

Static Analysis

- T6, T7

Project Team

T8

Date

2017-06-01

T8 Team Information

201211347 박성근

201211376 임제현

201411270 김태홍

Table of Contents

1. T6 Static Analysis
 - 1.1. Level1 PMD
 - 1.2. Level2 Eclipse Metrics Plugin
 - 1.3. Level3 FindBugs
2. T7 Static Analysis
 - 2.1. Level1 PMD
 - 2.2. Level2 Eclipse Metrics Plugin
 - 2.3. Level3 FindBugs

1. T6 Static Analysis

1.1. Level 1 PMD

<package WIN2>

Element	# Violations	# Violations/KLOC	# Violations/Method	Project
WIN2	20	50.5	1.43	T6_source
Window2.java	9	37.7	1.80	T6_source
VariableNamingConventions	7	29.3	1.40	T6_source
MethodNamingConventions	2	8.4	0.40	T6_source
Filter.java	8	92.0	1.33	T6_source
ConstructorCallsOverridableMethod	3	34.5	0.50	T6_source
VariableNamingConventions	4	46.0	0.67	T6_source
MethodNamingConventions	1	11.5	0.17	T6_source
CaseComponentSet.java	3	42.9	1.00	T6_source
VariableNamingConventions	2	28.6	0.67	T6_source
MethodNamingConventions	1	14.3	0.33	T6_source

<Window2.java>

▶ 194	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶ 31	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'f_rv_Panel' is not final.
▶ 90	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'scrollPane_3' is not final.
▶ 82	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'scrollPane_2' is not final.
▶ 24	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'f_property' is not final.
▶ 29	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'f_pro_Panel' is not final.
▶ 208	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶ 25	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'f_value' is not final.
▶ 30	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix). 'b_workReturn' is not final.

주로 명명 문제에서 많은 에러가 발생하였다.

```

194 public void set_property(){
195     // 이거 그 filter property 배치하는 함수
196     // 여기서 이를 설정하는 것만 property list 받아서 설정하면 됨
197

```

MethodNamingConventions : 메소드명에 언더바(_)가 들어가 있으면 안된다.

```

30 private JButton b_workReturn;
31 private JPanel f_rv_Panel;
32

```

VariableNamingConventions : final이 아닌 변수에는 언더바(_)가 들어가 있으면 안된다.

<Filter.java>

P	Line	created	Rule	Error Message
	121	Tue May 30...	VariableNamingConventions	Variables should start with a lowercase character, 'T' starts with uppercase character.
▶	57	Tue May 30...	ConstructorCallsOverridableMethod	Overridable method 'GUInit' called during object construction
▶	21	Tue May 30...	VariableNamingConventions	Variables should start with a lowercase character, 'Pro_Val' starts with uppercase character.
▶	37	Tue May 30...	ConstructorCallsOverridableMethod	Overridable method 'GUInit' called during object construction
▶	21	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix), 'Pro_Val' is not final.
▶	64	Tue May 30...	MethodNamingConventions	Method names should not start with capital letters
▶	47	Tue May 30...	ConstructorCallsOverridableMethod	Overridable method 'GUInit' called during object construction
▶	22	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix), 'check_filter' is not final.

```

56         this.selected = selected;
57         GUIinit(master, name);
58         this.pindex = pindex;
59         this.type = "Property";

```

ConstructorCallsOverridableMethod : 오버라이드 가능한 메소드를 호출하게 되면 불완전한 객체로 메소드가 호출될 위험이 있어, 디버그하기 어려워진다.

<CaseComponentSet.java>

▶	45	Tue May 30...	MethodNamingConventions	Method names should not start with capital letters
▶	24	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix), 'check_true_false' is not final.
▶	94	Tue May 30...	VariableNamingConventions	Variables should start with a lowercase character, 'T' starts with uppercase character.

명명에러 밖에 없다.

<package WIN1>

Element	# Violations	# Violations/KLOC	# Violations/Method	Project
WIN1	10	16.3	0.77	T6_source
Window1.java	6	16.1	2.00	T6_source
VariableNamingConventions	6	16.1	2.00	T6_source
CstComponentSet.java	2	17.9	0.40	T6_source
VariableNamingConventions	1	8.9	0.20	T6_source
MethodNamingConventions	1	8.9	0.20	T6_source
ComponentSet.java	2	15.6	0.40	T6_source
VariableNamingConventions	2	15.6	0.40	T6_source

명명에러 밖에 없다.

<package TEST>

Element	# Violations	# Violations/KLOC	# Violations/Method	Project
TEST	17	114.1	1.00	T6_source
Task_test.java	6	125.0	1.00	T6_source
MethodNamingConventions	6	125.0	1.00	T6_source
RepValue_test.java	5	96.2	1.00	T6_source
MethodNamingConventions	5	96.2	1.00	T6_source
Property_testTest.java	3	120.0	1.00	T6_source
MethodNamingConventions	3	120.0	1.00	T6_source
Category_test.java	3	125.0	1.00	T6_source
MethodNamingConventions	3	125.0	1.00	T6_source

명명에러 밖에 없다.

<package EXTRA>

Element	# Violations	# Violations/KLOC	# Violations/Method	Project
EXTRA	9	20.3	0.36	T6_source
WinHelp.java	2	19.2	1.00	T6_source
VariableNamingConventions	1	9.6	0.50	T6_source
AvoidLosingExceptionInformation	1	9.6	0.50	T6_source
Window3.java	5	74.6	1.67	T6_source
VariableNamingConventions	5	74.6	1.67	T6_source
Bonus.java	2	16.7	0.33	T6_source
VariableNamingConventions	2	16.7	0.33	T6_source

<WinHelp.java>

P Line	created	Rule	Error Message
70	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix), 'scrollPane_2' is not final.
137	Tue May 30...	AvoidLosingExceptionInformation	Avoid statements in a catch block that invoke accessors on the exception without using the information

```

135         }catch(IOException ie)
136         {
137             ie.getMessage();
138         }

```

AvoidLosingExceptioninformation : getMessage한 것을 사용하지 않고 있다.

