

# 1<sup>st</sup> Testing

## TEAM 1

PT

200711437 성하진

200511355 정용구

200911436 조성완

201260053 Abbos Shomurodov

201260058 채승흠

# Contents

**1. Test Environment**

**2. 1<sup>st</sup> Testing \_Team1**

**3. 1<sup>st</sup> Testing \_Team4**

# Testing Environment

1. Testing Environment for CTIP
2. Introduce Allpairs

# Testing Environment(CTIP)

---

**[OS]** Windows7

**[Language]** Java

**[Unit Test Tool]** JUnit

**[Build Automation]** Ant

**[SCM Tool]** Visual SVN(Subversion)

**[CI Tool]** Hudson-2.2.0

<WAS : Tomcat 7.0>

**[RE Tool]** JFeature

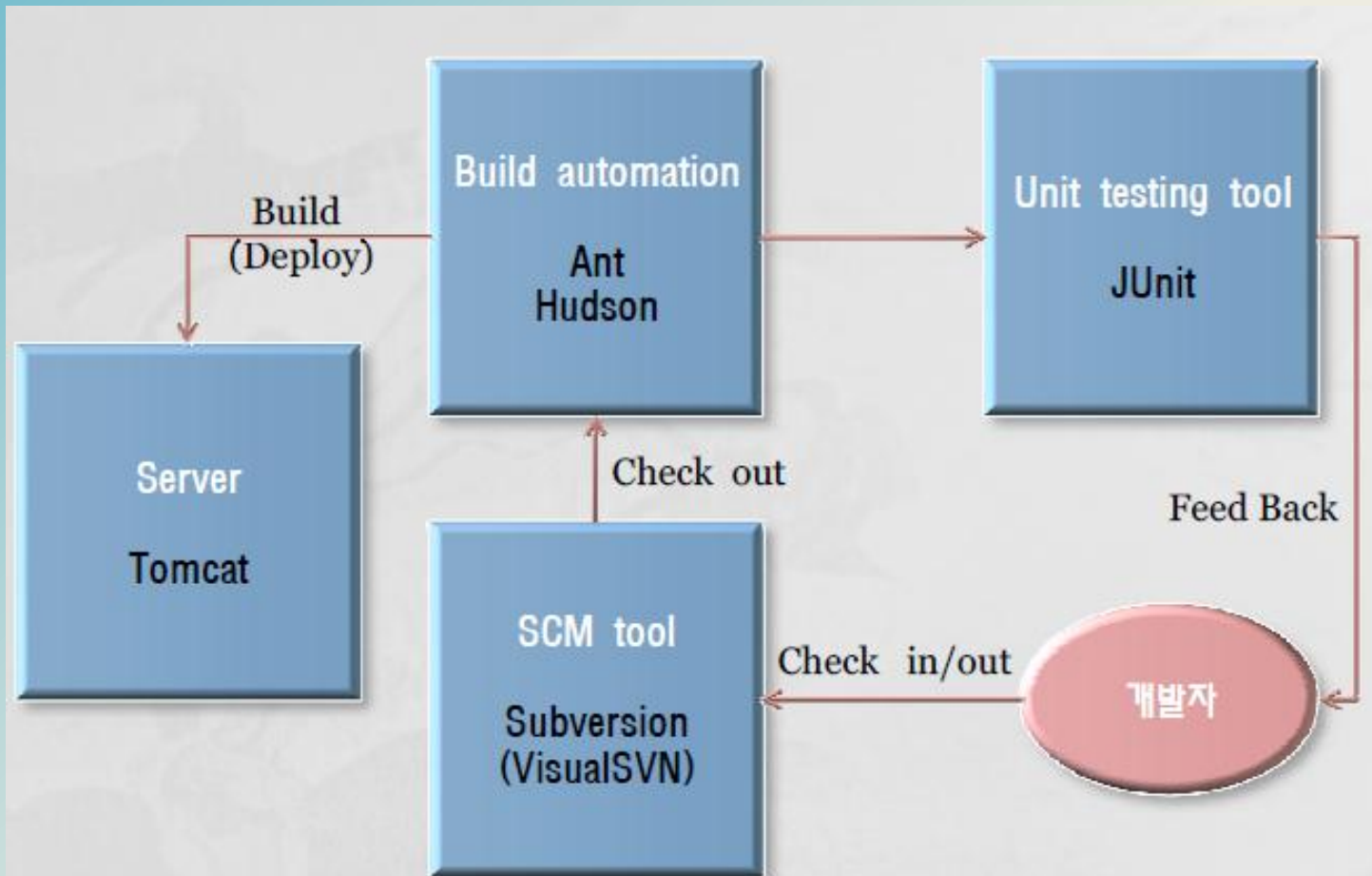
# Testing Environment(CTIP) cont.

## RE Tool 변경[OSRMT -> JFeature]

- 이전에, 사용이 편리하고 언어 독립적이라는 장점을 들어 Requirements를 OSRMT로 관리하고자 하였으나 Jfeature가 현재의 소규모 테스트 환경에 적합하다고 판단하여 사용 Tool을 변경

# Testing Environment(CTIP) cont.

## ✓ CTIP Environment



# Pairwise

- Pairwise란?

- 어떤 Testing은 다양한 값들을 가질 수 있는 많은 변수들을 가지고 있어, 그 변수와 값들로 다양한 Combination을 생성하고 Testing하는 것을 요구함

- Pairwise Tools

- Test 대상 Combination의 수가 늘어나면 테스터가 일일이 구할 수가 없으므로 자동화된 Tool이 등장

# Pairwise(cont.)

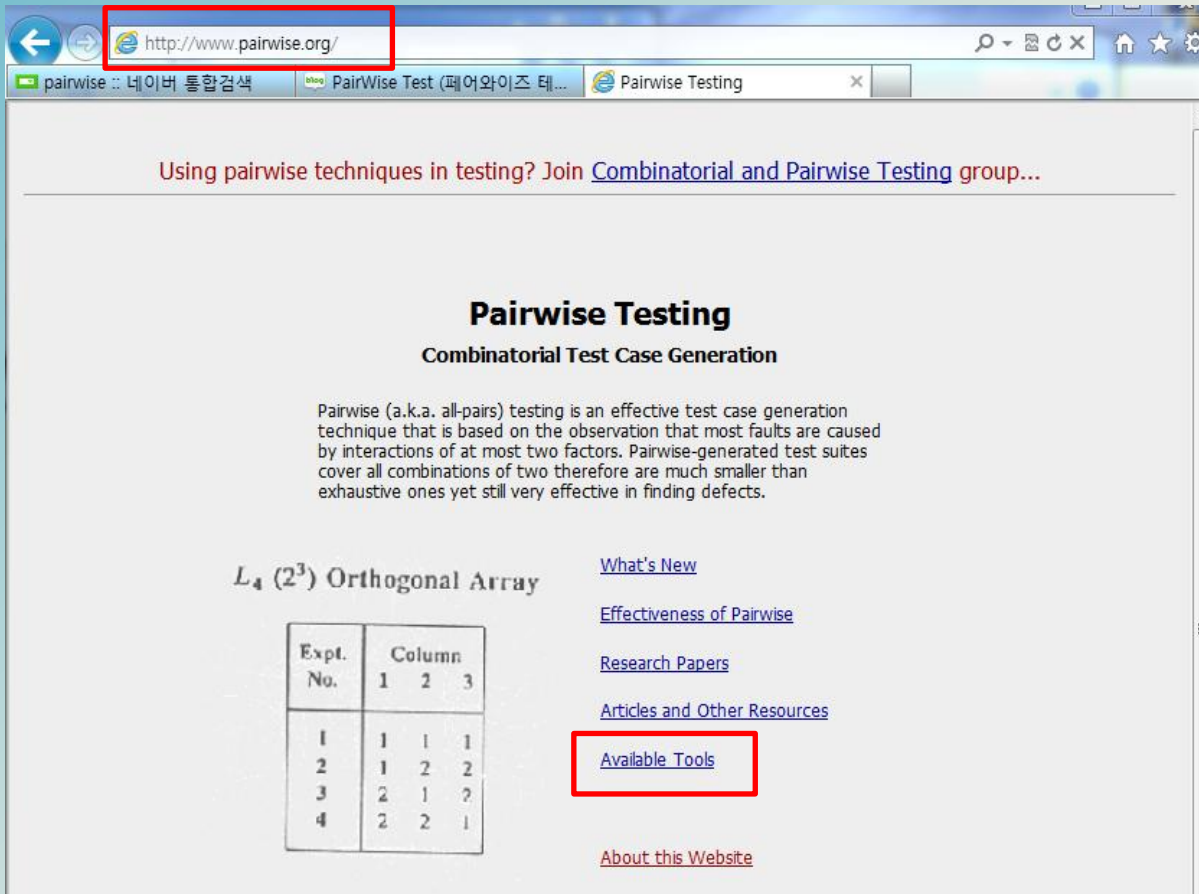
---

- Combination은 모든 value의 쌍(pair)이 서로 결합되는 방식으로 생성 - 완벽한 Test case는 아님
- [www.pairwise.org](http://www.pairwise.org)에서 여러 종류의 Tool 다운로드 가능
  - ✓ 사용 Pairwise Tool - Allpairs



