<Software Verification>

슈퍼맨과 함께하는 영어 끝말잇기 Verification

<Static Analysis Report #1>

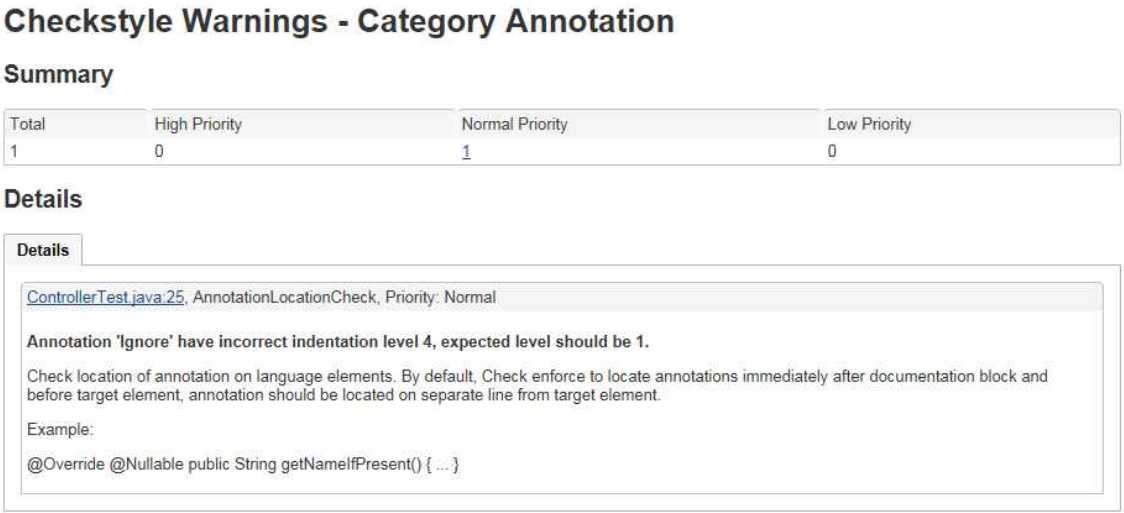
Tema 4.
200911393 박현규
201010768 최정한
201111339 김민우
201211389 함진아

1. Checkstyle

- CheckStyle은 코딩 표준에 맞게 소스코드를 작성하도록 도와주는 도구이다.
- 개발자들이 잘 지키지는 않지만 중요한 것들을 지적해준다 (ex : private 설정)
- 협업시에 코딩스타일을 맞출 수 있고, 잠재적인 결함 발견도 가능하다. 팀의 코딩규칙을 문서로 정리해서 지키기로 하면, 규칙이 많아질 수록 어기기 쉽다.

- Checkstyle에서 기본적으로 제공하는 ruleset을 사용할 경우
- 14개의 카테고리 및 149개의 rule
  → 1667개의 warning 발생

1) Annotation



ControllerTest.java:25 - 들여쓰기에 대한 지적

```
24   @Test
25     @Ignore("Not yet")
26   public void testChildMode() throws FileNotFoundException, URISyntaxException {
27     Controller cnt = new Controller();
28     cnt.childMode();
29     assertEquals("name",cnt.getTemp());
30     assertEquals(false,cnt.checkChildAccount("name"));
31     assertEquals(true,cnt.checkChildAccount("以□□빌□얇"));
32   }
```

2) Metrics

Checkstyle Warnings - Category Metrics

Summary

Total	High Priority	Normal Priority	Low Priority
21	0	21	0

Details

Folders | Files | **Types** | Warnings | Details

Type	Total	Distribution
ClassDataAbstractionCouplingCheck	11	
ClassFanOutComplexityCheck	2	
JavaNCSSCheck	8	
Total	21	

* ClassDataAbstractionCouplingCheck
  - 클래스에서 사용된 레퍼런스 타입의 수를 계산한다.
  - 최대 7개까지가 적정선.

ex) Controller.java:15 – 8개의 참조 발견

```
015 public class Controller
016 {
017 //Variables
018   private Parent parent;

        BufferedReader in = new BufferedReader(new FileReader(getClass().getClassLoader().getResource
        rent.txt").getPath().replace("%5c", "\\")));
        temp=in.readLine();
        if(checkParentAccount(temp)==false)
        {
          new GUI_Account(this);
        }
        else
        {
          new GUI_Check_Password(this,this.superman);
        }
```

* ClassFanOutComplexityCheck
  - 참조하여 사용하는 class의 개수
  - 최대 20개까지가 적정선.