<package CPT>

Element	# Violations	# Violations/KLOC	# Violations/Method	Project
▼ CPT	43	42.0	0.39	T6_source
▼ Task.java	28	49.5	0.57	T6_source
▶ AvoidLosingExceptionInformation	2	3.5	0.04	T6_source
▶ SystemPrintln	6	10.6	0.12	T6_source
▶ VariableNamingConventions	2	3.5	0.04	T6_source
▶ MethodNamingConventions	18	31.8	0.37	T6_source
▼ Result.java	2	64.5	0.40	T6_source
▶ ProperCloneImplementation	1	32.3	0.20	T6_source
▶ MethodNamingConventions	1	32.3	0.20	T6_source
▼ RepValue.java	5	31.4	0.26	T6_source
▶ MethodNamingConventions	5	31.4	0.26	T6_source
▼ Property.java	3	41.1	0.30	T6_source
▶ MethodNamingConventions	3	41.1	0.30	T6_source
▼ Constraints.java	2	28.2	0.33	T6_source
▶ MethodNamingConventions	2	28.2	0.33	T6_source
▼ Category.java	3	68.2	0.27	T6_source
▶ MethodNamingConventions	3	68.2	0.27	T6_source

<Task.java>

P	Line	created	Rule	Error Message
▶	516	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	137	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	104	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	165	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	482	Tue May 30...	MethodNamingConventions	Method names should not start with capital letters
▶	26	Tue May 30...	VariableNamingConventions	Only variables that are final should contain underscores (except for underscores in standard prefix/suffix), 'testCaseNumber_filtered' is not final.
▶	186	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	79	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	109	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	74	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	151	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	283	Tue May 30...	SystemPrintln	System.out.println is used
▶	436	Tue May 30...	SystemPrintln	System.out.println is used
▶	682	Tue May 30...	AvoidLosingExceptionInformation	Avoid statements in a catch block that invoke accessors on the exception without using the information
▶	262	Tue May 30...	SystemPrintln	System.out.println is used
▶	564	Tue May 30...	AvoidLosingExceptionInformation	Avoid statements in a catch block that invoke accessors on the exception without using the information
▶	684	Tue May 30...	SystemPrintln	System.out.println is used
▶	603	Tue May 30...	SystemPrintln	System.out.println is used
▶	313	Tue May 30...	SystemPrintln	System.out.println is used

```

435     {
436     System.out.println("work");
437 }
    
```

SystemPrintln : 디버그를 위해 남겨둔 System.out.println 이 안지워지고 남아있다.

<Result.java>

P	Line	created	Rule	Error Message
▶	22	Tue May 30...	MethodNamingConventions	Method names should not contain underscores
▶	48	Tue May 30...	ProperCloneImplementation	Object clone() should be implemented with super.clone()

```
48 public Result clone()
```

ProperCloneImplementation : clone을 사용할때는 super.clone()으로 사용하여야 한다.

<CPD Test>

```
> 3 | src.WIN0.Filter | src.WIN1.CstComponent
> 5 | src.WIN0.Filter | src.WIN1.Component
> 1 | src.WIN2.Window2 | src.WIN2.Windc
> 3 | src.WIN2.Window2 | src.WIN1.Windc
> 2 | src.EXTRA.Bonus | src.EXTRA.Boi
> 3 | src.TEST.Task_test | src.TEST.Task_I
> 2 | src.EXTRA.Window3 | src.EXTRA.WinH
> 3 | src.WIN2.Window2 | src.WIN1.Windc
> 3 | src.WIN2.Window2 | src.WIN1.Windc
> 3 | src.WIN2.Window2 | src.WIN1.Windc
> 3 | src.EXTRA.Bonus | src.EXTRA.Boi
> 5 | src.EXTRA.Bonus | src.EXTRA.Boi
> 1 | src.EXTRA.ArcDesign | src.EXTRA.ArcDes
> 1 | src.WIN1.Window1 | src.WIN1.Windc
> 5 | src.WIN2.Window2 | src.WIN1.Windc
> 3 | src.EXTRA.Bonus | src.EXTRA.Boi
> 1 | src.EXTRA.Bonus | src.EXTRA.Boi
> 1 | src.EXTRA.Bonus | src.EXTRA.Boi
> 6 | src.CPT.Task | src.CPT.T
> 6 | src.TEST.Task_test | src.TEST.Task_I
> 5 | src.TEST.Task_test | src.TEST.Task_I
> 5 | src.TEST.RepValue_test | src.TEST.RepValue_I
> 9 | src.CPT.Task | src.CPT.T
> 4 | src.WIN1.Window1 | src.WIN1.Windc
> 2 | src.EXTRA.Bonus | src.EXTRA.Boi
> 5 | src.TEST.RepValue_test | src.TEST.RepValue_I
> 5 | src.TEST.Property_test | src.TEST.Property_TestI
> 2 | src.EXTRA.Bonus | src.EXTRA.Boi
> 3 | src.EXTRA.Bonus | src.EXTRA.Boi
> 2 | src.EXTRA.Bonus | src.EXTRA.Boi
> 1 | src.WIN1.Window1 | src.WIN1.Windc
> 1 | src.WIN1.Window1 | src.WIN1.Windc
> 7 | src.CPT.Task | src.CPT.T
> 6 | src.TEST.Category_test | src.TEST.Category_I
```

이와 같이 코드 중복되는 내용을 확인하였고 자세한 세부사항은

```
1 Found a 27 line (81 tokens) duplication in the following files:
2 Starting at line 76 of C:\Users\sunggeun\workspace\T6_source\src\CPT\Property.java
3 Starting at line 174 of C:\Users\sunggeun\workspace\T6_source\src\CPT\RepValue.java
4
5 public CPT.RepValue getConst(int cstindex, String type)
6 {
7     switch(type)
8     {
9         case "Property":
10            return cstPropList.get(cstindex);
11         case "IFProperty":
12            return cstIfPropList.get(cstindex);
13         default:
14            return null;
15     }
16 }
17
18 public int getCstSize(String type)
19 {
20     switch(type)
21     {
22         case "Property":
23            return cstPropList.size();
24         case "IFProperty":
25            return cstIfPropList.size();
26         default:
27            return -1;
28     }
29 }
30
31 public ArrayList<RepValue> getCstList(String type)
32 ~~
```

이런식으로 아래의 27line이 Property.java의 76번째 줄, RepValue.java의 174번째 줄에서 나타난다고 알려줍니다.