# Pairwise -Allpairs

## ✓ Allpairs 다운로드



Using pairwise techniques in testing? Join [Combinatorial and Pairwise Testing group...](#)

### Pairwise Testing

#### Combinatorial Test Case Generation

Pairwise (a.k.a. all-pairs) testing is an effective test case generation technique that is based on the observation that most faults are caused by interactions of at most two factors. Pairwise-generated test suites cover all combinations of two therefore are much smaller than exhaustive ones yet still very effective in finding defects.

$L_4 (2^3)$  Orthogonal Array

Expt. No.	Column		
	1	2	3
1	1	1	1
2	1	2	2
3	2	1	?
4	2	2	1

[What's New](#)

[Effectiveness of Pairwise](#)

[Research Papers](#)

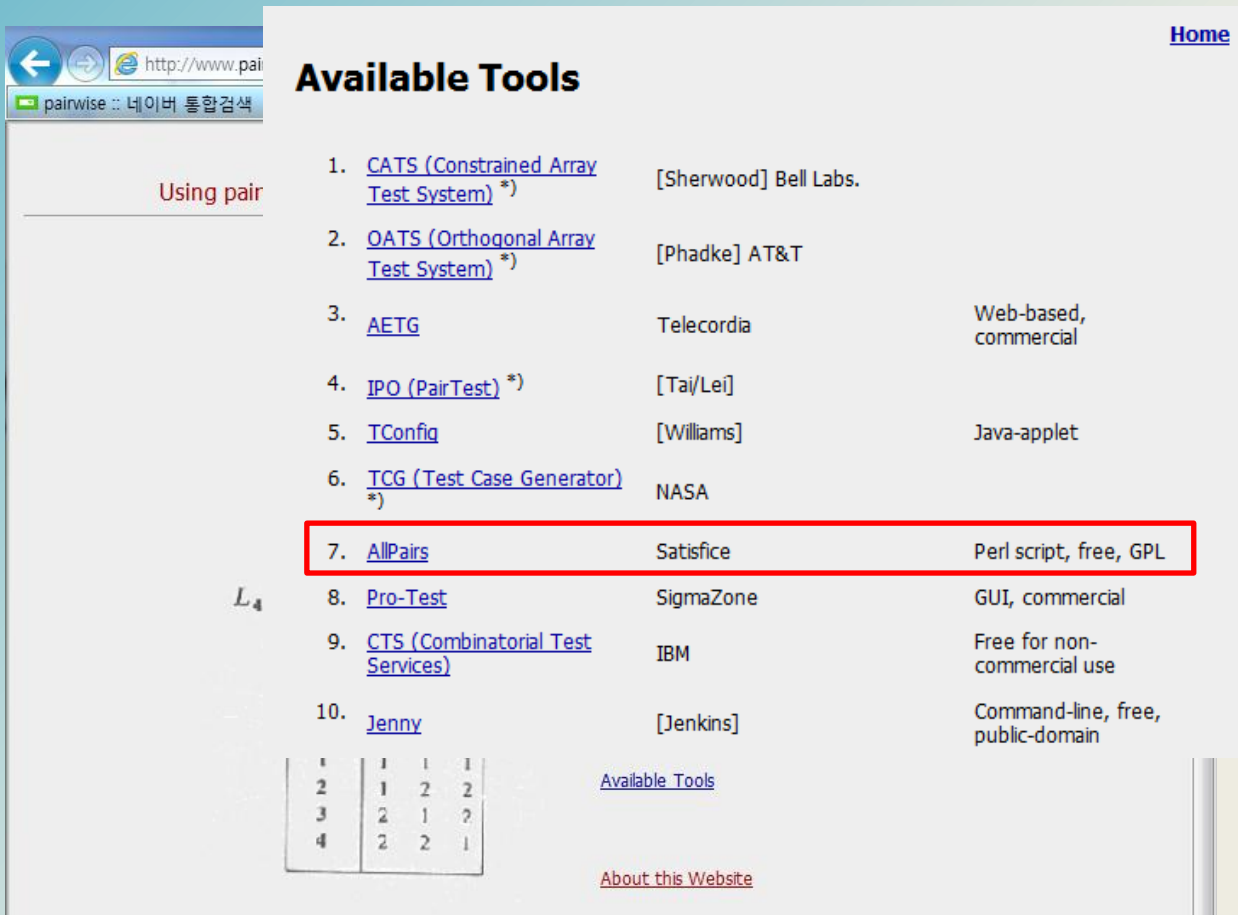
[Articles and Other Resources](#)

[Available Tools](#)

[About this Website](#)

# Pairwise -Allpairs

## ✓ Allpairs 다운로드



Using pair

Available Tools

Home

1. [CATS \(Constrained Array Test System\) \\*](#)) [Sherwood] Bell Labs.
2. [OATS \(Orthogonal Array Test System\) \\*](#)) [Phadke] AT&T
3. [AETG](#) Telecordia Web-based, commercial
4. [IPO \(PairTest\) \\*](#)) [Tai/Lei]
5. [TConfig](#) [Williams] Java-applet
6. [TCG \(Test Case Generator\) \\*](#)) NASA
7. [AllPairs](#) Satisfice Perl script, free, GPL
8. [Pro-Test](#) SigmaZone GUI, commercial
9. [CTS \(Combinatorial Test Services\)](#) IBM Free for non-commercial use
10. [Jenny](#) [Jenkins] Command-line, free, public-domain

$L_4$

1	1	1	1
2	1	2	2
3	2	1	?
4	2	2	1

[Available Tools](#)

[About this Website](#)

# Pairwise -Allpairs

## ✓ Allpairs 설치 및 사용

압축 해제

이름	수정한 날짜	유형	크기
allpairs.exe	2012-05-08 오후...	응용 프로그램	728KB
allpairs.pl	2012-05-08 오후...	PL 파일	13KB
allpairs.rtf	2012-05-08 오후...	서식 있는 텍스트	1,815KB
example.txt	2012-05-08 오후...	텍스트 문서	1KB
license.txt	2012-05-08 오후...	텍스트 문서	16KB
printing.txt	2012-05-08 오후...	텍스트 문서	1KB
printing.xls	2012-05-08 오후...	Microsoft Excel 9...	189KB