File	Source Folder	Line	Priority	Type	Category
Controller.java:15	C:/Users/함진아/workspace/t4_project/src/Classes	15	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Account.java:24	C:/Users/함진아/workspace/t4_project/src/GUI	24	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Account.java:34	C:/Users/함진아/workspace/t4_project/src/GUI	34	Normal	JavaNCSSCheck	Metrics
GUI_Check_Goal.java:22	C:/Users/함진아/workspace/t4_project/src/GUI	22	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Check_Goal.java:32	C:/Users/함진아/workspace/t4_project/src/GUI	32	Normal	JavaNCSSCheck	Metrics
GUI_Check_Password.java:25	C:/Users/함진아/workspace/t4_project/src/GUI	25	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Child_Mode.java:25	C:/Users/함진아/workspace/t4_project/src/GUI	25	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Child_Mode.java:37	C:/Users/함진아/workspace/t4_project/src/GUI	37	Normal	JavaNCSSCheck	Metrics
GUI_Dictionary.java:33	C:/Users/함진아/workspace/t4_project/src/GUI	33	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Dictionary.java:48	C:/Users/함진아/workspace/t4_project/src/GUI	48	Normal	JavaNCSSCheck	Metrics
GUI_Goal_Setting.java:26	C:/Users/함진아/workspace/t4_project/src/GUI	26	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Goal_Setting.java:37	C:/Users/함진아/workspace/t4_project/src/GUI	37	Normal	JavaNCSSCheck	Metrics
GUI_Parent_Mode.java:22	C:/Users/함진아/workspace/t4_project/src/GUI	22	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Parent_Mode.java:34	C:/Users/함진아/workspace/t4_project/src/GUI	34	Normal	JavaNCSSCheck	Metrics
GUI_Send_Message.java:25	C:/Users/함진아/workspace/t4_project/src/GUI	25	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Wordtrain_Game.java:37	C:/Users/함진아/workspace/t4_project/src/GUI	37	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Wordtrain_Game.java:37	C:/Users/함진아/workspace/t4_project/src/GUI	37	Normal	ClassFanOutComplexityCheck	Metrics
GUI_Wordtrain_Game.java:55	C:/Users/함진아/workspace/t4_project/src/GUI	55	Normal	JavaNCSSCheck	Metrics
GUI_Wordtrain_Practice.java:31	C:/Users/함진아/workspace/t4_project/src/GUI	31	Normal	ClassFanOutComplexityCheck	Metrics
GUI_Wordtrain_Practice.java:31	C:/Users/함진아/workspace/t4_project/src/GUI	31	Normal	ClassDataAbstractionCouplingCheck	Metrics
GUI_Wordtrain_Practice.java:49	C:/Users/함진아/workspace/t4_project/src/GUI	49	Normal	JavaNCSSCheck	Metrics

3) Naming

Checkstyle Warnings - Category Naming

Summary

Total	High Priority	Normal Priority	Low Priority
43	0	43	0

Details

Folders | Files | **Types** | Warnings | Details

Type	Total	Distribution
LocalVariableNameCheck	1	
MemberNameCheck	5	
PackageNameCheck	25	
TypeNameCheck	12	
Total	43	

* LocalVariableNameCheck
  - Local 변수의 이름이 제시하는 format에 맞는가 ?
  - Naming의 전반적인 내용이 언더바 사용에 대한 지적

WordtrainTest.java:38, LocalVariableNameCheck, Priority: Normal

Name 'voca_next' must match pattern '^[a-z][a-zA-Z0-9]*$'.

No description available. Please upgrade to latest checkstyle version.

```
public void testPrintWordList() throws IOException, URISyntaxException {
  Wordtrain word = new Wordtrain();
  Voca voca[] = new Voca[3];
  Voca voca_next[] = new Voca[3];
```