1.2. Level 2 Eclipse Metrics Plugin

McCabe Cyclomatic Complexity: 프로그램 복잡도를 나타내는 software 단위. 높으면 높을수록 복잡하다.

Number of Parameters: 매개변수의 수

Nested Block Depth: 중첩된 {}의 개수

Total: 합계 / Mean: 평균 / Std. Dev.: 표준편차 / Maximum: 최대값

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▼ McCabe Cyclomatic Complexity (avg/max per method)	3.073	3.819		23	/T6_source/src/CPT/Task.java	load_work
▶ CPT	2.844	3.577		23	/T6_source/src/CPT/Task.java	load_work
▶ WIN1	7.692	7.342		21	/T6_source/src/WIN1/ComponentSet.java	actionPerformed
▶ EXTRA	2.52	2.352		10	/T6_source/src/EXTRA/ArcDesign.java	update
▶ WIN2	2.786	1.473		6	/T6_source/src/WIN2/Window2.java	updatePanel
▶ TEST	2.059	1.162		4	/T6_source/src/TEST/Category_test.java	testModify_rep_value
▼ Number of Parameters (avg/max per method)	1.051	1.474		12	/T6_source/src/EXTRA/ArcDesign.java	ArcDesign
▶ EXTRA	1.32	2.362		12	/T6_source/src/EXTRA/ArcDesign.java	ArcDesign
▶ CPT	0.862	1.053		6	/T6_source/src/CPT/Task.java	modify_const
▶ WIN1	2.154	1.46		5	/T6_source/src/WIN1/ComponentSet.java	ComponentSet
▶ WIN2	2.286	1.623		5	/T6_source/src/WIN2/CaseComponent...	CaseComponentSet
▶ TEST	0	0		0	/T6_source/src/TEST/Category_test.java	testNew_rep_value
▼ Nested Block Depth (avg/max per method)	1.517	1.023		6	/T6_source/src/CPT/Task.java	workContainValue
▶ CPT	1.495	1.146		6	/T6_source/src/CPT/Task.java	workContainValue
▶ EXTRA	1.56	0.941		4	/T6_source/src/EXTRA/WinHelp.java	actionPerformed
▶ WIN1	1.769	0.973		4	/T6_source/src/WIN1/ComponentSet.java	actionPerformed
▶ TEST	1.353	0.478		2	/T6_source/src/TEST/Category_test.java	testNew_rep_value
▶ WIN2	1.571	0.495		2	/T6_source/src/WIN2/CaseComponent...	PutPanel
▶ Afferent Coupling (avg/max per packageFragment)	5	5.55		15	/T6_source/src/CPT	
▶ Efferent Coupling (avg/max per packageFragment)	4.4	1.497		7	/T6_source/src/CPT	
▶ Instability (avg/max per packageFragment)	0.617	0.243		1	/T6_source/src/TEST	
▶ Abstractness (avg/max per packageFragment)	0.029	0.057		0.143	/T6_source/src/EXTRA	
▶ Normalized Distance (avg/max per packageFragment)	0.354	0.225		0.682	/T6_source/src/CPT	
▶ Depth of Inheritance Tree (avg/max per type)	2.083	1.913		6	/T6_source/src/CPT/MainFrame.java	
▶ Weighted methods per Class (avg/max per type)	547	22.792	32.811	168	/T6_source/src/CPT/Task.java	
▶ Number of Children (avg/max per type)	2	0.083	0.4	2	/T6_source/src/EXTRA/Caller.java	
▶ Number of Overridden Methods (avg/max per type)	3	0.125	0.439	2	/T6_source/src/EXTRA/Bonus.java	
▶ Lack of Cohesion of Methods (avg/max per type)	0.362	0.275	0.865		/T6_source/src/CPT/Task.java	
▶ Number of Attributes (avg/max per type)	147	6.125	6.153	26	/T6_source/src/WIN1/Window1.java	
▶ Number of Static Attributes (avg/max per type)	6	0.25	0.878	4	/T6_source/src/EXTRA/Icon.java	
▶ Number of Methods (avg/max per type)	175	7.292	9.467	49	/T6_source/src/CPT/Task.java	
▶ Number of Static Methods (avg/max per type)	3	0.125	0.439	2	/T6_source/src/EXTRA/Icon.java	
▶ Specialization Index (avg/max per type)		0.05	0.202	1	/T6_source/src/EXTRA/Bonus.java	
▶ Number of Classes (avg/max per packageFragment)	24	4.8	1.833	7	/T6_source/src/CPT	
▶ Number of Interfaces (avg/max per packageFragment)	1	0.2	0.4	1	/T6_source/src/EXTRA	
▶ Number of Packages	5					
▶ Total Lines of Code	3111					
▶ Method Lines of Code (avg/max per method)	2207	12.399	21.39	178	/T6_source/src/WIN1/Window1.java	Window1

<전체 프로젝트>

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▼ McCabe Cyclomatic Complexity (avg/max per method)		2.844	3.577	23	/T6_source/src/CPT/Task.java	load_work
▼ Task.java		3.429	4.371	23	/T6_source/src/CPT/Task.java	load_work
▼ Task		3.429	4.371	23	/T6_source/src/CPT/Task.java	load_work
load_work	23					
modify_const	18					
save_work	10					
wrapValue	9					
workContainValue	9					
sort	9					
calculate	9					
remove_rep_value	5					
isContains	5					

