[Back to Dashboard](#)

[Status](#)

[Changes](#)

[Workspace](#)

[Build Now](#)

[Delete Project](#)

[Configure](#)

**Build History** [\(trend\)](#)

- #9 [2012. 5. 11 오전 12:37:26](#)
- #8 [2012. 5. 11 오전 12:02:30](#)
- #7 [2012. 5. 11 오전 12:00:22](#)
- #6 [2012. 5. 10 오후 11:20:44](#)
- #5 [2012. 5. 10 오후 11:19:36](#)
- #4 [2012. 5. 10 오후 11:13:13](#)
- #3 [2012. 5. 10 오후 11:09:09](#)

## Project Team2Test

[Workspace](#)

[Recent Changes](#)

[Latest Test Result](#) (267 failures / +265)

[Latest Console output](#)

### Test Result Trend

[소개 내용 입력](#) [Disable Project](#)

[\(just show failures\)](#) [enlarge](#)

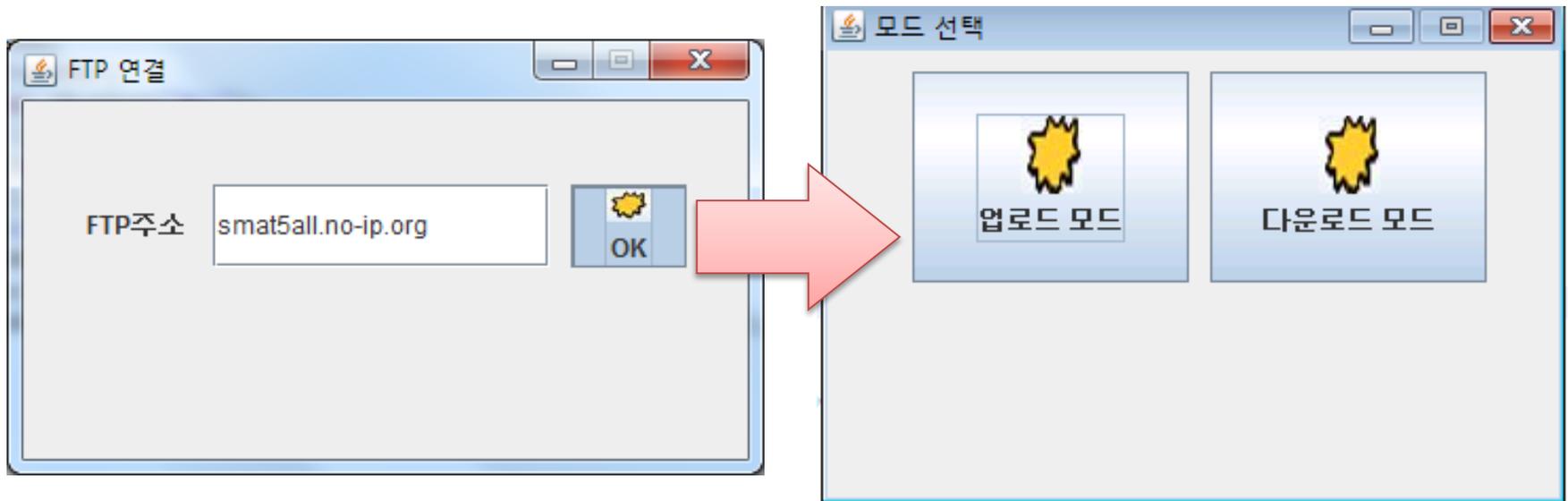
### Permalinks

- [Last build \(#9\), 4 min 3 sec ago](#)
- [Last successful build \(#9\), 4 min 3 sec ago](#)
- [Last failed build \(#7\), 41 min ago](#)

# Testing [Team 5]

“ 자동분류 자료관리 시스템 ”

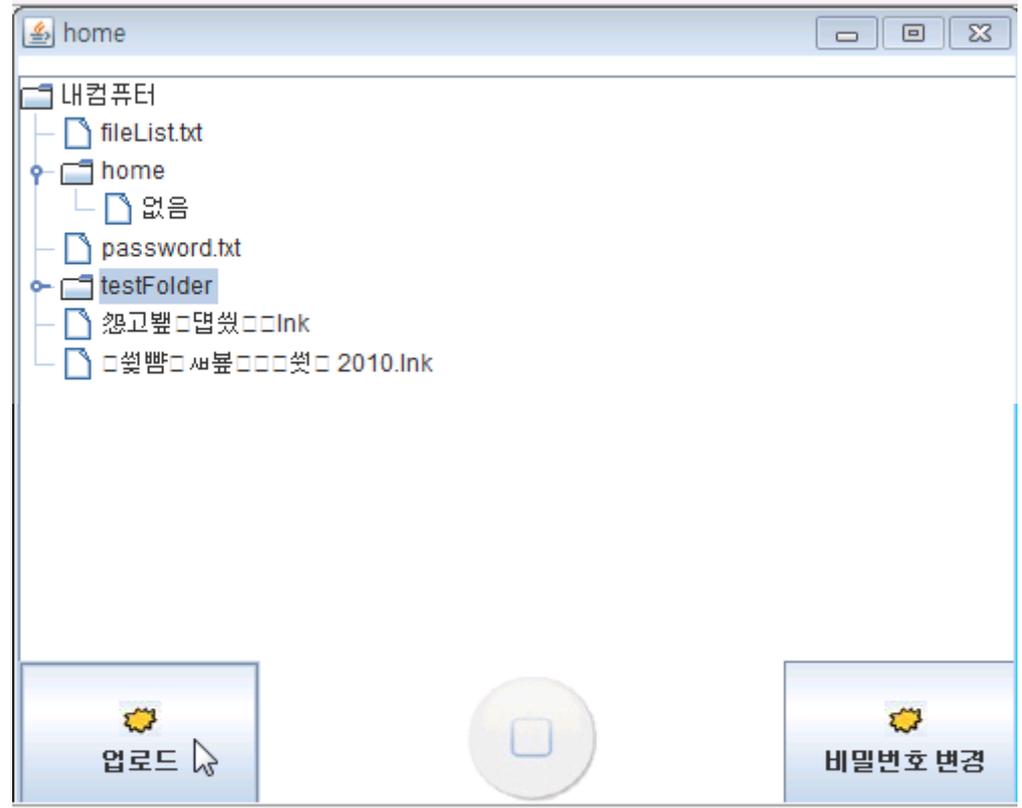
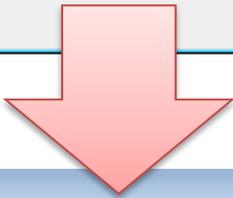
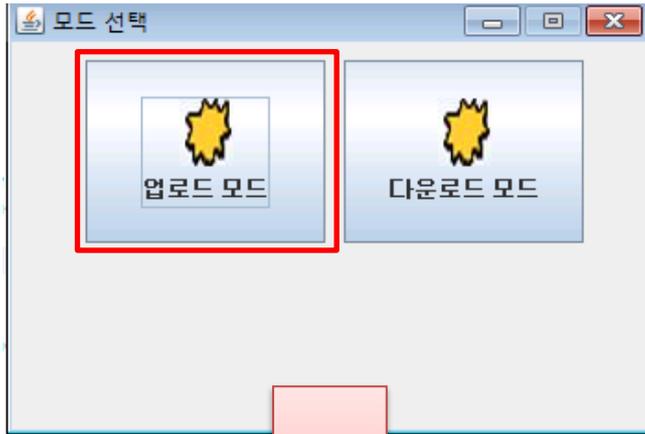
# Execution, 서버 접속