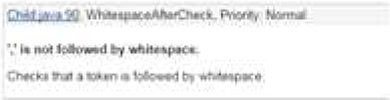
File	Source Folder	Line	Priority	Type	Category
Child.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
ChildTest.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Controller.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
ControllerTest.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Dictionary.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
DictionaryTest.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Main.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Parent.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
ParentTest.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Superman.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Voca.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
Wordtrain.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
WordtrainTest.java:1	C:/Users/함진아/workspace/t4_project/src/Classes	1	Normal	PackageNameCheck	Naming
WordtrainTest.java:38	C:/Users/함진아/workspace/t4_project/src/Classes	38	Normal	LocalVariableNameCheck	Naming
GUI_Account.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Account.java:24	C:/Users/함진아/workspace/t4_project/src/GUI	24	Normal	TypeNameCheck	Naming
GUI_Basic.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Basic.java:11	C:/Users/함진아/workspace/t4_project/src/GUI	11	Normal	TypeNameCheck	Naming
GUI_Basic.java:16	C:/Users/함진아/workspace/t4_project/src/GUI	16	Normal	MemberNameCheck	Naming
GUI_Basic.java:17	C:/Users/함진아/workspace/t4_project/src/GUI	17	Normal	MemberNameCheck	Naming
GUI_Basic.java:18	C:/Users/함진아/workspace/t4_project/src/GUI	18	Normal	MemberNameCheck	Naming
GUI_Basic.java:19	C:/Users/함진아/workspace/t4_project/src/GUI	19	Normal	MemberNameCheck	Naming
GUI_Basic.java:20	C:/Users/함진아/workspace/t4_project/src/GUI	20	Normal	MemberNameCheck	Naming
GUI_Check_Goal.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Check_Goal.java:22	C:/Users/함진아/workspace/t4_project/src/GUI	22	Normal	TypeNameCheck	Naming
GUI_Check_Password.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Check_Password.java:25	C:/Users/함진아/workspace/t4_project/src/GUI	25	Normal	TypeNameCheck	Naming
GUI_Child_Mode.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Child_Mode.java:25	C:/Users/함진아/workspace/t4_project/src/GUI	25	Normal	TypeNameCheck	Naming
GUI_Dictionary.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Dictionary.java:33	C:/Users/함진아/workspace/t4_project/src/GUI	33	Normal	TypeNameCheck	Naming
GUI_Goal_Setting.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Goal_Setting.java:26	C:/Users/함진아/workspace/t4_project/src/GUI	26	Normal	TypeNameCheck	Naming
GUI_Main_View.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Main_View.java:17	C:/Users/함진아/workspace/t4_project/src/GUI	17	Normal	TypeNameCheck	Naming
GUI_Parent_Mode.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Parent_Mode.java:22	C:/Users/함진아/workspace/t4_project/src/GUI	22	Normal	TypeNameCheck	Naming
GUI_Send_Message.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Send_Message.java:25	C:/Users/함진아/workspace/t4_project/src/GUI	25	Normal	TypeNameCheck	Naming
GUI_Wordtrain_Game.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Wordtrain_Game.java:37	C:/Users/함진아/workspace/t4_project/src/GUI	37	Normal	TypeNameCheck	Naming
GUI_Wordtrain_Practice.java:1	C:/Users/함진아/workspace/t4_project/src/GUI	1	Normal	PackageNameCheck	Naming
GUI_Wordtrain_Practice.java:31	C:/Users/함진아/workspace/t4_project/src/GUI	31	Normal	TypeNameCheck	Naming

4) Whitespace

Folders | Files | **Types** | Warnings | Details

Type	Total	Distribution
EmptyLineSeparatorCheck	119	
FileTabCharacterCheck	25	
WhitespaceAfterCheck	84	
WhitespaceAroundCheck	1374	
Total	1602	

- EmptyLineSeparatorCheck
  → 초기화 시에 공백이 필요, 혹은 잘못된 공백 사용
- FileTabCharacterCheck
  → 소스코드 내 tab 사용
- WhitespaceAfterCheck
  → 토큰 뒤에 공백이 필요
- WhitespaceAroundCheck
  → 토큰 앞뒤에 공백이 필요