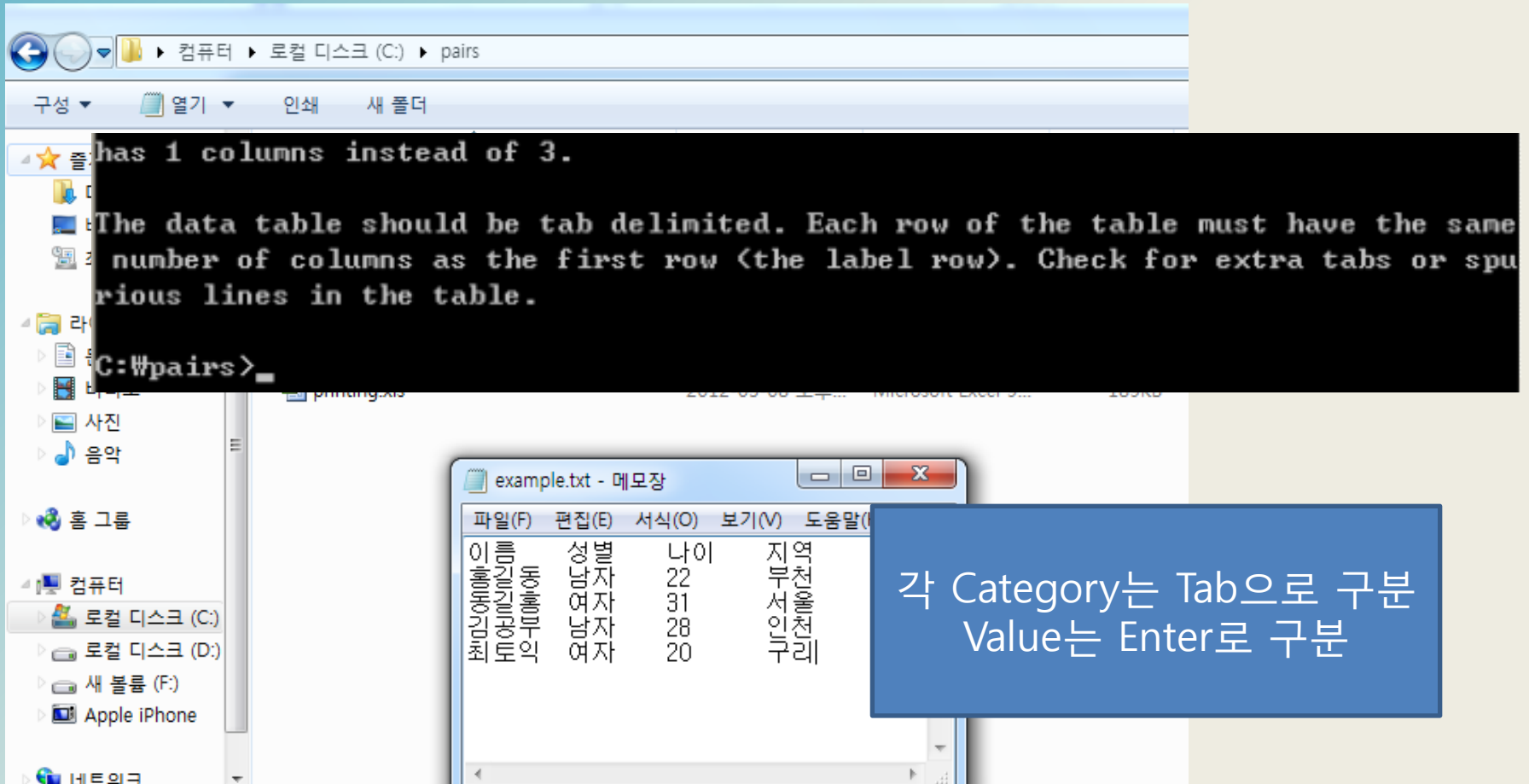
example.txt - 메모장

파일(F)	편집(E)	서식(O)	보기(V)	도움말(H)
이름	성별	나이	지역	
홍길동	남자	22	부천	
김영희	여자	31	서울	
정민준	남자	28	인천	
최토의	여자	20	구리	

조합을 구하려는 항목을 \*.txt파일에 입력

# Pairwise -Allpairs

## ✓ Allpairs txt파일 작성 주의사항



The image shows a Windows file explorer window with a black error message overlay. The error message reads: "has 1 columns instead of 3. The data table should be tab delimited. Each row of the table must have the same number of columns as the first row (the label row). Check for extra tabs or spurious lines in the table." Below the error message, a Notepad window titled "example.txt - 메모장" is open, displaying a table with four columns: 이름, 성별, 나이, and 지역. The table contains four rows of data, with values separated by tabs.

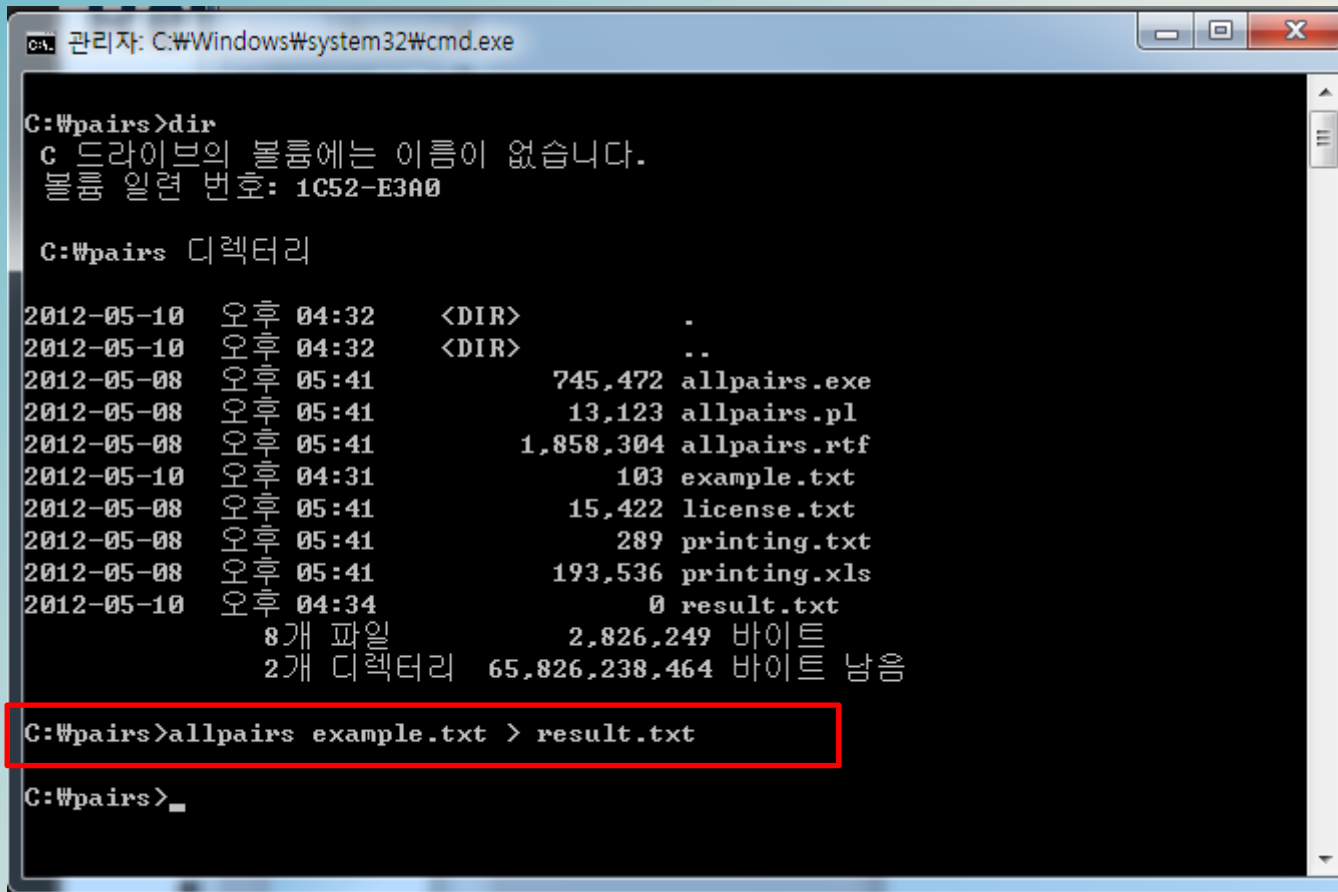
```
C:#pairs>_
```

이름	성별	나이	지역
홍길동	남자	22	부천
김영희	여자	31	서울
정민준	남자	28	인천
최토의	여자	20	구리

각 Category는 Tab으로 구분  
Value는 Enter로 구분

# Pairwise -Allpairs

## ✓ Allpairs 사용



The screenshot shows a Windows command prompt window titled "관리자: C:\Windows\system32\cmd.exe". The prompt is at "C:\wpairs>". The user has entered the command "dir", which has returned the following output:

```
C:\wpairs>dir
C 드라이브의 볼륨에는 이름이 없습니다.
볼륨 일련 번호: 1C52-E3A0

C:\wpairs 디렉터리

2012-05-10 오후 04:32 <DIR> .
2012-05-10 오후 04:32 <DIR> ..
2012-05-08 오후 05:41          745,472 allpairs.exe
2012-05-08 오후 05:41          13,123 allpairs.pl
2012-05-08 오후 05:41     1,858,304 allpairs.rtf
2012-05-10 오후 04:31           103 example.txt
2012-05-08 오후 05:41          15,422 license.txt
2012-05-08 오후 05:41           289 printing.txt
2012-05-08 오후 05:41       193,536 printing.xls
2012-05-10 오후 04:34              0 result.txt
            8개 파일              2,826,249 바이트
            2개 디렉터리     65,826,238,464 바이트 남음
```

The command "allpairs example.txt > result.txt" is entered and highlighted with a red box. The prompt then returns to "C:\wpairs>".