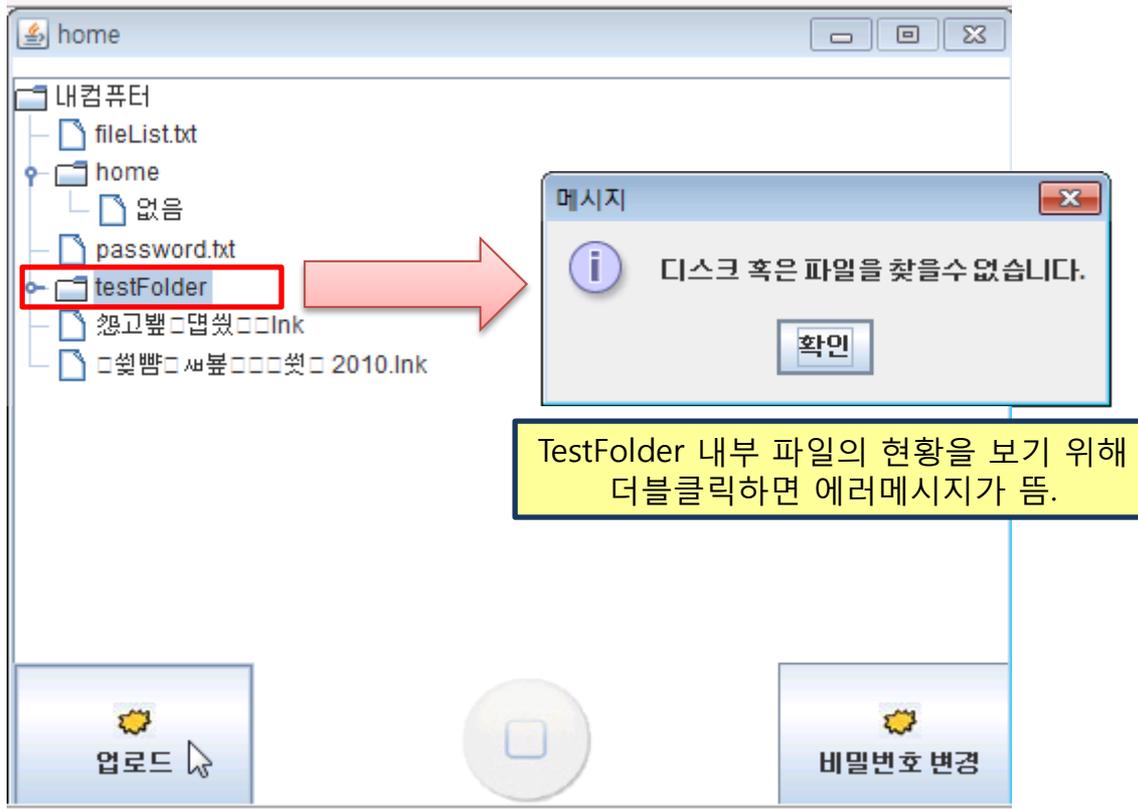
# 시나리오 선정

1. 로그인하여 파일 업로드
2. 파일 다운로드
3. 비밀번호 변경
4. 비밀번호 찾기

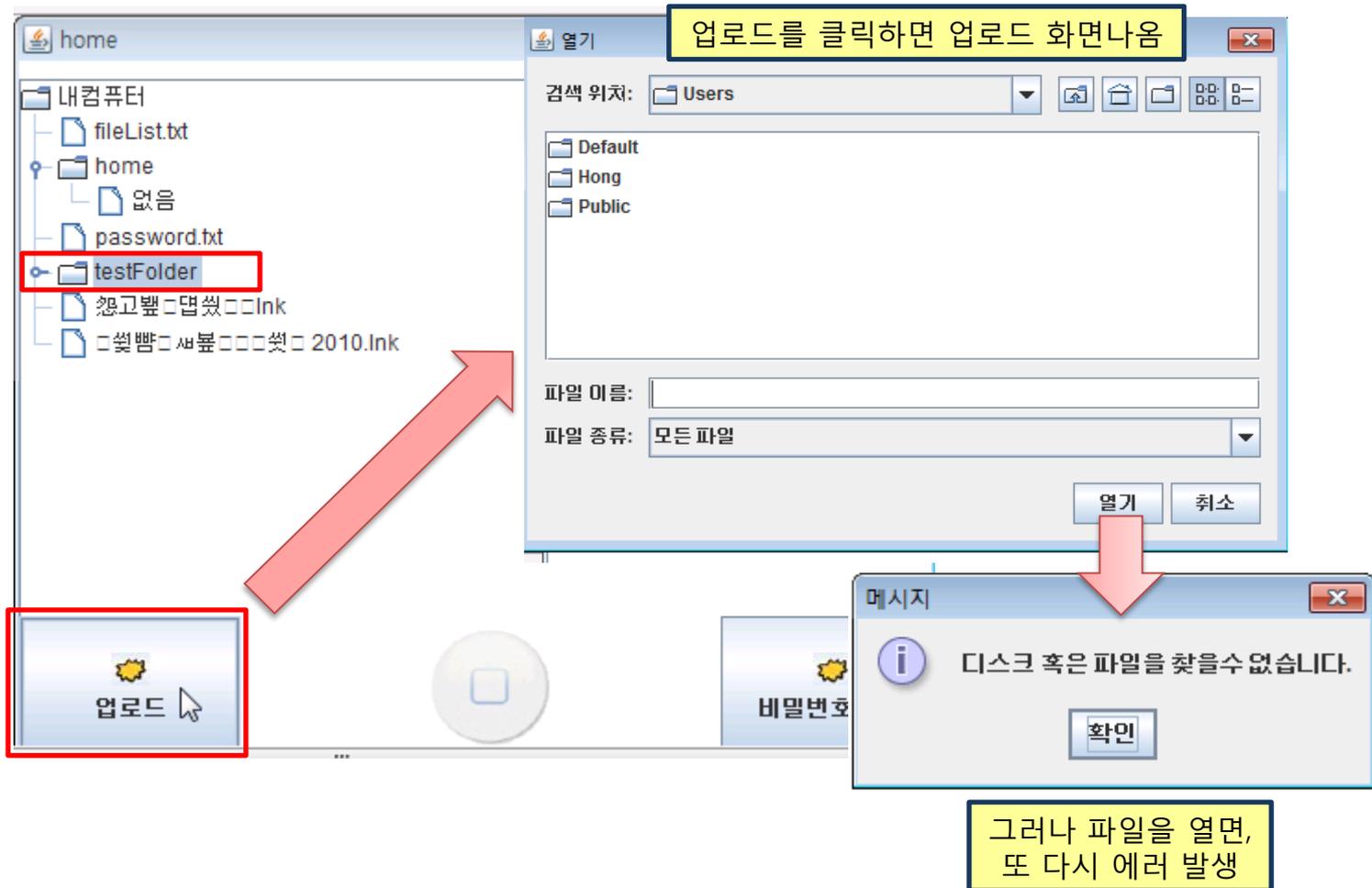