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.844	3.577	23	/T6_source/src/CPT/Task.java	load_work
▶ Task.java		3.429	4.371	23	/T6_source/src/CPT/Task.java	load_work
▼ RepValue.java		2.947	3.692	17	/T6_source/src/CPT/RepValue.java	isCompatible
▼ RepValue		2.947	3.692	17	/T6_source/src/CPT/RepValue.java	isCompatible
isCompatible	17					
remove_const	7					
add_const	4					
remove_const	4					

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.844	3.577	23	/T6_source/src/CPT/Task.java	load_work
▼ Number of Parameters (avg/max per method)		0.862	1.053	6	/T6_source/src/CPT/Task.java	modify_const
▶ Task.java		1.041	1.277	6	/T6_source/src/CPT/Task.java	modify_const
▼ Task		1.041	1.277	6	/T6_source/src/CPT/Task.java	modify_const
modify_const	6					
remove_const	4					
check_filter	4					
modify_rep_value	3					
remove_const	3					
modify_category	2					

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.844	3.577	23	/T6_source/src/CPT/Task.java	load_work
▼ Number of Parameters (avg/max per method)		0.862	1.053	6	/T6_source/src/CPT/Task.java	modify_const
▼ Nested Block Depth (avg/max per method)		1.495	1.146	6	/T6_source/src/CPT/Task.java	workContainValue
▶ Task.java		1.898	1.515	6	/T6_source/src/CPT/Task.java	workContainValue
▼ Task		1.898	1.515	6	/T6_source/src/CPT/Task.java	workContainValue
workContainValue	6					
sort	6					
load_work	6					
wrapValue	5					
save_work	5					
remove_rep_value	4					

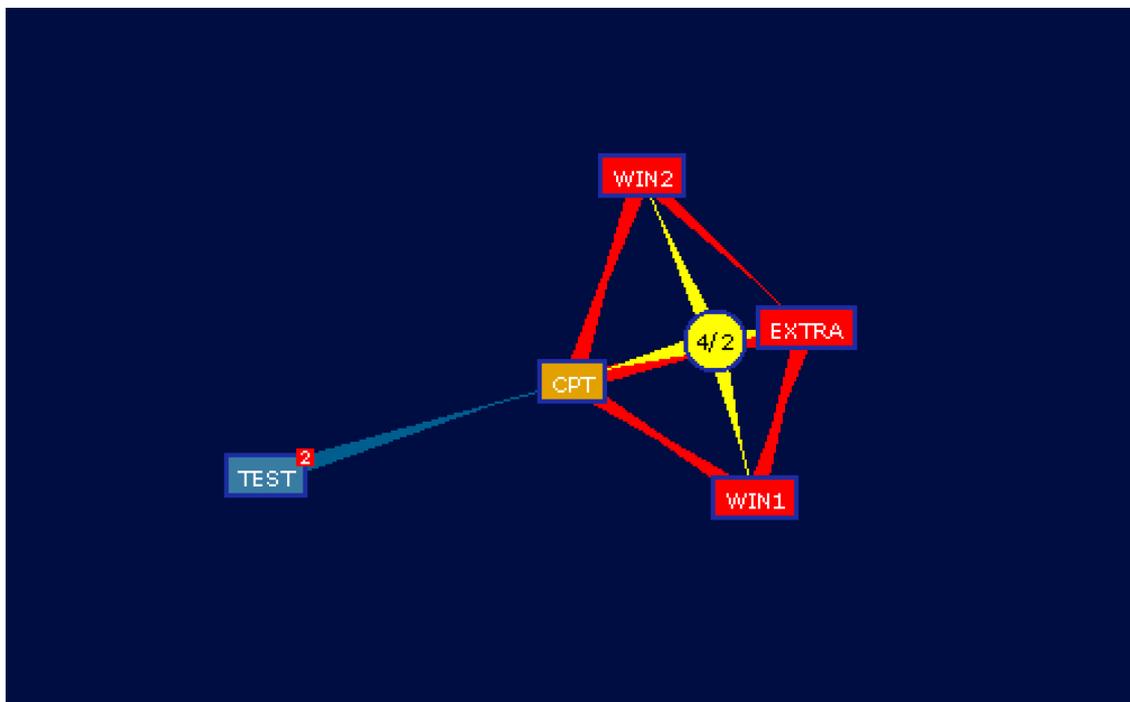
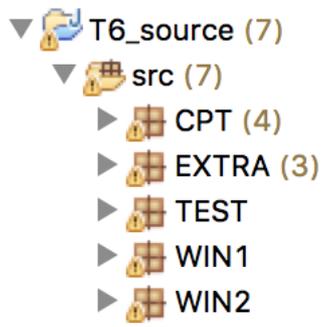
<package CPT>

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▶ McCabe Cyclomatic Complexity (avg/max per method)		2.52	2.352	10	/T6_source/src/EXTRA/ArcDesign.java	update
▼ Number of Parameters (avg/max per method)		1.32	2.362	12	/T6_source/src/EXTRA/ArcDesign.java	ArcDesign
▼ ArcDesign.java		4.333	5.437	12	/T6_source/src/EXTRA/ArcDesign.java	ArcDesign
▼ ArcDesign		4.333	5.437	12	/T6_source/src/EXTRA/ArcDesign.java	ArcDesign
ArcDesign	12					
paint	1					
update	0					
▶ Icon.java		2	2	4	/T6_source/src/EXTRA/Icon.java	makeColorRGBA
▶ FileSelector.java		0.571	0.728	2	/T6_source/src/EXTRA/FileSelector.java	FileSelector
▶ Window3.java		1.667	0.471	2	/T6_source/src/EXTRA/Window3.java	Window3