File	Source Folder	Line	Priority	Type	Category
Child.java:15	C:/Users/함진아/workspace/t4_project/src/Classes	15	Normal	FileTabCharacterCheck	Whitespace
Child.java:16	C:/Users/함진아/workspace/t4_project/src/Classes	16	Normal	EmptyLineSeparatorCheck	Whitespace
Child.java:17	C:/Users/함진아/workspace/t4_project/src/Classes	17	Normal	EmptyLineSeparatorCheck	Whitespace
Child.java:25	C:/Users/함진아/workspace/t4_project/src/Classes	25	Normal	WhitespaceAroundCheck	Whitespace
Child.java:25	C:/Users/함진아/workspace/t4_project/src/Classes	25	Normal	WhitespaceAroundCheck	Whitespace
Child.java:85	C:/Users/함진아/workspace/t4_project/src/Classes	85	Normal	WhitespaceAroundCheck	Whitespace
Child.java:85	C:/Users/함진아/workspace/t4_project/src/Classes	85	Normal	WhitespaceAroundCheck	Whitespace
Child.java:85	C:/Users/함진아/workspace/t4_project/src/Classes	85	Normal	WhitespaceAroundCheck	Whitespace
Child.java:85	C:/Users/함진아/workspace/t4_project/src/Classes	85	Normal	WhitespaceAroundCheck	Whitespace
Child.java:85	C:/Users/함진아/workspace/t4_project/src/Classes	85	Normal	WhitespaceAroundCheck	Whitespace
Child.java:85	C:/Users/함진아/workspace/t4_project/src/Classes	85	Normal	WhitespaceAroundCheck	Whitespace
Child.java:90	C:/Users/함진아/workspace/t4_project/src/Classes	90	Normal	WhitespaceAfterCheck	Whitespace
Child.java:95	C:/Users/함진아/workspace/t4_project/src/Classes	95	Normal	WhitespaceAfterCheck	Whitespace
Child.java:100	C:/Users/함진아/workspace/t4_project/src/Classes	100	Normal	WhitespaceAfterCheck	Whitespace
ChildTest.java:9	C:/Users/함진아/workspace/t4_project/src/Classes	9	Normal	FileTabCharacterCheck	Whitespace
Controller.java:18	C:/Users/함진아/workspace/t4_project/src/Classes	18	Normal	FileTabCharacterCheck	Whitespace
Controller.java:19	C:/Users/함진아/workspace/t4_project/src/Classes	19	Normal	EmptyLineSeparatorCheck	Whitespace
Controller.java:20	C:/Users/함진아/workspace/t4_project/src/Classes	20	Normal	EmptyLineSeparatorCheck	Whitespace
Controller.java:21	C:/Users/함진아/workspace/t4_project/src/Classes	21	Normal	EmptyLineSeparatorCheck	Whitespace
Controller.java:22	C:/Users/함진아/workspace/t4_project/src/Classes	22	Normal	EmptyLineSeparatorCheck	Whitespace
Controller.java:52	C:/Users/함진아/workspace/t4_project/src/Classes	52	Normal	WhitespaceAroundCheck	Whitespace
Controller.java:52	C:/Users/함진아/workspace/t4_project/src/Classes	52	Normal	WhitespaceAroundCheck	Whitespace
Controller.java:67	C:/Users/함진아/workspace/t4_project/src/Classes	67	Normal	WhitespaceAroundCheck	Whitespace
Controller.java:67	C:/Users/함진아/workspace/t4_project/src/Classes	67	Normal	WhitespaceAroundCheck	Whitespace
Controller.java:68	C:/Users/함진아/workspace/t4_project/src/Classes	68	Normal	WhitespaceAroundCheck	Whitespace
Controller.java:68	C:/Users/함진아/workspace/t4_project/src/Classes	68	Normal	WhitespaceAroundCheck	Whitespace
Controller.java:68	C:/Users/함진아/workspace/t4_project/src/Classes	68	Normal	WhitespaceAroundCheck	Whitespace

2. PMD

- Program May Dependable의 약자
- 정해진 규칙에 따라 소스코드를 검사해서 위반사항을 찾아낼 수 있다.
- Eclipse –plugin으로 사용할 수 있으며, 별도로 커맨드라인에서 사용 가능하다.
- 분석 대상은 java, javaScript, XML 등이 있다.
- PMD에서 제공하는 카테고리로부터, 분석 대상 프로그램 언어인 Java에 맞는 카테고리 선정

● 카테고리의 큰 틀 16가지

Basic (rulesets/basic.xml)	**Code size (rulesets/codesize.xml)**
Naming (rulesets/naming.xml)	Javabeans (rulesets/javabeans.xml)
Unused code (rulesets/unusedcode.xml)	Finalizers
Design (rulesets/design.xml)	Clone (rulesets/clone.xml)
Import statements (rulesets/imports.xml)	Coupling (rulesets/coupling.xml)
JUnit tests (rulesets/junit.xml)	Strict exceptions (rulesets/strictexception.xml)
Strings (rulesets/string.xml)	Controversial (rulesets/controversial.xml)
Braces (rulesets/braces.xml)	Logging (rulesets/logging-java.xml)