# ① 로그인하여 파일 업로드



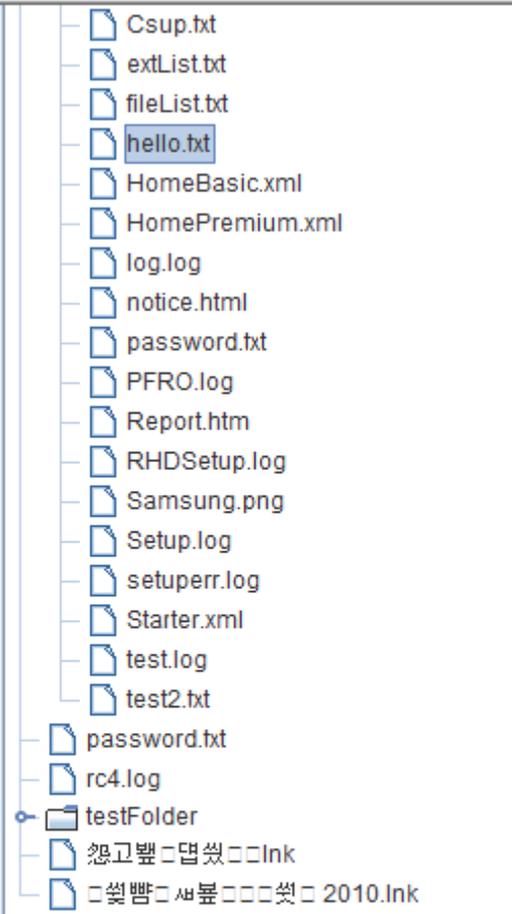
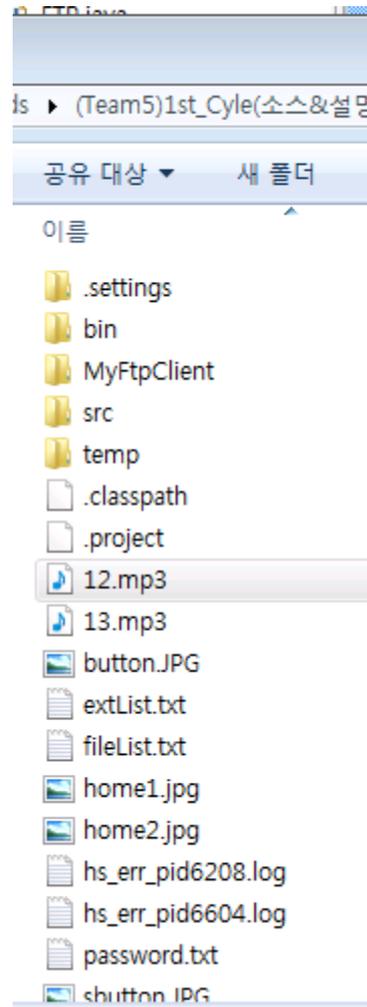
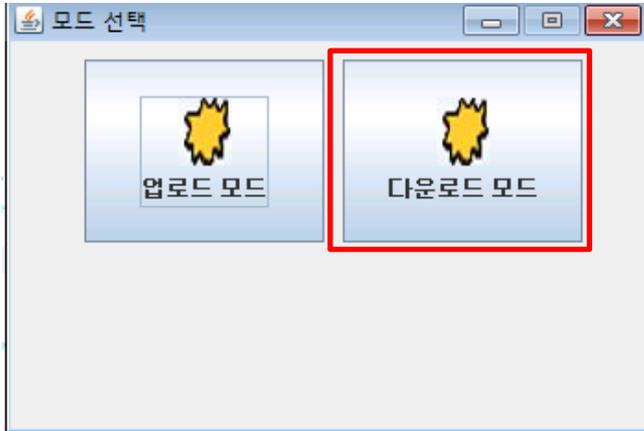
# ① 로그인하여 파일 업로드



