

Software Verification

Introduction to Software Testing & Static Analysis

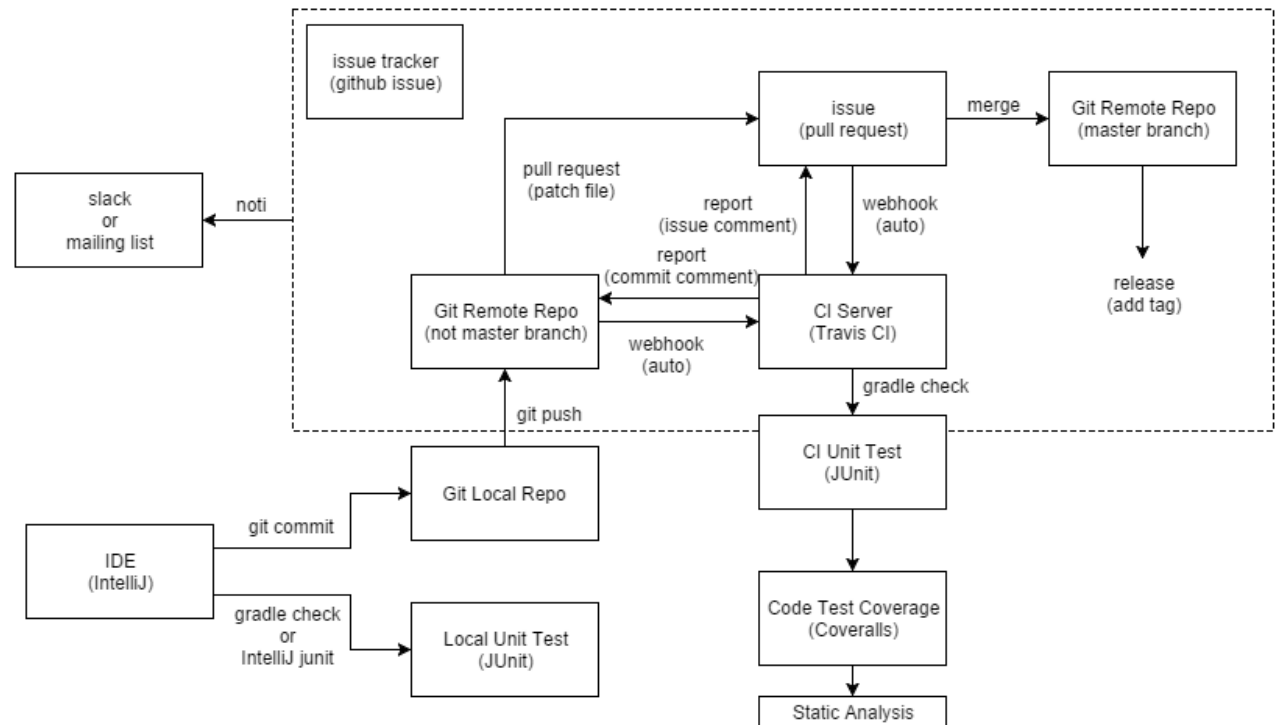
2조

이상혁

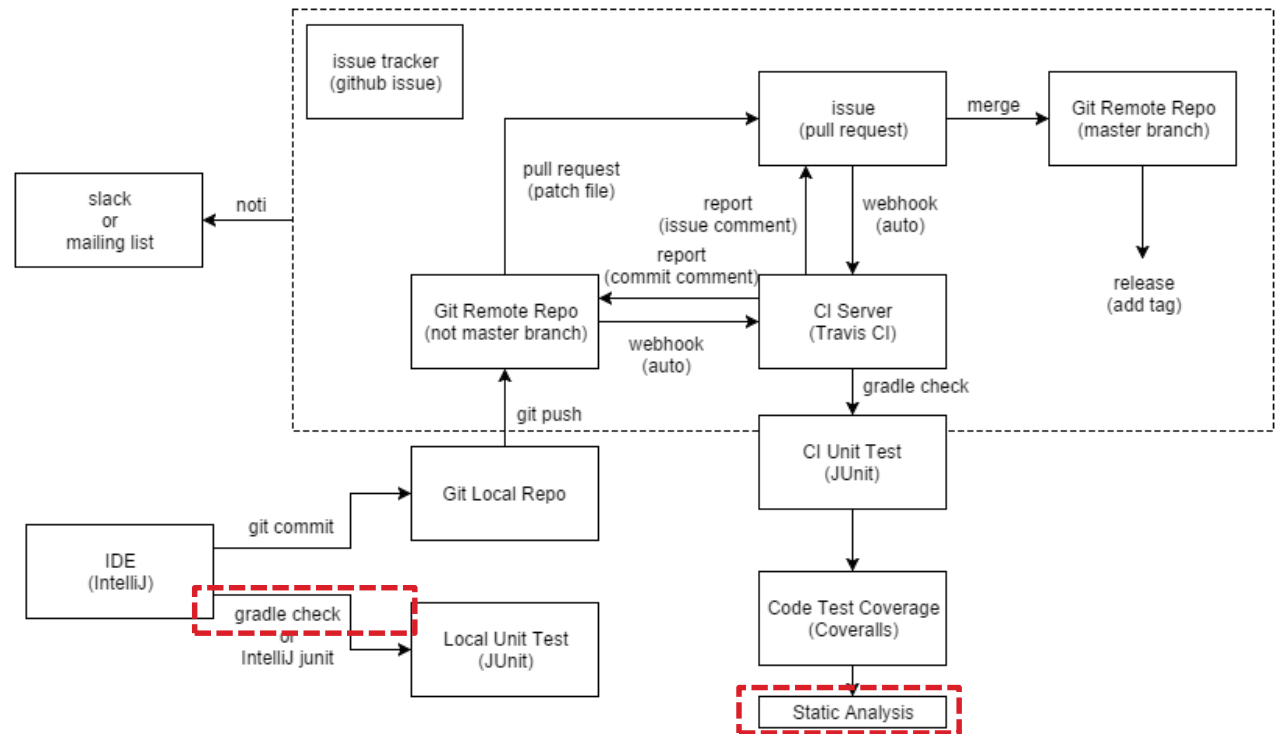
왕홍강

김태영

Overview



Overview



Overview

Static Analysis

- Checkstyle
- PMD
- Findbugs
- JDepend

Test Management

- TestRail

Checkstyle?

코딩 스타일을 체크해주는 툴로 프로젝트를 진행하는 개발자들이 동일한(표준의) 코딩 스타일을 사용 하도록 함

- IntelliJ Plugin : 개발자들이 코드 작성 도중 사용
- Gradle Plugin : 로컬/CI 서버에서 사용

Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

[링크\(https://plugins.jetbrains.com/plugin/1065\)](https://plugins.jetbrains.com/plugin/1065)에서 CheckStyle-IDEA 다운로드

CheckStyle-IDEA

Products: [IntelliJ IDEA](#), [Android Studio](#), [Rider](#)

Vendor: **James Shiell**

Email: [james at infernus.org](mailto:james@infernus.org)

Website: <https://github.com/jshiell/checkstyle-idea>

Source code: [CheckStyle-IDEA_1065.zip](#)

Authors:

[jshiell](#)

This plugin provides both real-time and on-demand scanning of Java files with CheckStyle 6.17 from within IDEA.