PMD Result

Warnings Trend

All Warnings	New Warnings	Fixed Warnings
14	1	149

Summary

Total	High Priority	Normal Priority	Low Priority
14	0	14	0

Details

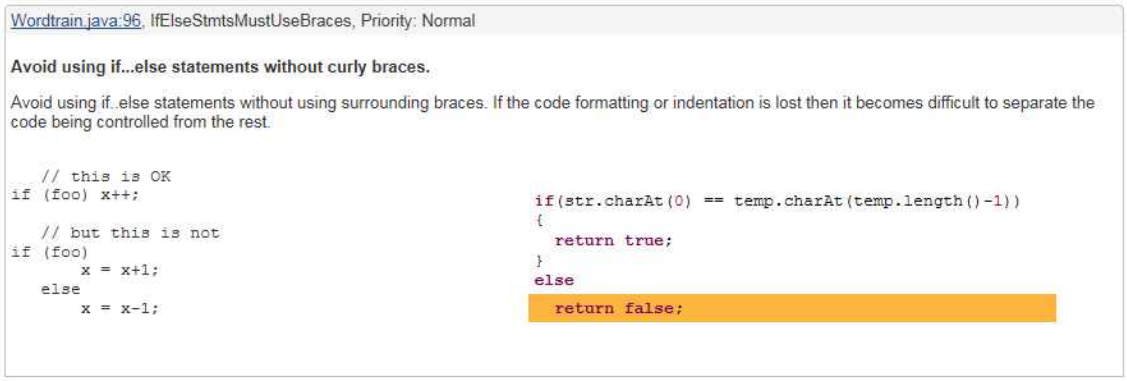
Packages | Files | **Categories** | Types | Warnings | Details | New | Fixed

Category	Total	Distribution
Braces	1	
Code Size	6	
String and StringBuffer	7	
Total	14	

1) Braces

- for, if, while, else 문장이 괄호를 사용하는지 여부 검사.

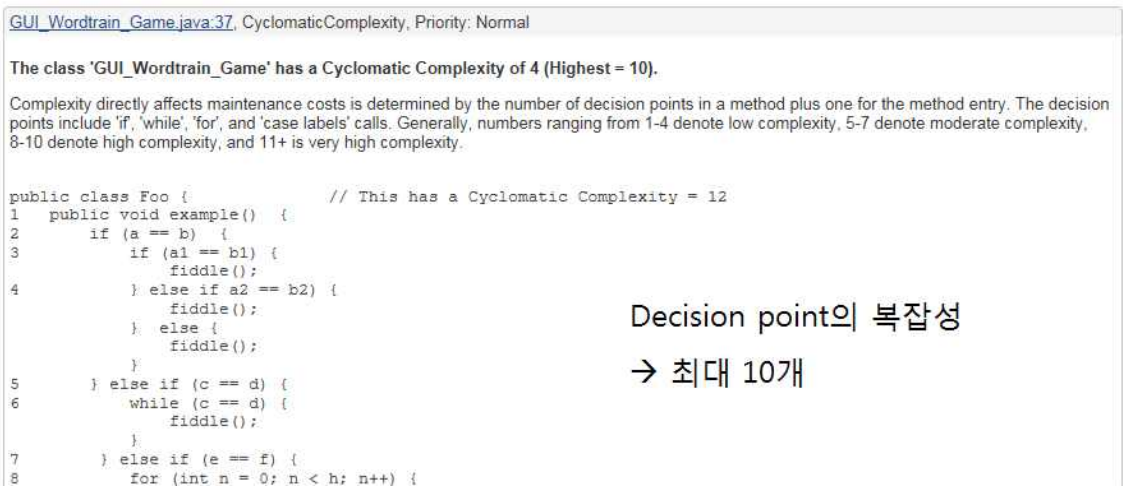
* Wordtrain.java:96 – if/else 문에서 괄호 사용 안함



2) Code Size

- 과도하게 긴 메소드, 너무 많은 메소드를 가진 클래스, 리팩토링에 대한 유사한 후보들을 위한 테스트.

ex)



GUI_Child_Mode.java:37, NcssConstructorCount, Priority: Normal

The constructor with 2 parameters has an NCSS line count of 115.

