



# Team5 ThreeS

컴퓨터공학부 201111383 전 훈  
컴퓨터공학부 201311259 권오승  
컴퓨터공학부 201311292 유효상  
컴퓨터공학부 201511266 배윤희

---

---

# INDEX.

1

**PMD**

=====

2

**J  
Depend**

=====

3

**Find  
Bug**

=====

# 1 PMD

## 기본설정

1. 코드 상의 불필요한 부분을 줄이는 것에 초점
2. 빈 코드, 불필요하거나 사용되지 않은 코드, 단순화시킬 수 있는 코드
3. RuleSet에서 필요한 Rule을 부분적으로 선택하여 설정

Rule	Description
UselessParentheses	필요 없는 괄호
UnnecessaryReturn	필요 없는 return
UnusedImports	사용되지 않은 라이브러리
UnusedLocalVariable	사용되지 않은 지역변수
EmptyIfStmt	비어 있는 조건문
SimplifyBooleanExpressions	축약 가능한 Boolean형
CollapsibleIfStatements	하나로 합칠 수 있는 If 중첩문

# 1 PMD

## T7

### Calculate.java

P	Line	created	Rule	Error Message
▶	109	Wed May 31 18:53:49 KST 2017	UselessParentheses	Useless parentheses.

### Input\_ManagerTest.java

P	Line	created	Rule	Error Message
▶	5	Wed May 31 18:53:49 KST 2017	UnusedImports	Avoid unused imports such as 'java.awt.List'
▶	6	Wed May 31 18:53:49 KST 2017	UnusedImports	Avoid unused imports such as 'java.util.ArrayList'

### RV.java

P	Line	created	Rule	Error Message
▶	75	Wed May 31 18:53:49 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions

### TC.java

P	Line	created	Rule	Error Message
▶	15	Wed May 31 18:53:49 KST 2017	UnusedLocalVariable	Avoid unused local variables such as 'tmp_rv'.

# 1 PMD

## T7

### Test.java

P	Line	created	Rule	Error Message
▶	23	Wed May 31 18:53:49 KST 2017	UnusedImports	Avoid unused imports such as 'javax.swing.JRadioButton'
▶	10	Wed May 31 18:53:49 KST 2017	UnusedImports	Avoid unused imports such as 'java.util.ArrayList'
▶	20	Wed May 31 18:53:49 KST 2017	UnusedImports	Avoid unused imports such as 'Package.UI.ActionHandler'

### UI.java

P	Line	created	Rule	Error Message
▶	4	Wed May 31 18:53:49 KST 2017	UnusedImports	Avoid unused imports such as 'java.awt.EventQueue'

# 1 PMD

## T8

### FeedbackController.java

P	Line	created	Rule	Error Message
▶	34	Wed May 31 19:30:10 KST 2017	CollapsibleIfStatements	These nested if statements could be combined
▶	42	Wed May 31 19:30:10 KST 2017	CollapsibleIfStatements	These nested if statements could be combined
▶	29	Wed May 31 19:30:10 KST 2017	CollapsibleIfStatements	These nested if statements could be combined
▶	55	Wed May 31 19:30:10 KST 2017	CollapsibleIfStatements	These nested if statements could be combined
▶	29	Wed May 31 19:30:10 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions
▶	34	Wed May 31 19:30:10 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions

### Main.java

P	Line	created	Rule	Error Message
▶	530	Wed May 31 18:53:48 KST 2017	EmptyIfStmt	Avoid empty if statements
▶	528	Wed May 31 18:53:48 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions

### MainController.java

P	Line	created	Rule	Error Message
▶	49	Wed May 31 18:53:48 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions
▶	45	Wed May 31 18:53:48 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions

# 1 PMD

## T8

### Table.java

P	Line	created	Rule	Error Message
▶	46	Wed May 31 18:53:48 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions
▶	50	Wed May 31 18:53:48 KST 2017	SimplifyBooleanExpressions	Avoid unnecessary comparisons in boolean expressions
▶	1	Wed May 31 18:53:48 KST 2017	UnusedImports	Avoid unused imports such as 'javax.swing.JTable'

### TestCaseController.java

P	Line	created	Rule	Error Message
▶	3	Wed May 31 18:53:48 KST 2017	UnusedImports	Avoid unused imports such as 'java.util.ArrayList'

### TestCaseControllerTest.java

P	Line	created	Rule	Error Message
▶	29	Wed May 31 19:30:10 KST 2017	UnnecessaryReturn	Avoid unnecessary return statements
▶	128	Wed May 31 19:30:10 KST 2017	UnnecessaryReturn	Avoid unnecessary return statements
▶	42	Wed May 31 19:30:10 KST 2017	UnnecessaryReturn	Avoid unnecessary return statements
▶	142	Wed May 31 19:30:10 KST 2017	CollapsibleIfStatements	These nested if statements could be combined
▶	99	Wed May 31 19:30:10 KST 2017	CollapsibleIfStatements	These nested if statements could be combined
▶	138	Wed May 31 19:30:10 KST 2017	UselessParentheses	Useless parentheses.
▶	173	Wed May 31 19:30:10 KST 2017	UselessParentheses	Useless parentheses.
▶	138	Wed May 31 19:30:10 KST 2017	UselessParentheses	Useless parentheses.