# ① 로그인하여 파일 업로드

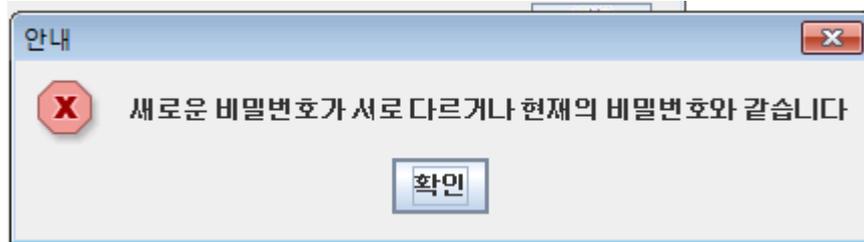
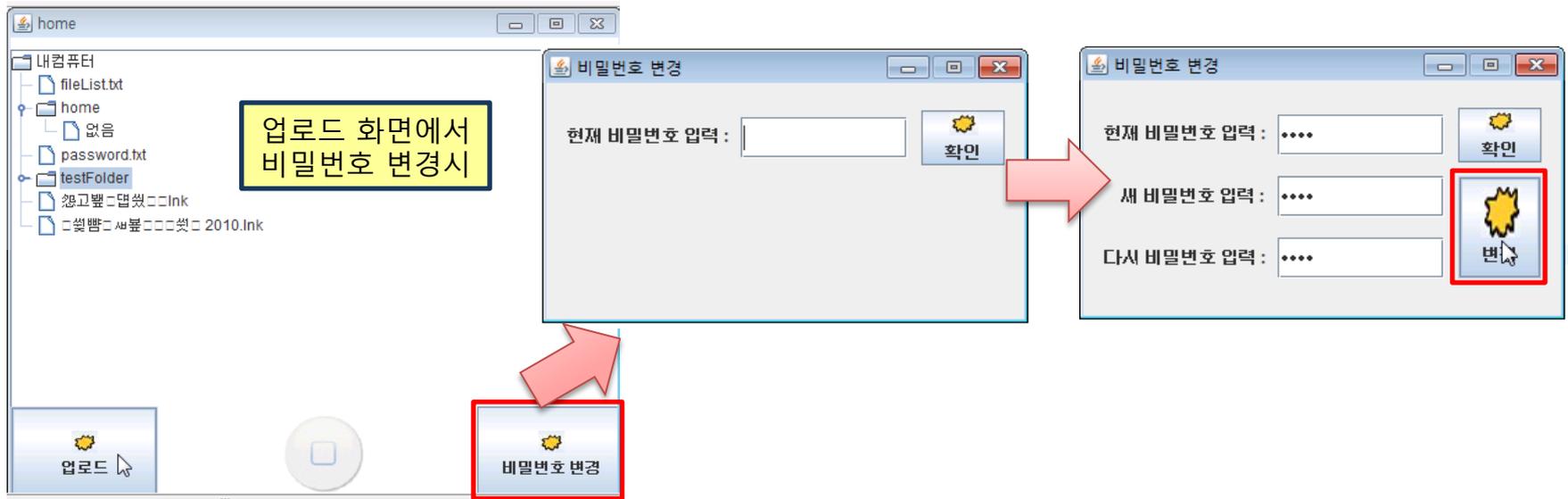


# ② 다운로드 모드

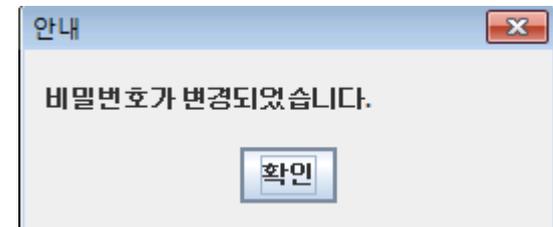


.... 구현이 되어있지 않음

# ③ 비밀번호 변경

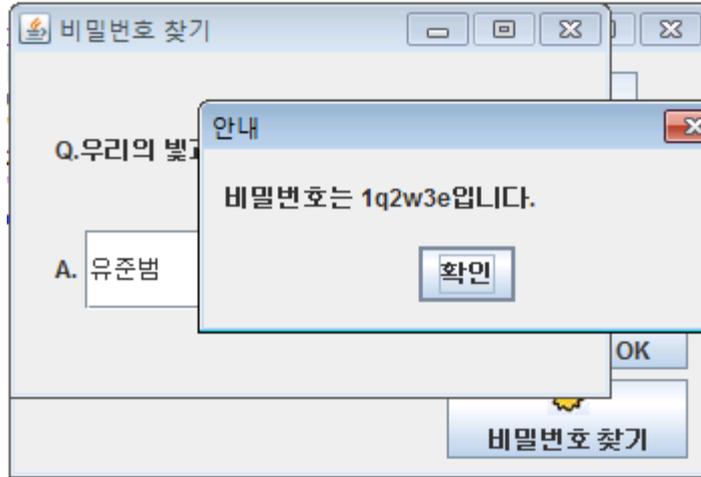


현재 비밀번호인 1234를 새 비밀번호에 입력했을 때, 오류 창

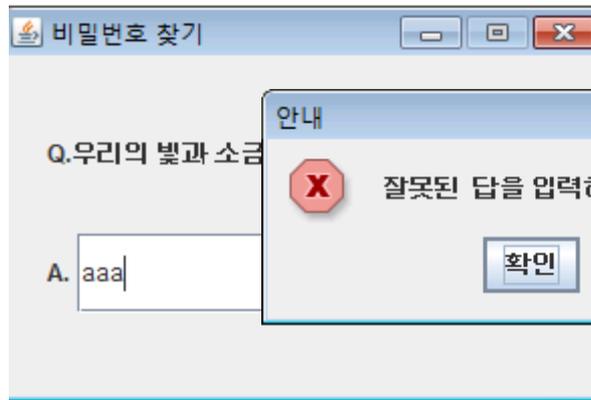


비밀번호가 제대로 바뀌었을 경우

# ④ 비밀번호 찾기



올바르게 입력하면,  
비밀번호를 알려준다.



잘못 입력하면,  
잘못 입력했다는 문구가 뜬다

질문과 답변을 바꿀 수가 없음.