This rule uses the NCSS (Non-Commenting Source Statements) algorithm to determine the number of lines of code for a given constructor. NCSS ignores comments, and counts actual statements. Using this algorithm, lines of code that are split are counted as one.

```
public class Foo extends Bar {
 public Foo() {
     super();




 //this constructor only has 1 NCSS lines
      super.foo();
 }
}
```

File	Line	Priority	Type	Category
GUI_Child_Mode.java:37	37	Normal	NcssConstructorCount	Code Size
GUI_Wordtrain_Game.java:37	37	Normal	CyclomaticComplexity	Code Size
GUI_Wordtrain_Game.java:188	188	Normal	CyclomaticComplexity	Code Size
GUI_Wordtrain_Practice.java:31	31	Normal	CyclomaticComplexity	Code Size
GUI_Wordtrain_Practice.java:49	49	Normal	NcssConstructorCount	Code Size
GUI_Wordtrain_Practice.java:207	207	Normal	CyclomaticComplexity	Code Size

3) String

스트링 관련 작업을 할 때 발생하는 일반적인 문제들 규명. 스트링 리터럴 중복, String 구조체 호출, String 객체에 toString() 호출하기 등.

ex)

Controller.java:66, AvoidDuplicateLiterals, Priority: Normal

The String literal "%5c" appears 8 times in this file; the first occurrence is on line 66.

Code containing duplicate String literals can usually be improved by declaring the String as a constant field.

```
private void bar() {
     buz("Howdy");
     buz("Howdy");
     buz("Howdy");
     buz("Howdy");
 }
 private void buz(String x) {}
```

File	Package	Line	Priority	Type	Category
Controller.java:66	Classes	66	Normal	AvoidDuplicateLiterals	String and StringBuffer
ControllerTest.java:18	Classes	18	Normal	AvoidDuplicateLiterals	String and StringBuffer
Parent.java:36	Classes	36	Normal	AvoidDuplicateLiterals	String and StringBuffer
Superman.java:22	Classes	22	Normal	AvoidDuplicateLiterals	String and StringBuffer
WordtrainTest.java:16	Classes	16	Normal	AvoidDuplicateLiterals	String and StringBuffer
WordtrainTest.java:18	Classes	18	Normal	AvoidDuplicateLiterals	String and StringBuffer
GUI_Child_Mode.java:61	GUI	61	Normal	AvoidDuplicateLiterals	String and StringBuffer

3. FindBugs

자바프로그램 분석도구로, 메릴랜드 대학에서 개발했다.
data flow, control flow 분석등을 이용한다.
버그 패턴을 자동으로 찾아서 알려준다 (ex : race condition, null Pointer exception 등)
source code(*.java 파일) 보다는 byte code(*.class 파일)를 기반으로 분석을 진행한다.●

FindBugs Result

Warnings Trend

All Warnings	New this build	Fixed Warnings
29	0	0

Summary

Total	High Priority	Normal Priority	Low Priority
29	16	13	0

Details

Packages | Files | Categories | **Types** | Warnings | Details | High | Normal

Type	Total	Distribution
DM_DEFAULT_ENCODING	16	
EI_EXPOSE_REP	5	
NP_DEREFERENCE_OF_READLINE_VALUE	3	
RV_DONT_JUST_NULL_CHECK_READLINE	1	
UC_USELESS_OBJECT	2	
UWF_UNWRITTEN_FIELD	2	
Total	29	

ex)

Child.java:23, DM_DEFAULT_ENCODING, Priority: High

Dm: Found reliance on default encoding in new Classes.Child(Superman): new java.util.Scanner(File)

Found a call to a method which will perform a byte to String (or String to byte) conversion, and will assume that the default platform encoding is suitable. This will cause the application behaviour to vary between platforms. Use an alternative API and specify a charset name or Charset object explicitly.

Controller.java:34, UWF_UNWRITTEN_FIELD, Priority: Normal

UwF: Unwritten field: Classes.Controller.child

This field is never written. All reads of it will return the default value. Check for errors (should it have been initialized?), or remove it if it is useless.

Controller.java:41, UWF_UNWRITTEN_FIELD, Priority: Normal

UwF: Unwritten field: Classes.Controller.parent

This field is never written. All reads of it will return the default value. Check for errors (should it have been initialized?), or remove it if it is useless.

Controller.java:68, NP_DEREFERENCE_OF_READLINE_VALUE, Priority: Normal

NP: Dereference of the result of readLine() without nullcheck in Classes.Controller.parentMode()

The result of invoking readLine() is dereferenced without checking to see if the result is null. If there are no more lines of text to read, readLine() will return null and dereferencing that will generate a null pointer exception.