<package EXTRA>

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing	Maximum	Method
▼ McCabe Cyclomatic Complexity (avg/max per method)		7.692	7.342	21	/T6_source/src/WIN1/ComponentSet.java		actionPerformed
▼ ComponentSet.java		6.8	7.521	21	/T6_source/src/WIN1/ComponentSet.java		actionPerformed
▼ ComponentSet		6.8	7.521	21	/T6_source/src/WIN1/ComponentSet.java		actionPerformed
actionPerformed	21						
yes	8						
ComponentSet	2						
putPanel	2						
no	1						
▼ Window1.java	12.333	8.055		19	/T6_source/src/WIN1/Window1.java		updatePanel
▼ Window1	12.333	8.055		19	/T6_source/src/WIN1/Window1.java		updatePanel
updatePanel	19						
actionPerformed	17						
Window1	1						
▼ CstComponentSet.java	5.8	5.231		16	/T6_source/src/WIN1/CstComponentS...		actionPerformed
▼ CstComponentSet	5.8	5.231		16	/T6_source/src/WIN1/CstComponentS...		actionPerformed
actionPerformed	16						
CstComponentSet	4						
CstComponentSet	4						
putPanel	4						
GUInit	1						
▶ Number of Parameters (avg/max per method)	2.154	1.46		5	/T6_source/src/WIN1/ComponentSet.java		ComponentSet
▶ Nested Block Depth (avg/max per method)	1.769	0.973		4	/T6_source/src/WIN1/ComponentSet.java		actionPerformed
▶ Different Counting	2						

<package WIN1>



<Dependency Graph>

1.3. Level 3 FindBugs

The top screenshot shows the Eclipse IDE with the FindBugs plugin. The Bug Explorer on the left lists several bugs. One bug is highlighted: "Possible null pointer dereference of put in CPT.Task.save_work() on exception path [Troubling(11), Normal confidence]". The main editor shows the source code of WinHelp.java, with the line `in.close();` highlighted at line 141.

The bottom screenshot shows the same IDE with the bug selected. The Bug Info panel on the right displays the following properties:

Property	Value
BUG	
BUGTYPE	NP_A
CONFIDENCE	High
Creation time	Tue 8
DETECTOR_PLUGIN_ID	edu.u
FINDBUGS_UNIQUE_ID	6cad
FIRST_VERSION	-1
lineNumber	568
Marker id	1386
Marker type	Null p
message	
PATTERN_TYPE	NP
PRIMARY_LINE	568
RANK	7
Resource	orig.e
severity	1
UNIQUE_JAVA_ID	=CPT

<Scary>

The screenshot shows the Eclipse IDE interface. On the left, the Bug Explorer displays a tree of bugs under the 'CPT' package. The selected bug is 'CPT.Result defines clone() but doesn't implement Cloneable' with a confidence of 'High' and a status of 'Troubling(14)'. The main editor shows the source code of the 'Result' class, with the 'clone()' method highlighted. The 'Bug info' panel on the right provides details for this bug, including its ID, creation time, and severity.

```
public void checkO{
    // GUI
    this.caseFlag = !caseFlag;
}

// encapsulation
public boolean getCaseFlag()
{
    return caseFlag;
}

public Result clone()
{
    Result cResult = new Result();
    cResult.combList = (ArrayList) this.combList.clone();
    cResult.combListProp = (ArrayList) this.combListProp.clone();
    return cResult;
}
```

The screenshot shows the Eclipse IDE interface. On the left, the Bug Explorer displays a tree of bugs under the 'EXTRA' package. The selected bug is 'Check for address that won't work for negative numbers' with a confidence of 'Normal' and a status of 'Troubling(13)'. The main editor shows the source code of the 'Bonus' class, with a loop containing a conditional check highlighted. The 'Bug info' panel on the right provides details for this bug, including its ID, creation time, and severity.

```
public Bonus(Task T)
{
    this.T = T;
    thread = new Thread(this);
    delay = 18;
}

ad2 = new ArrayList<ArcDesign>();
for(int i=0;i<8;i++)
{
    if(i%2 == 1)
    {
        ad2.add(new ArcDesign(247,217,115,118,4,0,0,false,45));
        ad2.add(new ArcDesign(247,217,95,3,4,0,0,false,45));
    }
    else
    {
        ad2.add(new ArcDesign(247,217,115,118,4,0,0,false,45));
        ad2.add(new ArcDesign(247,217,95,3,4,0,0,false,45));
    }
}
ad2.add(new ArcDesign(247,217,143,2,4,0,1,true,240,110,1));
ad2.add(new ArcDesign(247,217,143,2,4,0,1,true,150,30,30));
```

The screenshot shows the Eclipse IDE interface. On the left, the Bug Explorer displays a tree of bugs. The selected bug is "Possible null pointer dereference of in in EXTRA.WinHelp.actionPerformed(ActionEvent) on exception path [Troubling(11), Normal confidence]". The main editor shows the source code for WinHelp.java, with line 141 highlighted: `in.close();`. The Bug Info panel on the right shows the following properties:

Property	Value
BUGTYPE	NP_A
CONFIDENCE	High
Creation time	Tue 8
DETECTOR_PLUGIN_ID	edu.u
FINDBUGS_UNIQUE_ID	7916
FIRST_VERSION	-1
lineNumber	141
Marker id	1386
Marker type	edu.u
message	Null p
PATTERNTYPE	NP
PRIMARY_LINE	141
RANK	7
Resource	orig.e
severity	1
UNIQUE_JAVA_ID	=CPT

The screenshot shows the Eclipse IDE interface. On the left, the Bug Explorer displays a tree of bugs. The selected bug is "Possible null pointer dereference of put in CPT.Task.save_work() on exception path [Troubling(11), Normal confidence]". The main editor shows the source code for Task.java, with line 568 highlighted: `put.close();`. The Bug Info panel on the right shows the following properties:

Property	Value
BUGTYPE	NP_A
CONFIDENCE	High
Creation time	Tue 8
DETECTOR_PLUGIN_ID	edu.u
FINDBUGS_UNIQUE_ID	6cad
FIRST_VERSION	-1
lineNumber	568
Marker id	1386
Marker type	edu.u
message	Null p
PATTERNTYPE	NP
PRIMARY_LINE	568
RANK	7
Resource	orig.e
severity	1
UNIQUE_JAVA_ID	=CPT