# Junit Code 작성

## ◆ OSP 2030 의 Usecase에 대해 하나씩 작성

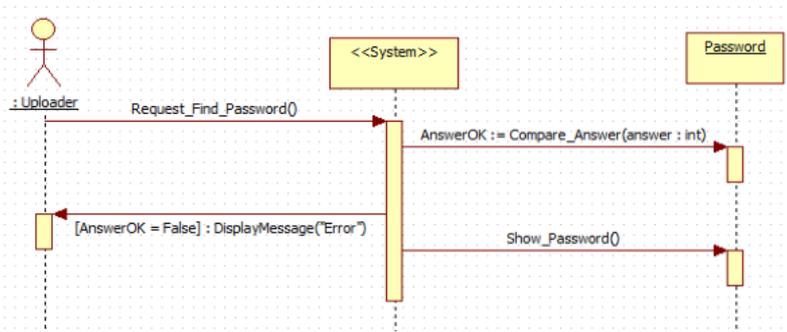
Use case	Name of Actor-Activated Event	System Operation
1. 확장자 기반 파일 분류	None	1.M_Auto_Classify()
2.카테고리 별 확장자 보기	1. Request_Category()	1.M_Category()
3.확장자 추가	1. Request_Add_Extension()	1.M_Add_Extension()
4.중복확장자 검사	None	1.M_Check_Extension()
5.확장자 삭제	1. Request_Delete_Extension()	1.M_Delete_Extension()
6.FTP 연결	1. Input_FTP_Address()	1.M_FTP_Address()
7.모드선택	1. Request_Select_Mode()	1.M_Select_Mode()
8.업로더 모드 로그인	1. Input_Password()	1.M_Password()
9.비밀번호 찾기	1. Request_Find_Password()	1.M_Find_Password()
10.비밀번호 변경	1. Request_Change_Password()	1.M_Change_Password()
11.파일 업로드	1. Request_File_Upload()	1.M_File_Upload()
12.서버 용량 검사	None	1.M_Server_Remain()
13.파일 용량 검사	None	1.M_File_Size()
14.파일 덮어쓰기	None	1.M_Overwrite()
15.파일 다운로드	1. Request_File_Download()	1.M_File_Download()
16.파일 삭제	1. Request_File_Delete()	1.M_File_Delete()

- n01\_Auto\_Classify\_Test.java 0 70. 1. 1 오전 9:00
- n02\_Show\_CategoryTest.java 0 70. 1. 1 오전 9:00
- n03\_Request\_Add\_Extension\_Test.java 0 70. 1. 1 오전 9:00
- n04\_Check\_Extension.java 0 70. 1. 1 오전 9:00
- n05\_Request\_Delete\_Extension\_Test.java 0 70. 1. 1 오전 9:00
- n06\_FTPConnect\_Test.java 0 70. 1. 1 오전 9:00
- n07\_Request\_Select\_Mode\_Test.java 0 70. 1. 1 오전 9:00
- n08\_Input\_Password\_Test.java 0 70. 1. 1 오전 9:00
- n09\_Request\_Find\_Password\_test.java 0 70. 1. 1 오전 9:00
- n10\_Request\_Change\_Password\_Test.java 0 70. 1. 1 오전 9:00
- n11\_Request\_File\_Upload\_Test.java 0 70. 1. 1 오전 9:00
- n12\_Server\_Remain\_Test.java 0 70. 1. 1 오전 9:00
- n13\_File\_Size\_Test.java 0 70. 1. 1 오전 9:00
- n14\_Overwrite\_Test.java 0 70. 1. 1 오전 9:00
- n15\_Request\_File\_Download\_Test.java 0 70. 1. 1 오전 9:00
- n16\_Request\_File\_Delete\_Test.java 0 70. 1. 1 오전 9:00

# Junit Code 작성

## ◆ Usecase #9, Request\_Find\_Password

### Test Code



```
public class Password {
    public String quizAnw = "유준범";
    public String tempAnw;
    public String quiz = "Q. 우리의 최고 소꿉꿘?";
    public String password = "";
    public String tempPw;
    public String insertPw;

    public boolean Compare_Answer(String answer)
    {
        if(quizAnw.equalsIgnoreCase(answer))
            return true;
        else
            return false;
    }
}
```

```
public class n09_Request_Find_Password_test {
    Password passwordtest = null;
    /*
    String
    -> null, 일반 값, 지정 값을 기본 representative value로 잡음.
    */
    @Test
    public void testCompare_Answer_1() throws IOException{
        passwordtest = new Password();
        assertTrue(passwordtest.Compare_Answer("유준범"));
        //quizAnw에 써져 있는거로 하면 됨. (유준범)
    }

    @Test
    public void testCompare_Answer_2() throws IOException{
        passwordtest = new Password();
        assertFalse(passwordtest.Compare_Answer("12345"));
        //오답 왔을때 됨
    }

    @Test
    public void testCompare_Answer_3() throws IOException {
        passwordtest = new Password();
        assertFalse(passwordtest.Compare_Answer(null));
        //null일때도 false다란 뜻임.
    }

    @Test
    public void testShow_Password() throws IOException {
        passwordtest = new Password();
        //그냥 GUI화면으로 보여주는 void 메소드
        assertEquals(passwordtest.password, "1234");
    }
}
```

Compare\_Answer() 메소드

인자를 String으로 받고,  
함수 내에서 비교하여 quizAnw의 값과  
같으면 True, 다르면 False를 나타냄  
이를 테스트케이스 3가지로 (참, 거짓, null) 테스트