# Pairwise -Allpairs(example)

## ✓ Result.txt 내용

```

result.txt - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

TEST CASES
case 이 나이 지
1 22 6
2 31 6
3 28 6
4 20 6
5 31 6
6 22 6
7 20 6
8 28 6
9 28 6
10 20 6
11 22 6
12 31 6
13 20 6
14 28 6
15 31 6
16 22 6

PAIRING DETAILS
var1 var2 value appearances cases
이 나이 22 1 1
이 나이 31 1 2
이 나이 28 1 3
이 나이 20 1 4
이 나이 31 1 5
이 나이 22 1 6
이 나이 20 1 7
이 나이 28 1 8
이 나이 28 1 9
이 나이 20 1 10
이 나이 22 1 11
이 나이 31 1 12
이 나이 20 1 13
이 나이 28 1 14
이 나이 31 1 15
이 나이 22 1 16
이 나이 22 1 1
이 나이 31 2
  
```

```

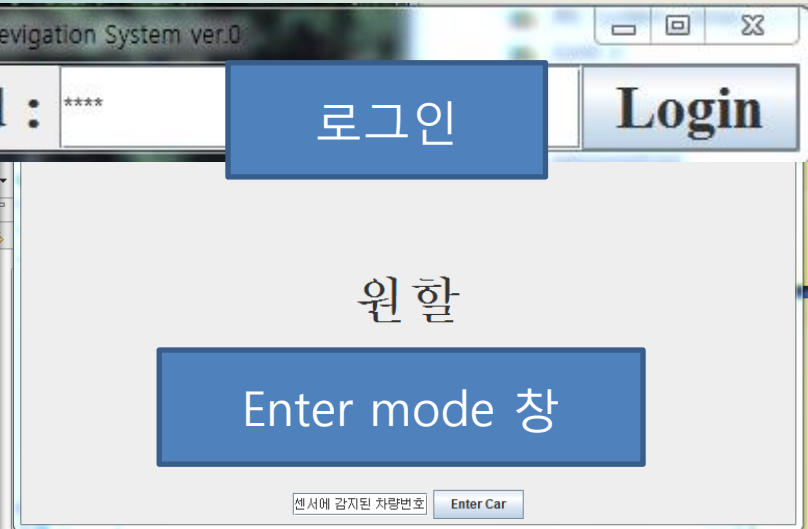
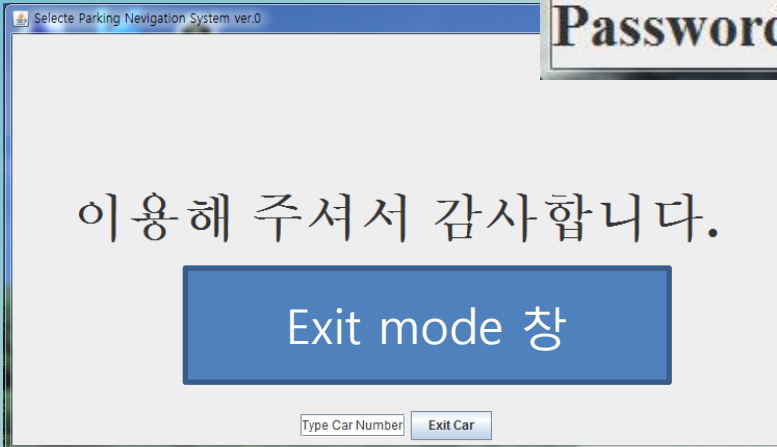
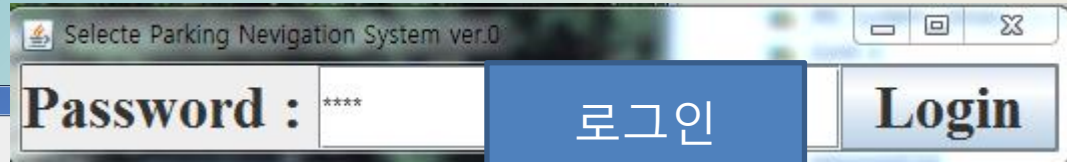
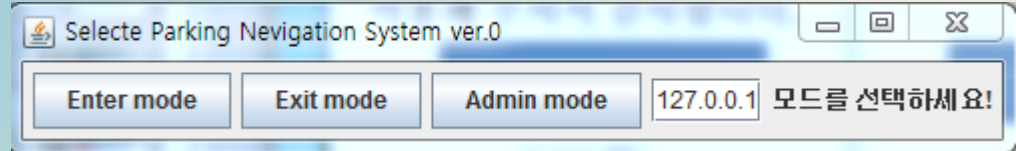
이 나이 28 1 3
이 나이 20 1 4
이 나이 22 1 6
이 나이 31 1 5
이 나이 28 1 8
이 나이 20 1 7
이 나이 31 1 11
이 나이 22 1 12
이 나이 28 1 9
이 나이 22 1 10
이 나이 31 1 16
이 나이 28 1 15
이 나이 20 1 14
이 나이 22 1 13
이 나이 28 1 1
이 나이 31 2
이 나이 20 1 11
이 나이 22 1 10
이 나이 31 1 9
이 나이 28 1 14
이 나이 22 1 13
이 나이 31 1 16
이 나이 20 1 15
이 나이 28 1 1
이 나이 31 5
이 나이 22 1 9
이 나이 20 1 13
이 나이 31 1 6
이 나이 28 1 2
이 나이 20 1 14
이 나이 22 1 10
이 나이 31 1 11
이 나이 28 1 15
  
```

# 1<sup>st</sup> Testing \_Team1

1. Program Introduction
2. System Testing
3. Test case example

# Team 1 Introduction

## ✓ 시작 화면





# Team 1 Introduction

## ✓ 시작 화면

이용해 주셔서 감사합니다.

Exit차량 입력

Type Car Number Exit Car

원할

Enter차량 입력

센서에 감지된 차량번호 Enter Car

1층 0번	1층 1번	1층 2번	1층 3번
1층 4번	1층 5번	1층 6번	1층 7번
1층 8번	1층 9번	1층 10번	1층 11번
1층 12번	1층 13번	1층 14번	1층 15번
1층 16번	1층 17번	1층 18번	1층 19번

Select Fee! Extra Information Modi... Password Modify! Find data!!

Sort data! Change Location!! < >

상태 화면에 입/출 차량 즉시 반영

# Team 1 System Testing

## ✓ Problem.1 Sort data

Select Fee!	Extra Information Modi...	Password Modify!	Find data!!
<b>Sort data!</b>	Change Location!!	<	>

Select Parking Ne **Lot.59(3층 19번) 차량 부재시** ver.0

정렬창

CarNum up

정렬하기

0 : 54#경기31바3456#201205011509#주차됨  
1 : 01#경기37나1564#20120511002016#주차됨  
2 : 39#경기37나5554#20120510233412#주차됨  
3 : 40#경기40나3212#20120503130709#주차됨  
4 : 44#경기40나3212#20120503130706#주차됨  
5 : 45#경기40나3212#20120503130644#주차됨  
6 : 48#경기40나3212#20120503130703#주차됨  
7 : 49#경기40나3212#20120503130648#주차됨  
8 : 51#경기40나3212#20120503130722#주차됨  
9 : 52#경기40나3212#20120503130659#주차됨  
10 : 53#경기40나3212#20120503130650#주차됨  
11 : 56#경기40나3212#20120503130656#주차됨  
12 : 57#경기40나3212#20120503130654#주차됨  
13 : 58#경기40나3212#20120503130711#주차됨  
14 : 22#경남22나1122#20120501022210#주차됨

CarNum up

정렬하기

40 : 31is empty!  
47 : 32is empty!  
48 : 35is empty!  
49 : 36is empty!  
50 : 37is empty!  
51 : 38is empty!  
52 : 41is empty!  
53 : 42is empty!  
54 : 43is empty!  
55 : 46is empty!  
56 : 47is empty!  
57 : 50is empty!  
58 : 55is empty!  
59 : 59is empty!