<Troubling>

The screenshot displays the Eclipse IDE interface with the FindBugs tool. The Bug Explorer on the left shows a tree of detected issues. The selected issue is 'Dead store to tempRepvList in CPT.Task.sort()' with a confidence of 'High' and a concern level of 'Concern(15)'. The central editor shows the Java code for the `sort()` method in `CPT.Task`. The Bug Info panel on the right provides a detailed view of the selected bug, including its properties:

Property	Value
Bug	
BUGTYPE	DLS_
CONFIDENCE	High
Creation time	Tue 8
DETECTOR_PLUGIN_ID	edu.k
FINDBUGS_UNIQUE_ID	6e52
FIRST_VERSION	-1
lineNumber	388
Marker id	1386
Marker type	edu.k
message	Dead
PATTERNTYPE	DLS
PRIMARY_LINE	388
RANK	15
Resource	orig.e
severity	1
UNIQUE_JAVA_ID	=CPT

<Concern>

2. T7 Static Analysis

2.1. Level 1 PMD

Element	# Violations	# Violations/KLOC	# Violations/Method	Project
Package	359	164.2	5.61	SoftwareMo...
UI.java	26	178.1	13.00	SoftwareMo...
ConstructorCallsOverridableMethod	3	20.5	1.50	SoftwareMo...
SystemPrintln	2	13.7	1.00	SoftwareMo...
VariableNamingConventions	21	143.8	10.50	SoftwareMo...
Test.java	62	76.5	6.20	SoftwareMo...
ConstructorCallsOverridableMethod	1	1.2	0.10	SoftwareMo...
SystemPrintln	5	6.2	0.50	SoftwareMo...
VariableNamingConventions	48	59.3	4.80	SoftwareMo...
MethodNamingConventions	8	9.9	0.80	SoftwareMo...
Test.java	62	76.5	6.20	SoftwareMo...
ConstructorCallsOverridableMethod	1	1.2	0.10	SoftwareMo...
SystemPrintln	5	6.2	0.50	SoftwareMo...
VariableNamingConventions	48	59.3	4.80	SoftwareMo...
MethodNamingConventions	8	9.9	0.80	SoftwareMo...
TC.java	12	226.4	6.00	SoftwareMo...
SystemPrintln	4	75.5	2.00	SoftwareMo...
VariableNamingConventions	6	113.2	3.00	SoftwareMo...
MethodNamingConventions	2	37.7	1.00	SoftwareMo...
RV.java	37	698.1	3.70	SoftwareMo...
VariableNamingConventions	26	490.6	2.60	SoftwareMo...
MethodNamingConventions	11	207.5	1.10	SoftwareMo...
Output_Manager.java	15	625.0	3.75	SoftwareMo...
VariableNamingConventions	8	333.3	2.00	SoftwareMo...
MethodNamingConventions	6	250.0	1.50	SoftwareMo...
SystemPrintln	1	41.7	0.25	SoftwareMo...
Input_Manager.java	99	811.5	5.50	SoftwareMo...
SystemPrintln	2	16.4	0.11	SoftwareMo...
VariableNamingConventions	65	532.8	3.61	SoftwareMo...
MethodNamingConventions	32	262.3	1.78	SoftwareMo...
Category.java	18	600.0	3.00	SoftwareMo...
VariableNamingConventions	14	466.7	2.33	SoftwareMo...
MethodNamingConventions	4	133.3	0.67	SoftwareMo...
Calculate.java	28	202.9	14.00	SoftwareMo...
SystemPrintln	4	29.0	2.00	SoftwareMo...
VariableNamingConventions	22	159.4	11.00	SoftwareMo...
MethodNamingConventions	2	14.5	1.00	SoftwareMo...

동일한 에러들이 많이 발견되었습니다.

```

124
▶ 125 addCategory(1,1);
▶ 126 addCategory(2,1);
▶ 127 addCategory(3,1);
128
    
```

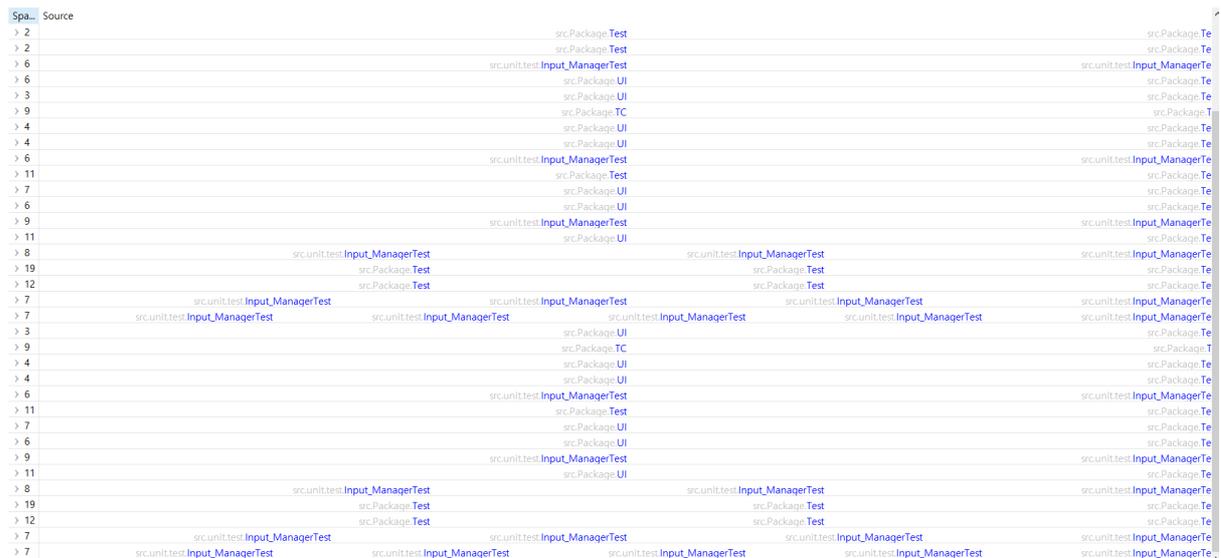
ConstructorCallsOverridableMethod : 오버라이드 가능한 메소드를 호출하게 되면 불완전한 객체로 메소드가 호출될 위험이 있어, 디버그하기 어려워진다.

```

27
▶ 28 private JTextField CategorytextField;
29 private JTextField RVtextField;
30
    
```

VariableNamingConventions : 변수는 소문자로 시작하여야합니다.