---

# 2 JDepend

## 기본설정

---

**CC** : 구체적인 클래스의 숫자

**AC** : 추상클래스 숫자

**C<sub>α</sub>** : 현재 패키지의 클래스를 다른 패키지가 종속(사용)하는 숫자

**C<sub>e</sub>** : 현재 패키지의 클래스가 다른 패키지를 종속(사용)하는 숫자

**A** : 인터페이스와 추상클래스의 개수 - 추상화된 패키지 비율이 높으면 A가 1에 가까워짐

**I** : 1에 가까울수록 불안정한 패키지의 값  $(CA+CE)/(CE)$

**D** : 추상적인 정도와 불안정성 사이의 균형 1에 가까울수록 결합력이 강함



# 2 JDepend

## T7

Package	CC(concr.cl.)	AC(abstr.cl.)	Ca(aff.)	Ce(aff.)	A	I	D	Cycle!
Package	30	0	0	0	0.00	0.00	1.00	

### Stats:

Total Classes: 30  
Concrete Classes: 30  
Abstract Classes: 0

Ca: 0  
Ce: 0

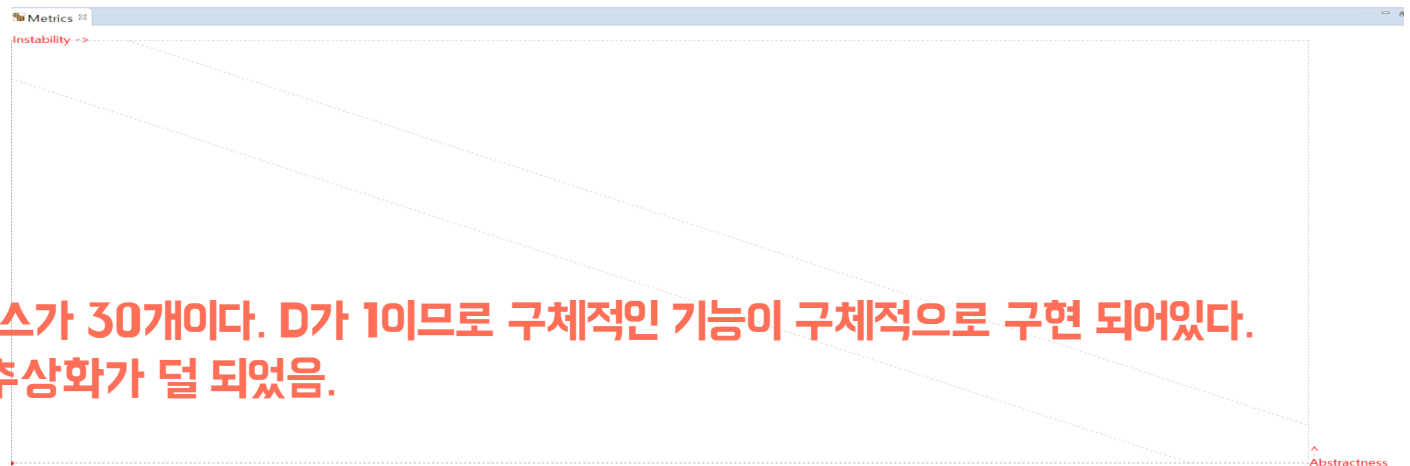
A: 0  
I: 0  
D: 1

### Depends Upon:

Not dependent on any packages.

### Used By:

Not used by any packages.



생성되는 클래스가 30개이다. D가 1이므로 구체적인 기능이 구체적으로 구현 되어있다.  
안정적이면서 추상화가 덜 되었음.