Wordtrain.java:43, EI_EXPOSE_REP, Priority: Normal

EI: Classes.Wordtrain.getWordList() may expose internal representation by returning Wordtrain.wordlist

Returning a reference to a mutable object value stored in one of the object's fields exposes the internal representation of the object. If instances are accessed by untrusted code, and unchecked changes to the mutable object would compromise security or other important properties, you will need to do something different. Returning a new copy of the object is better approach in many situations.

Wordtrain.java:109, RV_DONT_JUST_NULL_CHECK_READLINE, Priority: Normal

RV: Classes.Wordtrain.searchWordList(char) discards result of readLine after checking if it is nonnull

The value returned by readLine is discarded after checking to see if the return value is non-null. In almost all situations, if the result is non-null, you will want to use that non-null value. Calling readLine again will give you a different line.

Wordtrain.java:133, NP_DEREFERENCE_OF_READLINE_VALUE, Priority: Normal

NP: Dereference of the result of readLine() without nullcheck in Classes.Wordtrain.searchWordList(char)

The result of invoking readLine() is dereferenced without checking to see if the result is null. If there are no more lines of text to read, readLine() will return null and dereferencing that will generate a null pointer exception.

WordtrainTest.java:37, UC_USELESS_OBJECT, Priority: Normal

Useless object created

Our analysis shows that this object is useless. It's created and modified, but its value never go outside of the method or produce any side-effect. Either there is a mistake and object was intended to be used or it can be removed.

This analysis rarely produces false-positives. Common false-positive cases include:

- This object used to implicitly throw some obscure exception.

- This object used as a stub to generalize the code.

- This object used to hold strong references to weak/soft-referenced objects.

File	Package	Line	Priority	Rank	Type	Category
Child.java:23	Classes	23	High	19	DM_DEFAULT_ENCODING	I18N
Controller.java:34	Classes	34	Normal	12	UWF_UNWRITTEN_FIELD	CORRECTNESS
Controller.java:41	Classes	41	Normal	12	UWF_UNWRITTEN_FIELD	CORRECTNESS
Controller.java:66	Classes	66	High	19	DM_DEFAULT_ENCODING	I18N
Controller.java:68	Classes	68	Normal	15	NP_DEREFERENCE_OF_READLINE_VALUE	STYLE
Controller.java:87	Classes	87	High	19	DM_DEFAULT_ENCODING	I18N
Controller.java:89	Classes	89	Normal	15	NP_DEREFERENCE_OF_READLINE_VALUE	STYLE
Controller.java:109	Classes	109	High	19	DM_DEFAULT_ENCODING	I18N
Dictionary.java:39	Classes	39	High	19	DM_DEFAULT_ENCODING	I18N
Parent.java:36	Classes	36	High	19	DM_DEFAULT_ENCODING	I18N
ParentTest.java:26	Classes	26	High	19	DM_DEFAULT_ENCODING	I18N
ParentTest.java:43	Classes	43	High	19	DM_DEFAULT_ENCODING	I18N
Superman.java:23	Classes	23	High	19	DM_DEFAULT_ENCODING	I18N
Superman.java:52	Classes	52	High	19	DM_DEFAULT_ENCODING	I18N
Superman.java:67	Classes	67	High	19	DM_DEFAULT_ENCODING	I18N
Superman.java:83	Classes	83	High	19	DM_DEFAULT_ENCODING	I18N
Superman.java:99	Classes	99	High	19	DM_DEFAULT_ENCODING	I18N
Voca.java:31	Classes	31	High	19	DM_DEFAULT_ENCODING	I18N
Wordtrain.java:43	Classes	43	Normal	18	EI_EXPOSE_REP	MALICIOUS_CODE
Wordtrain.java:49	Classes	49	Normal	18	EI_EXPOSE_REP	MALICIOUS_CODE
Wordtrain.java:60	Classes	60	Normal	18	EI_EXPOSE_REP	MALICIOUS_CODE
Wordtrain.java:72	Classes	72	Normal	18	EI_EXPOSE_REP	MALICIOUS_CODE
Wordtrain.java:79	Classes	79	Normal	18	EI_EXPOSE_REP	MALICIOUS_CODE
Wordtrain.java:108	Classes	108	High	19	DM_DEFAULT_ENCODING	I18N
Wordtrain.java:109	Classes	109	Normal	17	RV_DONT_JUST_NULL_CHECK_READLINE	STYLE
Wordtrain.java:133	Classes	133	Normal	15	NP_DEREFERENCE_OF_READLINE_VALUE	STYLE
WordtrainTest.java:37	Classes	37	Normal	20	UC_USELESS_OBJECT	STYLE
WordtrainTest.java:38	Classes	38	Normal	20	UC_USELESS_OBJECT	STYLE
GUI_Wordtrain_Game.java:308	GUI	308	High	19	DM_DEFAULT_ENCODING	I18N