# Team 1 System Testing(cont.)

Select Fee!	Extra Information Modi...	Password Modify!	Find data!!
Sort data!	Change Location!!	<	>

Lot.59(3층 19번) 차량 입장시

Selecte Parking Navigation System ver.0

정렬창

CarNum up

정렬하기

```

0 : 54#경기31바3456#201205011509#주차됨
1 : 01#경기37나1564#20120511002016#주차됨
2 : 39#경기37나5554#20120510233412#주차됨
3 : 40#경기40나3212#20120503130709#주차됨
4 : 44#경기40나3212#20120503130706#주차됨
5 : 45#경기40나3212#20120503130644#주차됨
6 : 48#경기40나3212#20120503130703#주차됨
7 : 49#경기40나3212#20120503130648#주차됨
8 : 51#경기40나3212#20120503130722#주차됨
9 : 52#경기40나3212#20120503130659#주차됨
10 : 53#경기40나3212#20120503130650#주차됨
11 : 56#경기40나3212#20120503130656#주차됨
12 : 57#경기40나3212#20120503130654#주차됨
13 : 58#경기40나3212#20120503130711#주차됨
14 : 22#경남22마1122#20120501022210#주차됨
    
```

Selecte Parking Navigation System ver.0

정렬창

CarNum up

정렬하기

```

40 : 31is empty!
47 : 32is empty!
48 : 35is empty!
49 : 36is empty!
50 : 37is empty!
51 : 38is empty!
52 : 41is empty!
53 : 42is empty!
54 : 43is empty!
55 : 46is empty!
56 : 47is empty!
57 : 50is empty!
58 : 55is empty!
59 : 59#경기11하9970#20120511003035#주차됨
    
```

# Team 1 System Testing(cont.)

## ✓ Problem.2 Enter Path



주차공간을 선택한 경우 경로가 출력됩니다.

# Team 1 System Testing(cont.)

Selecte Parking Navigation System ver.0

Welcome SNP System!! To Server

실제 실행 시의 차량 입장화면

1층0...	1층1...	1층2...	1층3...
1층4...	1층5...	1층6...	1층7...
1층8...	1층9...	1층1...	1층1...
1층1...	1층1...	1층1...	1층1...
1층1...	1층1...	1층1...	1층1...

주차증 출력

Select Fee! Extra Information ...

Sort data! Change Location!!

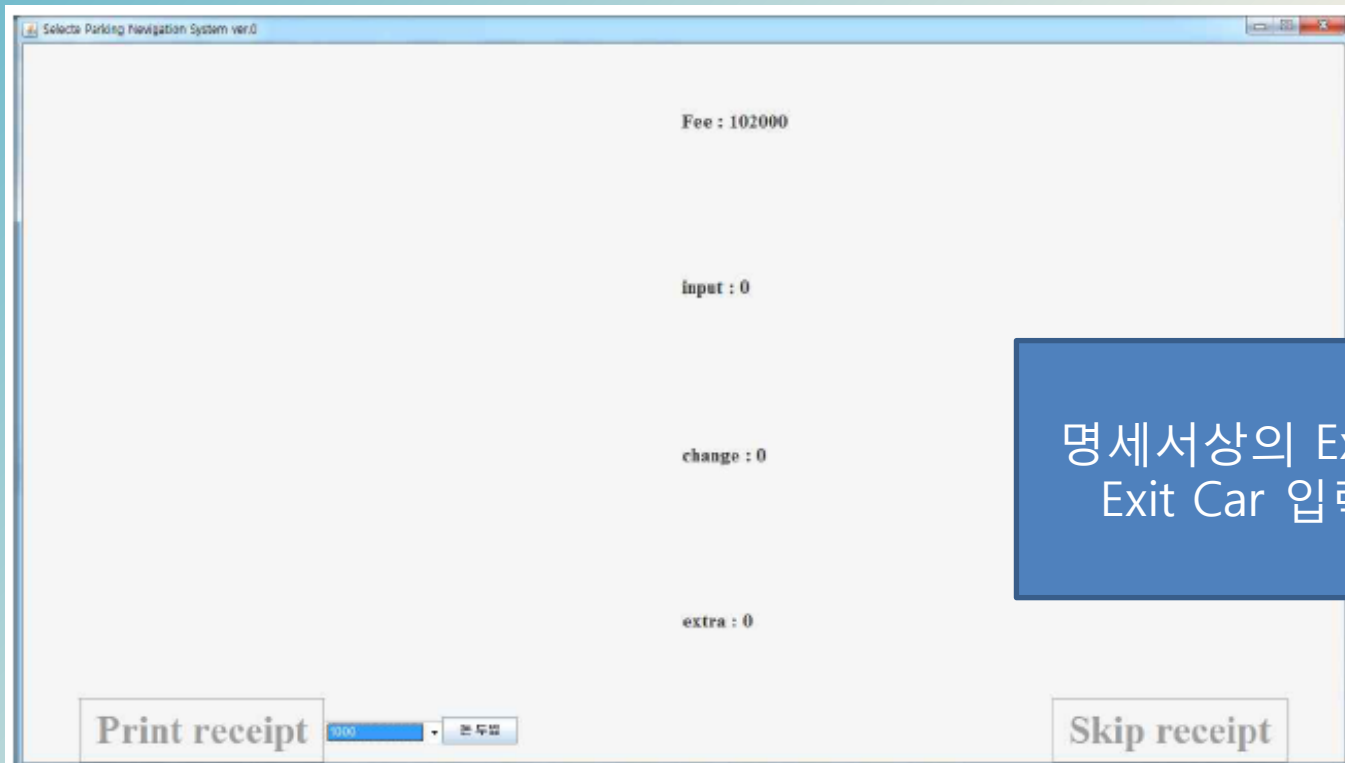
주차증

차량번호 : 경기33나2223  
1층4번 자리  
입장시간 : 2012년05월11일00시40분04초  
입구에서부터 바로 왼쪽으로 꺾으세요.  
3블록 직진하세요.  
오른쪽으로 꺾으세요.