<CPD Test>



```

246 =====
247 Found a 7 line (29 tokens) duplication in the following files:
248 Starting at line 61 of C:\Users\sunggeun\workspace\T7_source\SoftwareModeling\src\unit\test\Input_ManagerTest.java
249 Starting at line 86 of C:\Users\sunggeun\workspace\T7_source\SoftwareModeling\src\unit\test\Input_ManagerTest.java
250 Starting at line 139 of C:\Users\sunggeun\workspace\T7_source\SoftwareModeling\src\unit\test\Input_ManagerTest.java
251 Starting at line 304 of C:\Users\sunggeun\workspace\T7_source\SoftwareModeling\src\unit\test\Input_ManagerTest.java
252 Starting at line 323 of C:\Users\sunggeun\workspace\T7_source\SoftwareModeling\src\unit\test\Input_ManagerTest.java
253
254 public void testAdd_RV() {
255     Input_Manager im = new Input_Manager();
256
257     im.Add_Category();
258     im.Add_RV(0);
259     im.Add_RV(0);
260     im.Add_RV(0);
261 =====
    
```

중복된 코드들이 위와 같이 나왔습니다.

2.2. Level 2 Eclipse Metrics Plugin

McCabe Cyclomatic Complexity: 프로그램 복잡도를 나타내는 software 단위. 높으면 높을수록 복잡하다.

Number of Parameters: 매개변수의 수

Nested Block Depth: 중첩된 {}의 개수

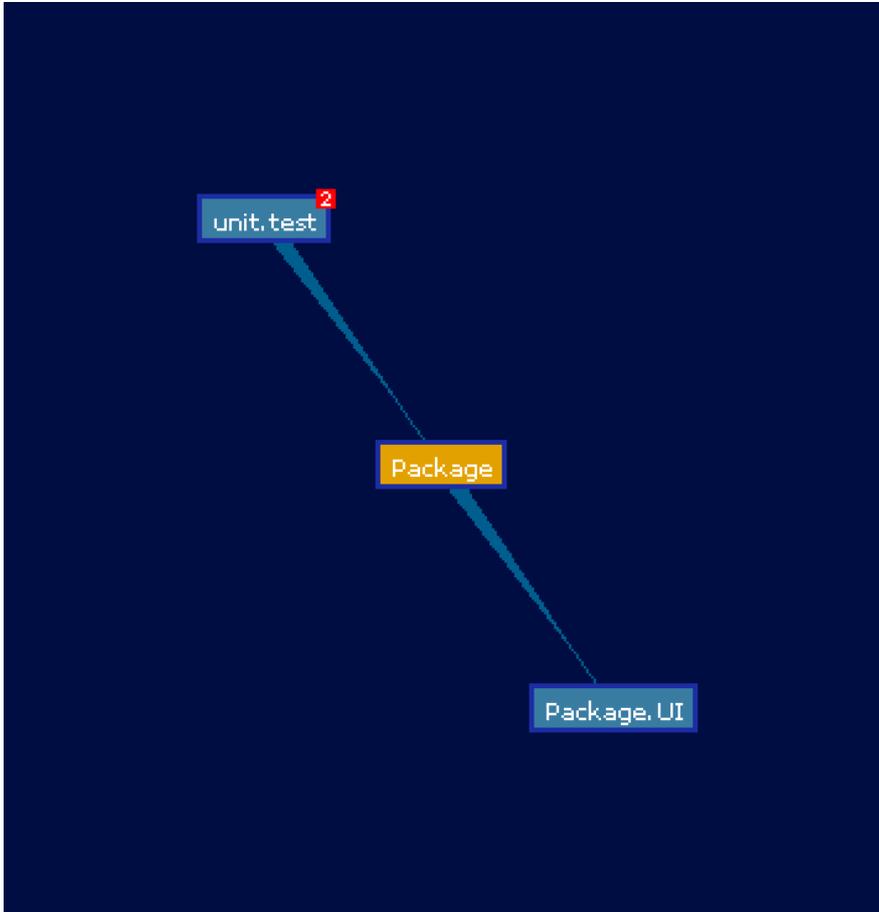
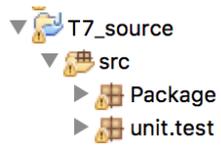
Total: 합계 / Mean: 평균 / Std. Dev.: 표준편차 / Maximum: 최대값