4. JDepend

패키지 의존성과 관련된 설계 품질을 관리/분석 할 수 있다.
클래스 파일들을 읽어서 분석을 한다.
각 패키지별로 의존성 측정이 가능하며, 수치화, 그래프로 표현이 가능하다.
개발과정 중에도 아키텍쳐 품질을 평가할 수 있다.
XML 형식으로도 저장 가능하다.

Package	TC	CC	AC	Ca	Ce	A	I	D	V
Classes	14	14	0	1	2	0.0%	67.0%	33.0%	1
GUI	12	12	0	1	1	0.0%	50.0%	50.0%	1

※ 결과가 가지는 의미

CC	Concrete Class
AC	Abstract Class 추상클래스나 인터페이스의 개수. 확정성의 척도
Ca	Afferent Couplings 현재 패키지에 종속성을 갖는 패키지 개수.
Ce	Efferent Couplings 현재 패키지가 종속하고 있는 패키지 개수.
A	추상화 정도. 0은 완전 구체적 패키지. 1은 추상적 패키지.
D	Main Sequence로 부터의 거리. Main Sequence란 추상적이면서 안정적이거나, 완전 구체적이면서 불완정한 패키지. 0일수록 가깝고 1일수록 멀다.

Term	Description
Number of Classes	The number of concrete and abstract classes (and interfaces) in the package is an indicator of the extensibility of the package.
Afferent Couplings	The number of other packages that depend upon classes within the package is an indicator of the package's responsibility.
Efferent Couplings	The number of other packages that the classes in the package depend upon is an indicator of the package's independence.
Abstractness	The ratio of the number of abstract classes (and interfaces) in the analyzed package to the total number of classes in the analyzed package. The range for this metric is 0 to 1, with A=0 indicating a completely concrete package and A=1 indicating a completely abstract package.
Instability	The ratio of efferent coupling (Ce) to total coupling (Ce / (Ce + Ca)). This metric is an indicator of the package's resilience to change. The range for this metric is 0 to 1, with I=0 indicating a completely stable package and I=1 indicating a completely instable package.
Distance	The perpendicular distance of a package from the idealized line A + I = 1. This metric is an indicator of the package's balance between abstractness and stability. A package squarely on the main sequence is optimally balanced with respect to its abstractness and stability. Ideal packages are either completely abstract and stable (x=0, y=1) or completely concrete and instable (x=1, y=0). The range for this metric is 0 to 1, with D=0 indicating a package that is coincident with the main sequence and D=1 indicating a package that is as far from the main sequence as possible.
Cycles	Packages participating in a package dependency cycle are in a deadly embrace with respect to reusability and their release cycle. Package dependency cycles can be easily identified by reviewing the textual reports of dependency cycles. Once these dependency cycles have been identified with JDepend, they can be broken by employing various object-oriented techniques.

Classes

Afferent Couplings	Efferent Couplings	Abstractness	Instability	Distance
1	2	0.0%	67.0%	33.0%

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	Classes.Child Classes.ChildTest Classes.Controller Classes.ControllerTest Classes.Dictionary Classes.DictionaryTest Classes.Main Classes.Main$1 Classes.Parent Classes.ParentTest Classes.Superman Classes.Voca Classes.Wordtrain Classes.WordtrainTest	GUI	GUI org.junit

GUI

Afferent Couplings	Efferent Couplings	Abstractness	Instability	Distance
1	1	0.0%	50.0%	50.0%

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	GUI.GUI_Account GUI.GUI_Basic GUI.GUI_Check_Goal GUI.GUI_Check_Password GUI.GUI_Child_Mode GUI.GUI_Dictionary GUI.GUI_Goal_Setting GUI.GUI_Main_View GUI.GUI_Parent_Mode GUI.GUI_Send_Message GUI.GUI_Wordtrain_Game GUI.GUI_Wordtrain_Practice	Classes	Classes

Cycles

[summary] [packages] [cycles] [explanations]

Package	Package Dependencies
Classes	GUI Classes
GUI	Classes GUI