확인

# Team 1 System Testing(cont.)

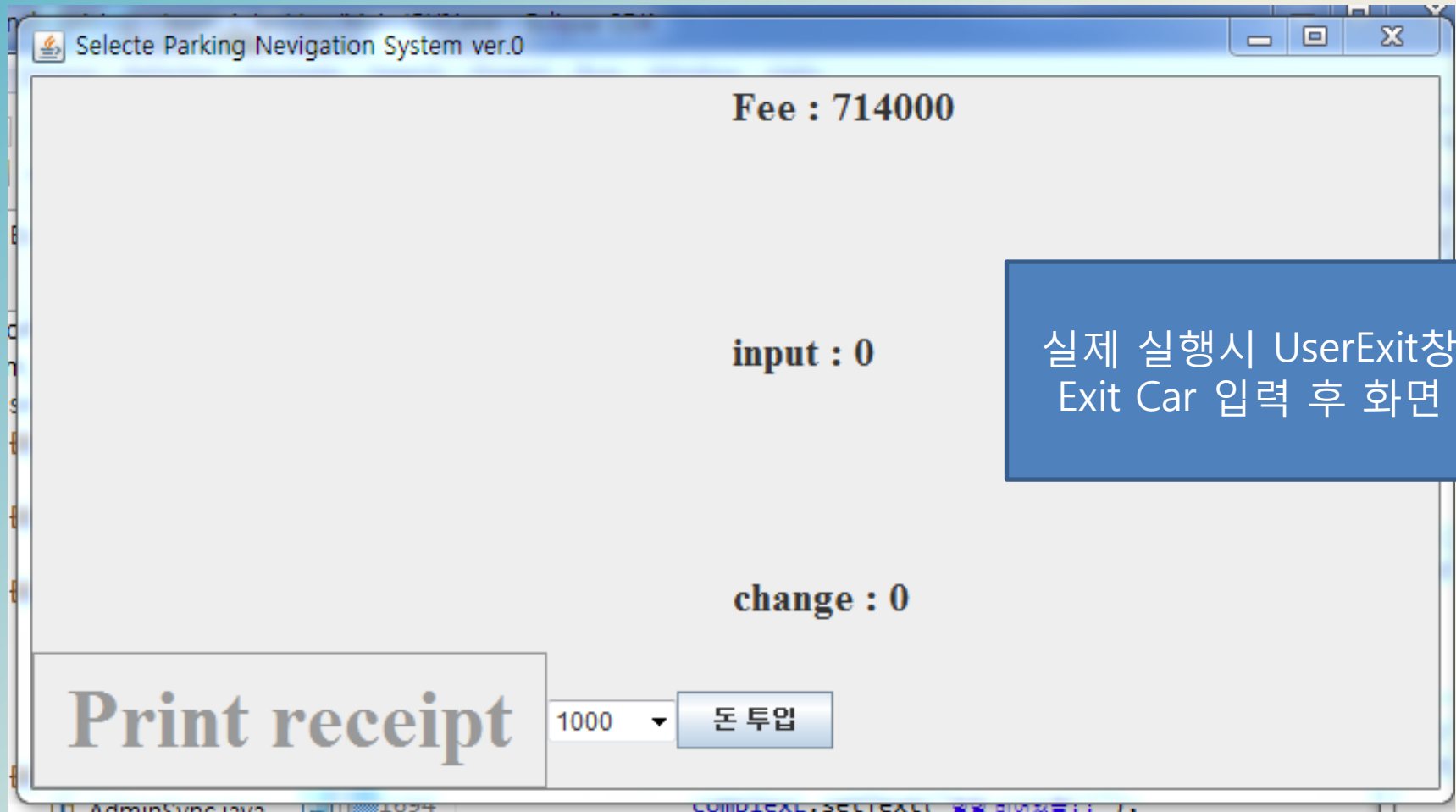
## ✓ Problem.3 Print Receipt



명세서상의 Exit mode 창  
Exit Car 입력 후 화면

요금이 출력되며 하단에 '돈 투입' 버튼을 통하여 돈을 지불 할 수 있습니다. input은 넣어 준 금액이며, change는 거스름돈, extra는 부족한 금액입니다.

# Team 1 System Testing(cont.)



# Team 1 System Testing(cont.)

## ✓ Problem.4 Parking Lot 공사중

마지막에 "주차됨"이라는 문자열을 넣어줍니다.

주차공간번호#NA#NA#NA#NA를 써줄 경우, 공사 중으로 간주됩니다.

완성된 car.txt 예

10#서울80나3828#201205021208#주차됨

12#경북84나7777#201205021245#주차됨

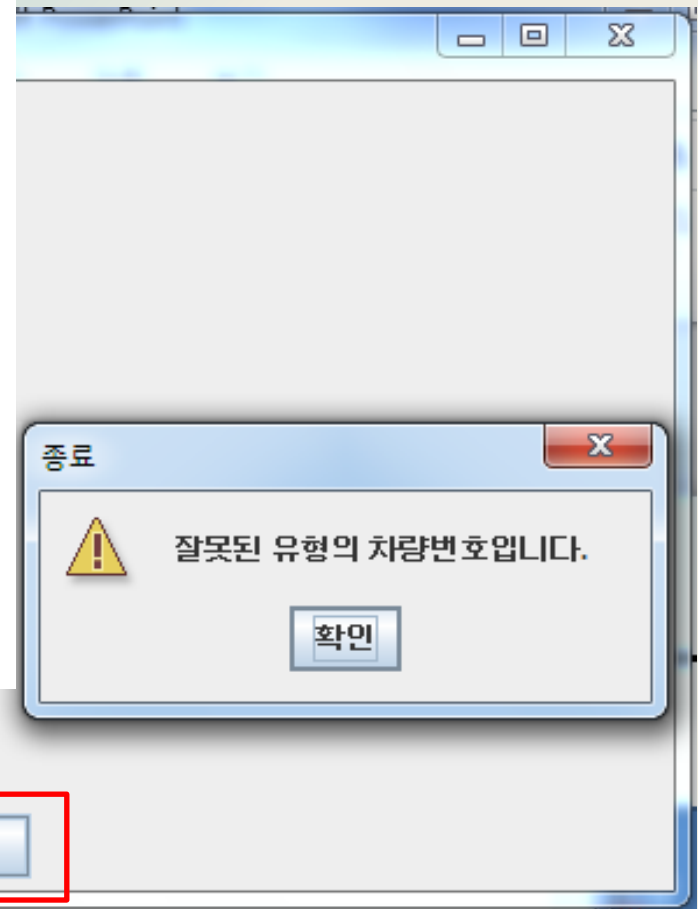
18#경기40나9212#20120503190821#주차됨

### 1.2.2 password.txt 만들기

관리자에서 쓰일 비밀번호를 텍스트 파일에 한 줄로 적어 넣습니다.

3#NA#NA#NA#

Enter Car





# Team 1 System Testing(cont.)

## ✓ Problem.5 Change Location

Select Fee!	Extra Information ...	Password Modity!	Find data!!
Sort data!	Confirm	<	>

첫번째 공간을 선택하세요.  
첫번째 선택 : 1층7번  
두번째를 선택하세요.  
두번째 선택 : 1층14번

Sort data!	Change Location!!	<	>
------------	-------------------	---	---

위치가 바뀌었습니다!

Change Location 화면

# Team 1 System Testing(cont.)

1층2번	1층3번
1층6번	1층7번
1층10번	1층11번
1층14번	1층15번
1층18번	1층19번

1층2번	1층3번
1층6번	1층7번
1층10번	1층11번
1층14번	1층15번
1층18번	1층19번

Sort data! Change Location!! < >

위치가 바뀌었습니다!