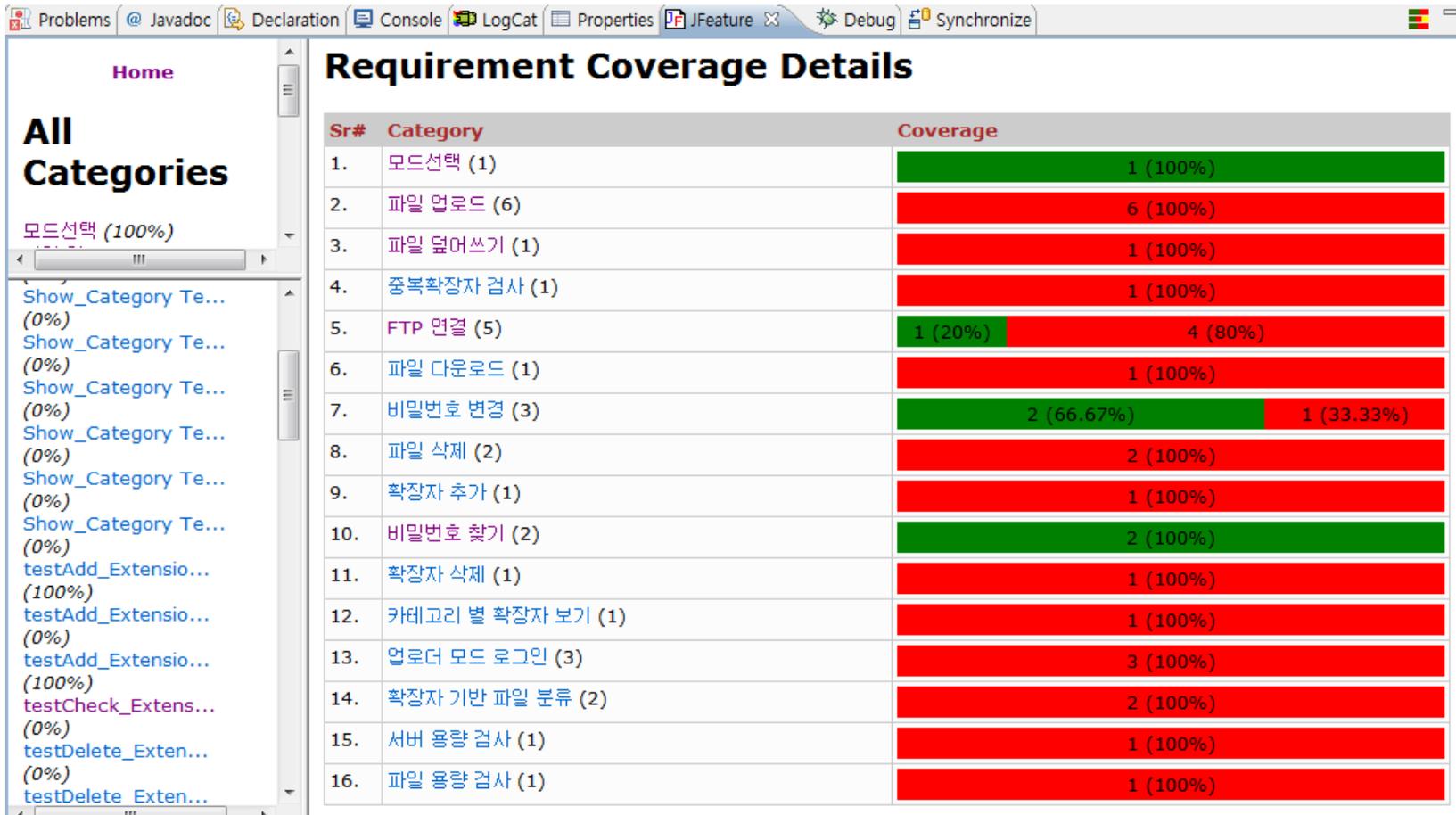
# Requirement 작성

◆ Testcase 별로 요구사항을 나누어 작성

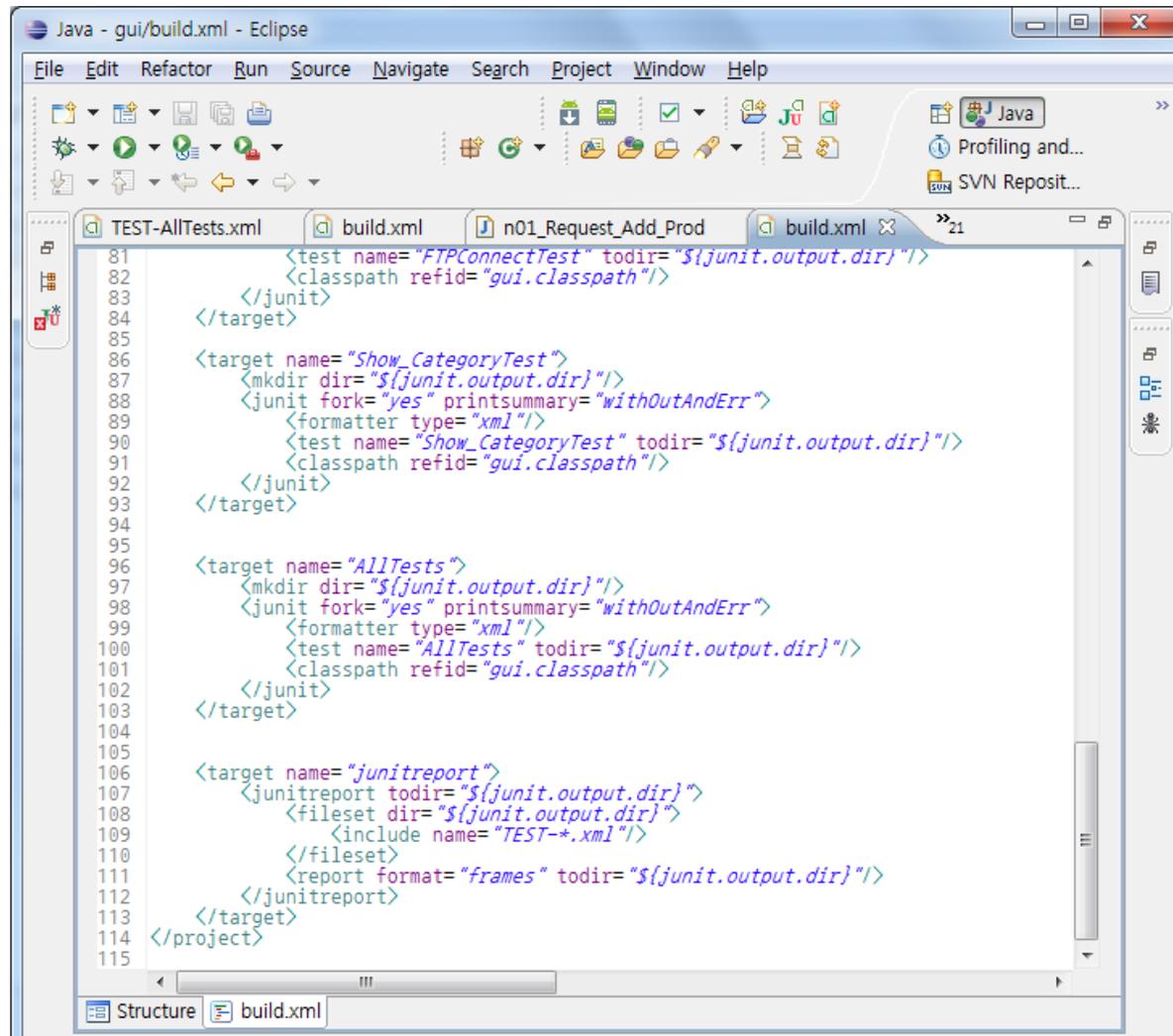
Requirement ID	Category	Sub Category	Title	T.	Priority	Must Have	Test Methods
1	확장자 기반 파일 분류	Determine_Extension()_Test	testDetermine_Extension		3	Yes	n01_Auto_Classify_Test.testDetermine_Extension
1	확장자 기반 파일 분류	Decide_UpPath()_Test	testDecide_UpPath		3	Yes	n01_Auto_Classify_Test.testDecide_UpPath
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_InputZero		3	Yes	n02_Show_CategoryTest.testShow_Category_InputZero
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_InputMaxValue		3	Yes	n02_Show_CategoryTest.testShow_Category_InputMaxVa...
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_InputMinValue		3	Yes	n02_Show_CategoryTest.testShow_Category_InputMinVal...
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_InputMinus		3	Yes	n02_Show_CategoryTest.testShow_Category_InputMinus
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_InputInteger		3	Yes	n02_Show_CategoryTest.testShow_Category_InputInteger
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_Read_InputInteger		3	Yes	n02_Show_CategoryTest.testShow_Category_Read_InputZ...
2	카테고리 별 확장자 보...	Show_Category()_Test	Show_Category_Test_Read_InputZero		3	Yes	n02_Show_CategoryTest.testShow_Category_Read_Input...
3	확장자 추가	Add_Extension()_Test	testAdd_Extension_1		3	Yes	n03_Request_Add_Extension_Test.testAdd_Extension_1
3	확장자 추가	Add_Extension()_Test	testAdd_Extension_2		3	Yes	n03_Request_Add_Extension_Test.testAdd_Extension_2
3	확장자 추가	Add_Extension()_Test	testAdd_Extension_3		3	Yes	n03_Request_Add_Extension_Test.testAdd_Extension_3
4	중복확장자 검사	Check_Extension()_Test	testCheck_Extension		3	Yes	n04_Check_Extension.testCheck_Extension
5	확장자 삭제	Delete_Extension()_Test	testDelete_Extension_1		3	Yes	n05_Request_Delete_Extension_Test.testDelete_Extension1
5	확장자 삭제	Delete_Extension()_Test	testDelete_Extension_2		3	Yes	n05_Request_Delete_Extension_Test.testDelete_Extension2
5	확장자 삭제	Delete_Extension()_Test	testDelete_Extension_3		3	Yes	n05_Request_Delete_Extension_Test.testDelete_Extension3
6	FTP 연결	FTP()_Test	testCreator_1		3	Yes	n06_FTPConnect_Test.testCreator_1
6	FTP 연결	FTP()_Test	testCreator_2		3	Yes	n06_FTPConnect_Test.testCreator_2
6	FTP 연결	FTP()_Test	testCreator_3		3	Yes	n06_FTPConnect_Test.testCreator_3
6	FTP 연결	FTP()_Test	testCreator_4		3	Yes	n06_FTPConnect_Test.testCreator_4
6	FTP 연결	FTP()_Test	testCreator_5		3	Yes	n06_FTPConnect_Test.testCreator_5
6	FTP 연결	FTP()_Test	testCreator_6		3	Yes	n06_FTPConnect_Test.testCreator_6
6	FTP 연결	FTP()_Test	testCreator_7		3	Yes	n06_FTPConnect_Test.testCreator_7
6	FTP 연결	FTP()_Test	testCreator_8		3	Yes	n06_FTPConnect_Test.testCreator_8