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▼ McCabe Cyclomatic Complexity (avg/max per method)		2.797	4.838	36	/T7_source/src/Package/Calculate.java	TC_Calculate
▶ Package		3.255	5.323	36	/T7_source/src/Package/Calculate.java	TC_Calculate
▶ unit.test		1	0	1	/T7_source/src/unit/test/Input_Manage...	testInput_Manager
▶ Number of Parameters (avg/max per method)		0.841	1.098	4	/T7_source/src/Package/Test.java	Print_RV
▼ Nested Block Depth (avg/max per method)		1.87	1.178	6	/T7_source/src/Package/Calculate.java	TC_Calculate
▶ Package		2.091	1.225	6	/T7_source/src/Package/Calculate.java	TC_Calculate
▶ unit.test		1	0	1	/T7_source/src/unit/test/Input_Manage...	testInput_Manager
▶ Afferent Coupling (avg/max per packageFragment)		0.5	0.5	1	/T7_source/src/Package	
▶ Efferent Coupling (avg/max per packageFragment)		1	0	1	/T7_source/src/Package	
▶ Instability (avg/max per packageFragment)		0.75	0.25	1	/T7_source/src/unit/test	
▶ Abstractness (avg/max per packageFragment)		0	0	0	/T7_source/src/Package	
▶ Normalized Distance (avg/max per packageFragment)		0.25	0.25	0.5	/T7_source/src/Package	
▶ Depth of Inheritance Tree (avg/max per type)		2	2	6	/T7_source/src/Package/Test.java	
▶ Weighted methods per Class (avg/max per type)	193	19.3	14.533	47	/T7_source/src/Package/Input_Manage...	
▶ Number of Children (avg/max per type)	0	0	0	0	/T7_source/src/Package/Calculate.java	
▶ Number of Overridden Methods (avg/max per type)	0	0	0	0	/T7_source/src/Package/Calculate.java	
▶ Lack of Cohesion of Methods (avg/max per type)		0.511	0.354	0.911	/T7_source/src/Package/Test.java	
▶ Number of Attributes (avg/max per type)	63	6.3	8.626	31	/T7_source/src/Package/Test.java	
▶ Number of Static Attributes (avg/max per type)	0	0	0	0	/T7_source/src/Package/Calculate.java	
▶ Number of Methods (avg/max per type)	68	6.8	5.51	18	/T7_source/src/Package/Input_Manage...	
▶ Number of Static Methods (avg/max per type)	1	0.1	0.3	1	/T7_source/src/Package/Test.java	
▶ Specialization Index (avg/max per type)		0	0	0	/T7_source/src/Package/Calculate.java	
▶ Number of Classes (avg/max per packageFragment)	10	5	4	9	/T7_source/src/Package	
▶ Number of Interfaces (avg/max per packageFragment)	0	0	0	0	/T7_source/src/Package	
▶ Number of Packages	2					
▶ Total Lines of Code	1453					
▶ Method Lines of Code (avg/max per method)	1162	16.841	34.431	208	/T7_source/src/Package/Test.java	Clear_Screen

<전체 프로젝트>

Metric	Total	Mean	Std. Dev.	Maximum	Resource causing Maximum	Method
▼ McCabe Cyclomatic Complexity (avg/max per method)		3.255	5.323	36	/T7_source/src/Package/Calculate.java	TC_Calculate
▼ Calculate.java		18.5	17.5	36	/T7_source/src/Package/Calculate.java	TC_Calculate
▼ Calculate		18.5	17.5	36	/T7_source/src/Package/Calculate.java	TC_Calculate
TC_Calculate	36					
Calculate	1					
▼ TC.java		10.5	9.5	20	/T7_source/src/Package/TC.java	If_Property_Check
▼ TC		10.5	9.5	20	/T7_source/src/Package/TC.java	If_Property_Check
If_Property_Check	20					
TC	1					
▶ Test.java		3.2	2.182	8	/T7_source/src/Package/Test.java	Print_Output_Screen
▶ Input_Manager.java		2.611	1.948	8	/T7_source/src/Package/Input_Manage...	Add_Constraint
▶ RV.java		2.1	1.044	4	/T7_source/src/Package/RV.java	setSingle_Error
▶ Output_Manager.java		1.75	1.299	4	/T7_source/src/Package/Output_Man...	Output_File
▶ Category.java		1.667	0.943	3	/T7_source/src/Package/Category.java	Search_RV
▶ UI.java		1.333	0.471	2	/T7_source/src/Package/UI.java	actionPerformed
▶ Number of Parameters (avg/max per method)		1.055	1.135	4	/T7_source/src/Package/Test.java	Print_RV
▼ Nested Block Depth (avg/max per method)		2.091	1.225	6	/T7_source/src/Package/Calculate.java	TC_Calculate
▼ Calculate.java		3.5	2.5	6	/T7_source/src/Package/Calculate.java	TC_Calculate
Calculate		3.5	2.5	6	/T7_source/src/Package/Calculate.java	TC_Calculate
TC_Calculate	6					
Calculate	1					
▼ TC.java		3.5	2.5	6	/T7_source/src/Package/TC.java	If_Property_Check
TC		3.5	2.5	6	/T7_source/src/Package/TC.java	If_Property_Check
If_Property_Check	6					
TC	1					
▶ Test.java		2.6	0.917	4	/T7_source/src/Package/Test.java	Clear_Screen
▶ Input_Manager.java		2	0.943	4	/T7_source/src/Package/Input_Manage...	Add_Constraint
▶ Output_Manager.java		1.75	1.299	4	/T7_source/src/Package/Output_Man...	Output_File
▶ Category.java		1.667	0.943	3	/T7_source/src/Package/Category.java	Search_RV
▶ RV.java		1.8	0.748	3	/T7_source/src/Package/RV.java	Delete_Property
▶ UI.java		1.333	0.471	2	/T7_source/src/Package/UI.java	actionPerformed
Afferent Coupling	1					
Efferent Coupling	1					
Instability	0.5					
Abstractness	0					
Normalized Distance	0.5					
▶ Depth of Inheritance Tree (avg/max per type)		2.111	2.079	6	/T7_source/src/Package/Test.java	
▶ Weighted methods per Class (avg/max per type)	179	19.889	15.206	47	/T7_source/src/Package/Input_Manage...	
▶ Number of Children (avg/max per type)	0	0	0	0	/T7_source/src/Package/Calculate.java	
▶ Number of Overridden Methods (avg/max per type)	0	0	0	0	/T7_source/src/Package/Calculate.java	
▶ Lack of Cohesion of Methods (avg/max per type)		0.567	0.327	0.911	/T7_source/src/Package/Test.java	
▶ Number of Attributes (avg/max per type)	63	7	8.819	31	/T7_source/src/Package/Test.java	
▶ Number of Static Attributes (avg/max per type)	0	0	0	0	/T7_source/src/Package/Calculate.java	

<package Package>



<Dependency Graph>

2.3. Level 3 FindBugs