# 2 JDepend

## T8

Package	CC(concr.cl.)	AC(abstr.cl.)	Ca(aff.)	Ce(aff.)	A	I	D	Cycle!
Default	25	0	0	1	0.00	1.00	0.00	
org.junit	0	0	1	0	0.00	0.00	1.00	

JDepend

Depends Upon:  
org.junit

- Package: Default

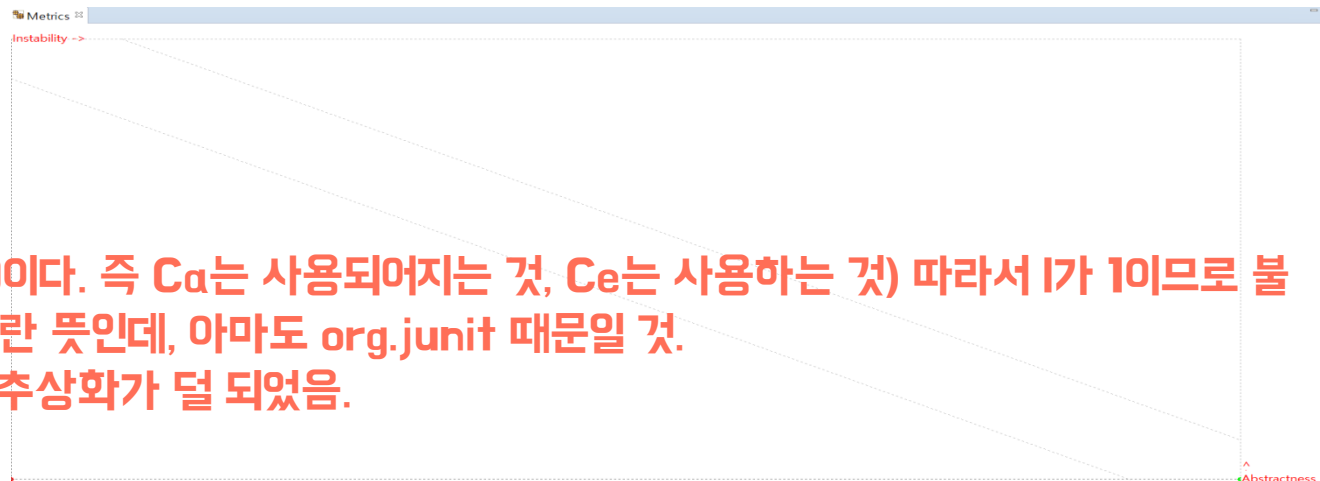
Stats:

Total Classes: 25  
Concrete Classes: 25  
Abstract Classes: 0

Used By:  
Not used by any packages.

Ca: 0  
Ce: 1

A: 0  
I: 1  
D: 0



Ca랑 Ce는 0이다. 즉 Ca는 사용되어지는 것, Ce는 사용하는 것) 따라서 I가 1이므로 불안정한 패키지란 뜻인데, 아마도 org.junit 때문일 것.  
안정적이면서 추상화가 덜 되었음.

---

# 3 FindBug

## T7

---

### 7조 분석 결과

FindBug를 사용하여 분석한 결과 아무 warning도 발생하지 않아서인지, 버그리포팅 파일이 생성되지 않았다.

# 3 FindBug

## T7

### 8조 분석 결과

FindBug 를 사용하여 분석한 결과, 총 2개의 warning이 발견 되었다.

```
> SMA2017_FeesualCPT (2) [SMA2017_FeesualCPT master ↑ 2]
  ▾ Scariest (1)
    ▾ High confidence (1)
      ▾ Call to equals() comparing different types (1)
        Call to java.util.ArrayList<java.lang.String>.equals(java.util.ArrayList<SingleTestCase>) in Test
    ▾ Troubling (1)
      ▾ Normal confidence (1)
        ▾ Comparison of String objects using == or != (1)
          Comparison of String objects using == or != in TestCaseController.makeTc(Table) [Troubling(
TestCaseTest.java: 50
Navigation
Call to java.util.ArrayList<java.lang.String>.equals(java.util.ArrayList<SingleTestCase>) in TestCaseTest.aetSinaTcListTest1()
Actual tvpe java.util.ArrayList<SingleTestCase>
Expected java.util.ArrayList<java.lang.String>
Return value of TestCase.aetSinaTcList() of tvpe java.util.ArrayList
Value loaded from stlist
java.util.ArrayList.equals(Object) used to determine equality
```

**Bug:** Call to java.util.ArrayList<java.lang.String>.equals(java.util.ArrayList<SingleTestCase>) in TestCaseTest.getSingTcListTest1()

This method calls equals(Object) on two references of different class types and analysis suggests they will be to objects of different classes at runtime. Further, examination of the equals methods that would be invoked suggest that either this call will always return false, or else the equals method is not be symmetric (which is a property required by the contract for equals in class Object).

**Rank:** Scariest (1), **confidence:** High  
**Pattern:** EC\_UNRELATED\_TYPES  
**Type:** EC, **Category:** CORRECTNESS (Correctness)



**감사합니다.**

---

---