6	FTP 연결	FTP()_Test	testCreator_9		3	Yes	n06_FTPConnect_Test.testCreator_9
6	FTP 연결	FTP()_Test	testCreator_10		3	Yes	n06_FTPConnect_Test.testCreator_10
6	FTP 연결	Connect()_Test	testConnect_1		3	Yes	n06_FTPConnect_Test.testConnect_1
6	FTP 연결	Connect()_Test	testConnect_2		3	Yes	n06_FTPConnect_Test.testConnect_2
6	FTP 연결	Connect()_Test	testConnect_3		3	Yes	n06_FTPConnect_Test.testConnect_3

# Requirement 작성

## ◆ Coverage Report



# Build 파일 생성



```
81 <test name="FTPConnectTest" todir="${junit.output.dir}"/>
82 <classpath refid="gui.classpath"/>
83 </junit>
84 </target>
85
86 <target name="Show_CategoryTest">
87 <mkdir dir="${junit.output.dir}"/>
88 <junit fork="yes" printsummary="withOutAndErr">
89 <formatter type="xml"/>
90 <test name="Show_CategoryTest" todir="${junit.output.dir}"/>
91 <classpath refid="gui.classpath"/>
92 </junit>
93 </target>
94
95
96 <target name="AllTests">
97 <mkdir dir="${junit.output.dir}"/>
98 <junit fork="yes" printsummary="withOutAndErr">
99 <formatter type="xml"/>
100 <test name="AllTests" todir="${junit.output.dir}"/>
101 <classpath refid="gui.classpath"/>
102 </junit>
103 </target>
104
105
106 <target name="junitreport">
107 <junitreport todir="${junit.output.dir}">
108 <fileset dir="${junit.output.dir}">
109 <include name="TEST-*.xml"/>
110 </fileset>
111 <report format="frames" todir="${junit.output.dir}"/>
112 </junitreport>
113 </target>
114 </project>
115
```

# Hudson에서 Build

Test Results > (root) 자동 갱신화 령기

## Test Result : (root)

46 failures (+2)

70 tests (+2)  
Took 42 sec.  
 소개 내용 입력

### All Failed Tests

Test Name	Duration	Age
>>> n01 Auto Classify Test.testDecide UpPath	0.015	16
>>> n01 Auto Classify Test.testDetermine Extension	0.006	16
>>> n02 Show CategoryTest.testShow Category InputInteger	0.001	18
>>> n02 Show CategoryTest.testShow Category InputMaxValue	0.003	18
>>> n02 Show CategoryTest.testShow Category InputMinValue	0.001	18
>>> n02 Show CategoryTest.testShow Category InputMinus	0.001	18
>>> n02 Show CategoryTest.testShow Category InputZero	0.001	18
>>> n02 Show CategoryTest.testShow Category Read InputInteger	0.008	18
>>> n02 Show CategoryTest.testShow Category Read InputZero	0.011	18
>>> n03 Request Add Extension Test.testAdd Extension 2	0.003	15
>>> n04 Check Extension.testCheck Extension	0.001	12
>>> n05 Request Delete Extension Test.testDelete Extension1	0.001	10
>>> n06 FTPConnect Test.testConnect 3	0.003	18
>>> n06 FTPConnect Test.testCreator 10	0.0	18
>>> n06 FTPConnect Test.testCreator 2	0.001	18
>>> n06 FTPConnect Test.testCreator 3	0.001	18
>>> n06 FTPConnect Test.testCreator 4	0.001	18
>>> n06 FTPConnect Test.testCreator 5	0.001	18
>>> n06 FTPConnect Test.testCreator 6	0.001	18