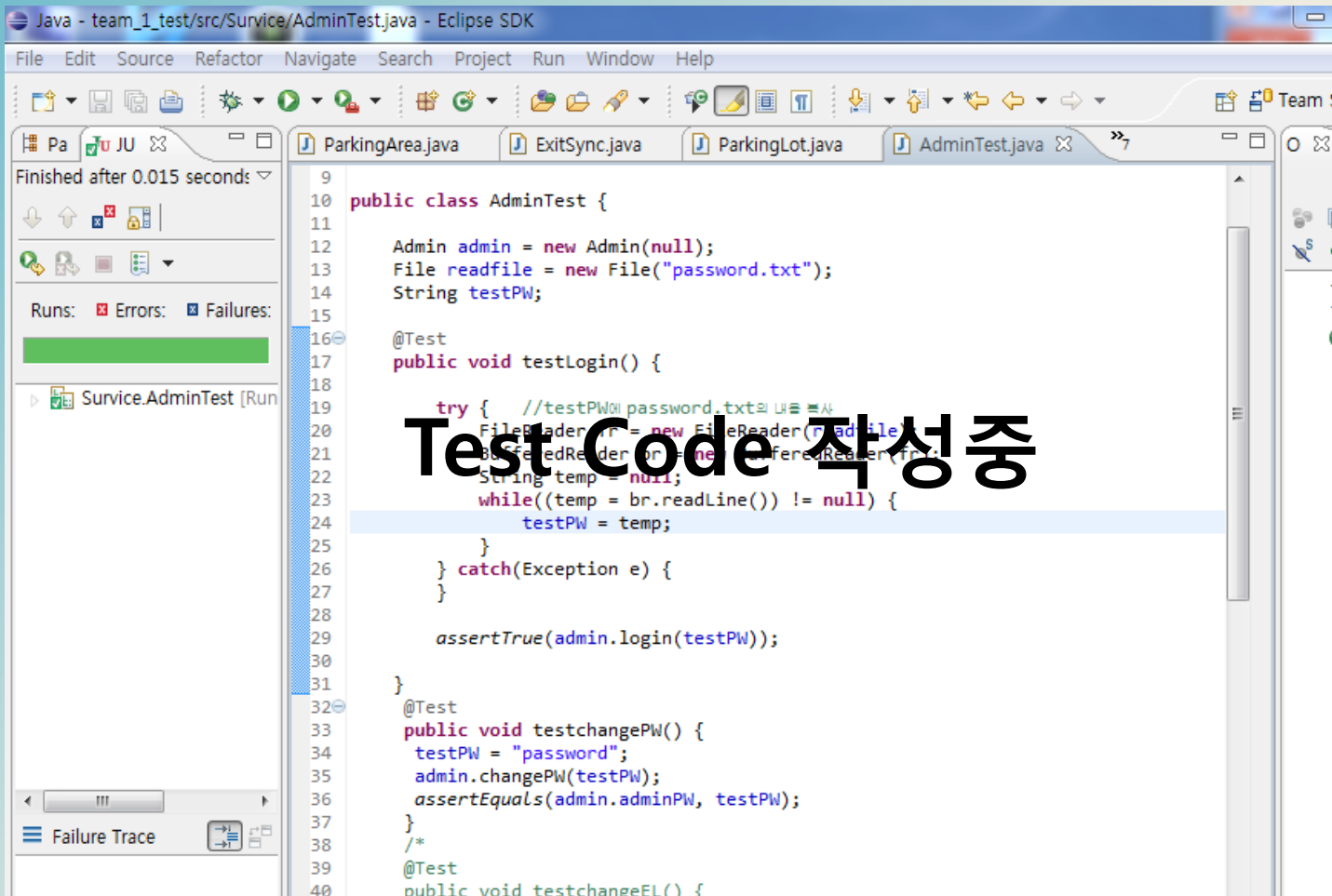
Floor 이동 후 복귀시  
Display 반영되지 않음  
(Program 재실행 후 반영)

Sort data! Change Location!!

위치가 바뀌었습니다!

# Team 1 Test Case Example

## ✓ Test Code



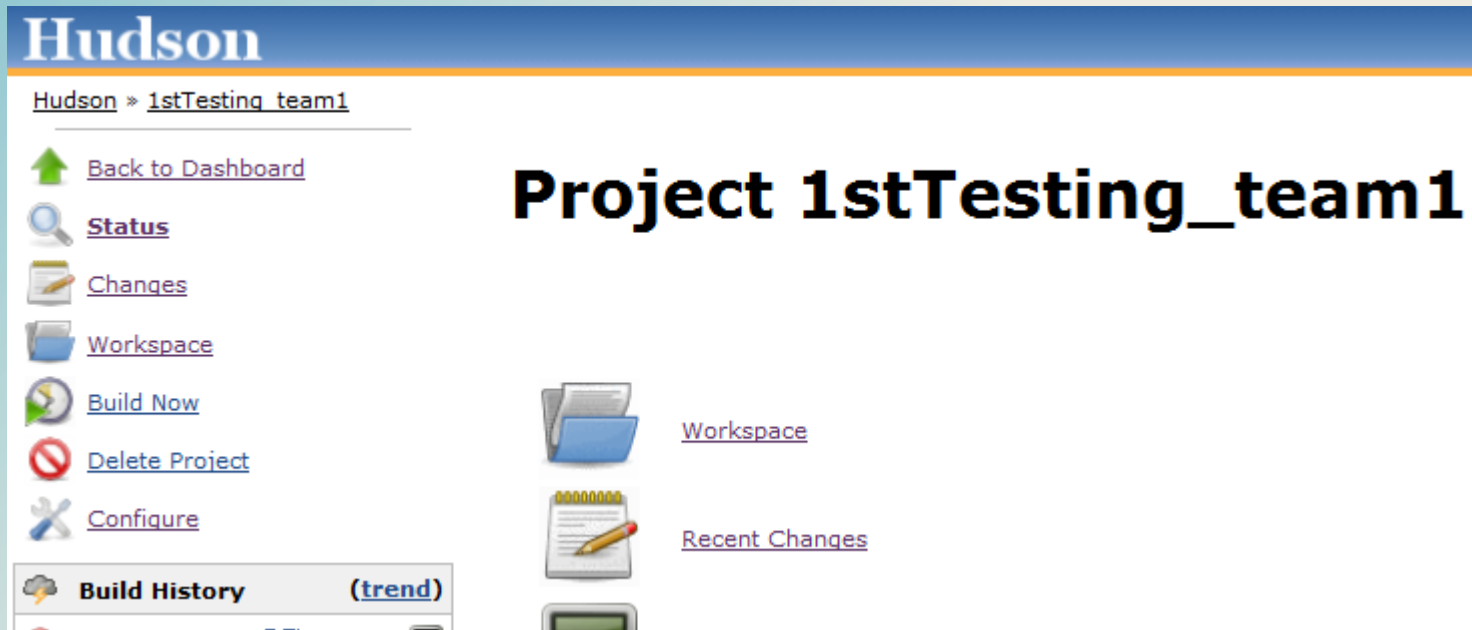
```
Java - team_1_test/src/Service/AdminTest.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Pa JU
Finished after 0.015 seconds:
Runs: Errors: Failures:
Service.AdminTest [Run]
Failure Trace

9
10 public class AdminTest {
11
12     Admin admin = new Admin(null);
13     File readfile = new File("password.txt");
14     String testPW;
15
16     @Test
17     public void testLogin() {
18
19         try { //testPW에 password.txt의 내용 복사
20             BufferedReader br = new BufferedReader(new FileReader(readfile));
21             BufferedReader br = new BufferedReader(new FileReader(readfile));
22             String temp = null;
23             while((temp = br.readLine()) != null) {
24                 testPW = temp;
25             }
26         } catch (Exception e) {
27         }
28
29         assertTrue(admin.login(testPW));
30
31     }
32     @Test
33     public void testchangePW() {
34         testPW = "password";
35         admin.changePW(testPW);
36         assertEquals(admin.adminPW, testPW);
37     }
38     /*
39     @Test
40     public void testchangeEL() {
```

# Team 1 Result Report

## Progressing

➤ Re-building for CTIP Environment..



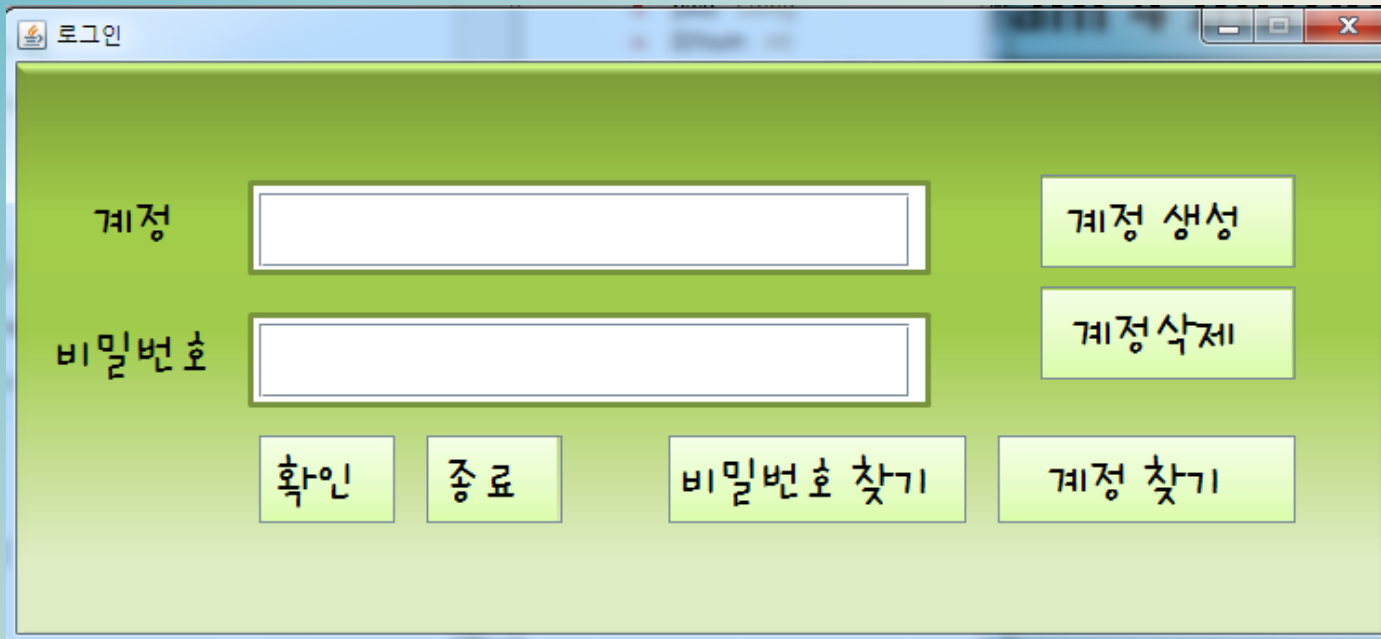
The screenshot shows the Hudson web interface for a project named "1stTesting\_team1". The interface has a blue header with the "Hudson" logo. Below the header, the breadcrumb "Hudson > 1stTesting\_team1" is visible. On the left side, there is a vertical menu with several links: "Back to Dashboard" (with a green arrow icon), "Status" (with a magnifying glass icon), "Changes" (with a notepad icon), "Workspace" (with a folder icon), "Build Now" (with a play button icon), "Delete Project" (with a red prohibition sign icon), and "Configure" (with a wrench icon). At the bottom left, there is a "Build History" section with a "(trend)" link. The main content area on the right features the project name "Project 1stTesting\_team1" in large, bold black text. Below the project name, there are two links: "Workspace" (with a folder icon) and "Recent Changes" (with a notepad and pencil icon).