>>> n11 Request File Upload Test.testFileUpload	0.001	6
>>> n11 Request File Upload Test.testFile Size Check	0.0	6
>>> n11 Request File Upload Test.testFile exist Check	0.001	6
>>> n11 Request File Upload Test.testServer Remain Check	0.001	6
>>> n12 Server Remain Test.testServer Remain Check	0.0	6
>>> n13 File Size Test.testFile Size Check	0.001	6
>>> n14 Overwrite Test.testFileUpload	0.001	6
>>> n15 Request File Download Test.testFile Download	0.001	2
>>> n16 Request File Delete Test.testDel FileArr	0.001	1
>>> n16 Request File Delete Test.testFileDelete	0.001	1

### All Tests

Class	Duration	Fail	(diff)	Skip	(diff)	Total	(diff)
n01 Auto Classify Test	21 ms	2		0		2	
n02 Show CategoryTest	26 ms	7		0		7	
n03 Request Add Extension Test	5 ms	1		0		3	
n04 Check Extension	1 ms	1		0		1	
n05 Request Delete Extension Test	3 ms	1		0		3	
n06 FTPConnect Test	41 sec	18		0		25	
n07 Request Select Mode Test	0 ms	0		0		1	
n08 Input Password Test	7 ms	3		0		5	
n09 Request Find Password test	4 ms	0		0		4	
n10 Request Change Password Test	7 ms	1		0		7	
n11 Request File Upload Test	4 ms	6		0		6	
n12 Server Remain Test	0 ms	1		0		1	
n13 File Size Test	1 ms	1		0		1	
n14 Overwrite Test	1 ms	1		0		1	
n15 Request File Download Test	1 ms	1		0		1	
n16 Request File Delete Test	2 ms	2	+2	0		2	+2

퍼포먼스

# Hudson에서 Build

- 프로젝트로 돌아가기
- 상태
- 변경사항
- Build Now
- 큐
- Configure
- History
- Tag this build
- Test Result
- Previous Build

## Test Result : n06\_FTPConnect\_Test

18 failures (±0)



### All Tests

Test name	Duration	Status
<a href="#">testConnect_1</a>	0.4 sec	Passed
<a href="#">testConnect_2</a>	0.15 sec	Passed
<a href="#">testConnect_3</a>	3 ms	Failed
<a href="#">testCreator_1</a>	27 ms	Passed
<a href="#">testCreator_10</a>	0 ms	Failed
<a href="#">testCreator_2</a>	1 ms	Failed
<a href="#">testCreator_3</a>	1 ms	Failed
<a href="#">testCreator_4</a>	1 ms	Failed
<a href="#">testCreator_5</a>	1 ms	Failed
<a href="#">testCreator_6</a>	1 ms	Failed
<a href="#">testCreator_7</a>	1 ms	Failed
<a href="#">testCreator_8</a>	1 ms	Failed
<a href="#">testCreator_9</a>	1 ms	Failed
<a href="#">testLogin_1</a>	11 sec	Failed
<a href="#">testLogin_2</a>	6.3 sec	Passed
<a href="#">testLogin_3</a>	11 sec	Passed
<a href="#">testLogin_4</a>	11 sec	Passed
<a href="#">testRead_Extension_FileName_1</a>	1 ms	Failed
<a href="#">testRead_Extension_FileName_2</a>	3 ms	Failed
<a href="#">testRead_Extension_FileName_3</a>	1 ms	Failed



# Hudson에서 Build

**Hudson** search

Hudson » [Team5Test](#) 자동 재실행 켜기

[Back to Dashboard](#)  
[Status](#)  
[Changes](#)  
[Workspace](#)  
[Build Now](#)  
[Delete Project](#)  
[Configure](#)

**Project Team5Test** [소개 내용 입력](#)  
Disable Project

**Test Result Trend**

Workspace  
Recent Changes  
Latest Test Result (46 failures / +2)  
Latest Console output

**Build History (trend)**

Build #	Timestamp	Status
#46	2012. 5. 10 오전 3:41:27	Success
#45	2012. 5. 10 오전 3:36:12	Success
#44	2012. 5. 10 오전 3:31:29	Success
#43	2012. 5. 10 오전 3:27:21	Success
#42	2012. 5. 10 오전 3:23:32	Success
#41	2012. 5. 10 오전 3:18:39	Success
#40	2012. 5. 10 오전 3:09:49	Success
#39	2012. 5. 10 오전 3:04:12	Success
#38	2012. 5. 10 오전 3:00:45	Failure
#37	2012. 5. 10 오전 2:48:08	Success
#36	2012. 5. 10 오전 2:40:39	Success
#35	2012. 5. 10 오전 2:26:09	Success
#34	2012. 5. 10 오전 2:22:44	Success

**Permalinks**

- [Last build \(#46\), 17 min ago](#)
- [Last stable build \(#3\), 7 hr 5 min ago](#)
- [Last successful build \(#46\), 17 min ago](#)
- [Last failed build \(#28\), 5 hr 9 min ago](#)
- [Last unstable build \(#46\), 17 min ago](#)
- [Last unsuccessful build \(#46\), 17 min ago](#)

**Test Result Trend Data (Estimated)**

Build #	Successful (Blue)	Failed (Red)	Total
#29	0	25	25
#30	0	25	25
#31	0	28	28
#32	0	28	28
#33	0	28	28
#34	0	28	28
#35	0	28	28
#36	10	22	32
#37	15	20	35
#38	15	20	35
#39	15	20	35
#40	25	10	35
#41	25	15	40
#42	25	15	40
#43	25	15	40
#44	25	15	40
#45	25	15	40
#46	25	15	40

(just show failures) enlarge