# 1<sup>st</sup> Testing \_Team4

1. Program Introduction
2. System Testing

# Team 4 Introduction

## ✓ 시작 화면



로그인

계정

비밀번호

계정 생성

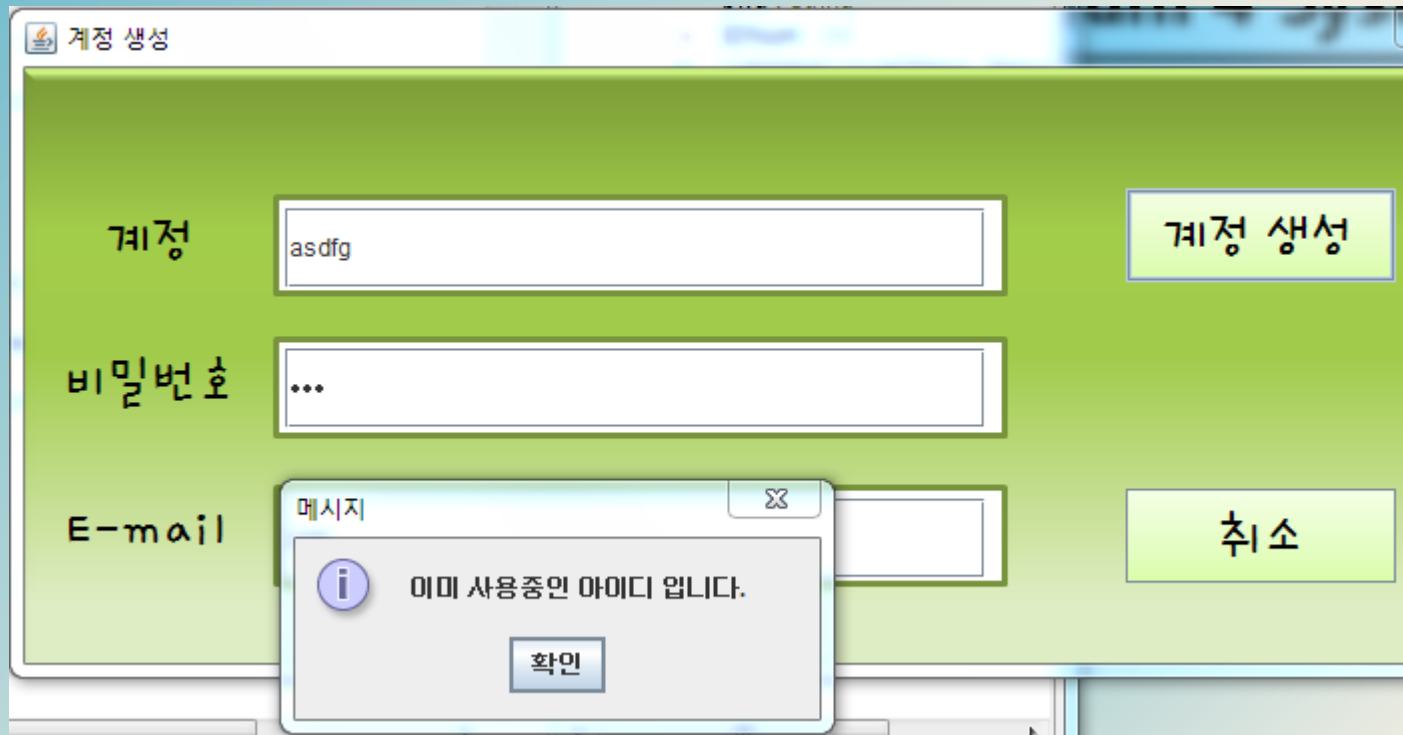
계정 삭제

확인 종료 비밀번호 찾기 계정 찾기

- 각 계정마다 따로 data를 기록하여 개인 물품의 위치 등을 조회하는 System

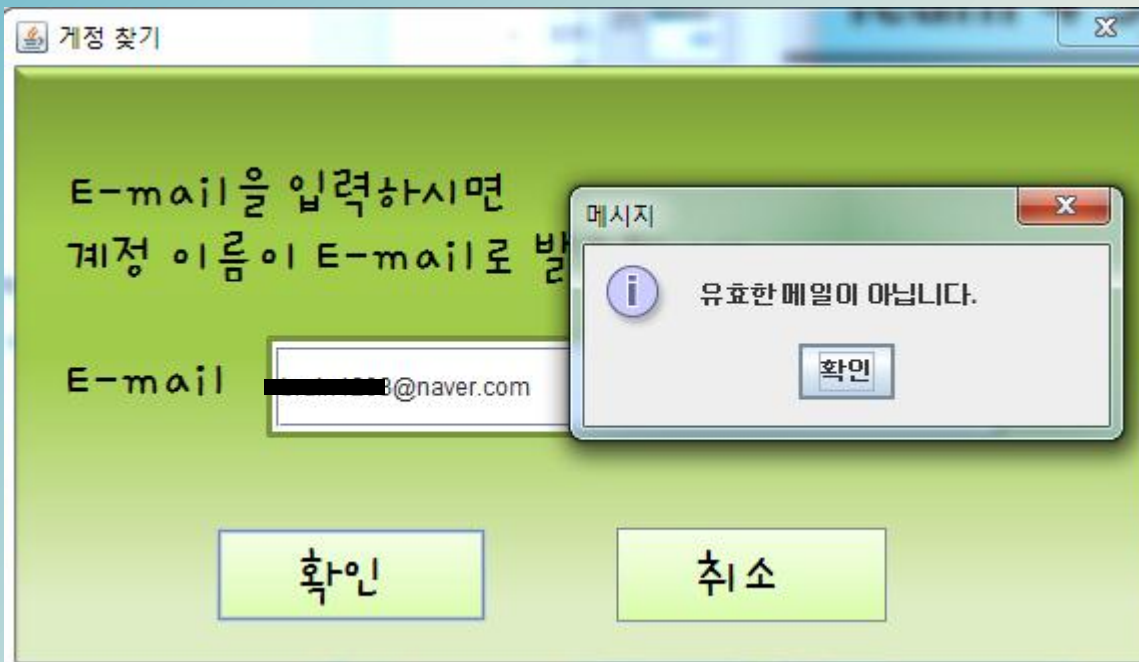
# Team 4 System Testing

## ✓ Problem – 계정 생성 불가



# Team 4 System Testing(cont.)

## ✓ Problem – E-mail 인식 불가





To be continued on 2<sup>nd</sup> Testing..