Downloads: **693566**

Rating: 

Participated in rating: **12**

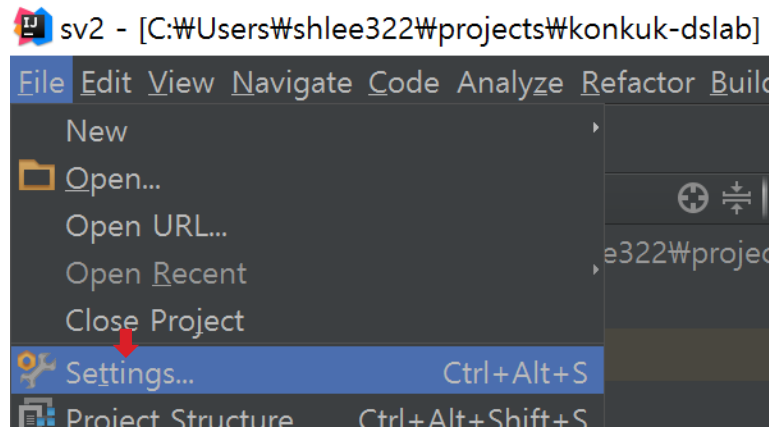
[Comment and rate](#)

Version	Compatible builds	Date	 Download
4.27.3	141.178 +	03.04.2016	

Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

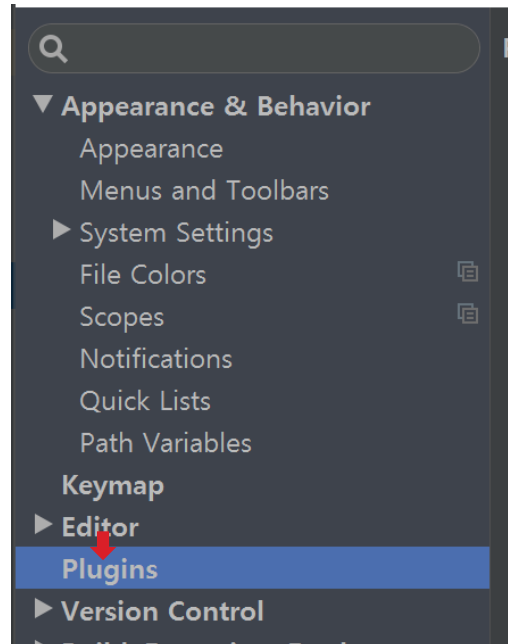
File > Settings 클릭



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

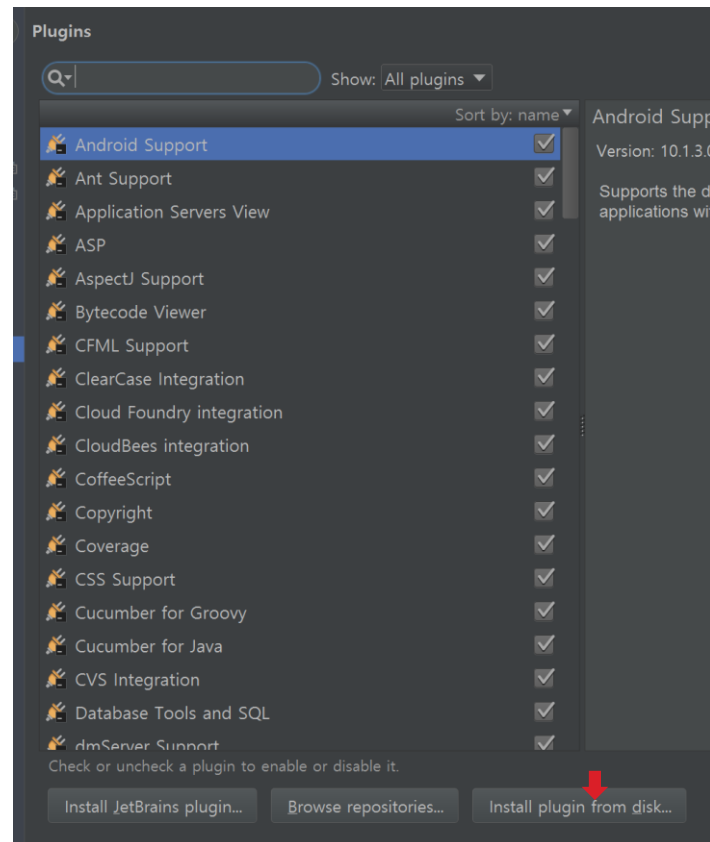
File > Settings 클릭 > 좌측 메뉴에서 Plugins 클릭



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

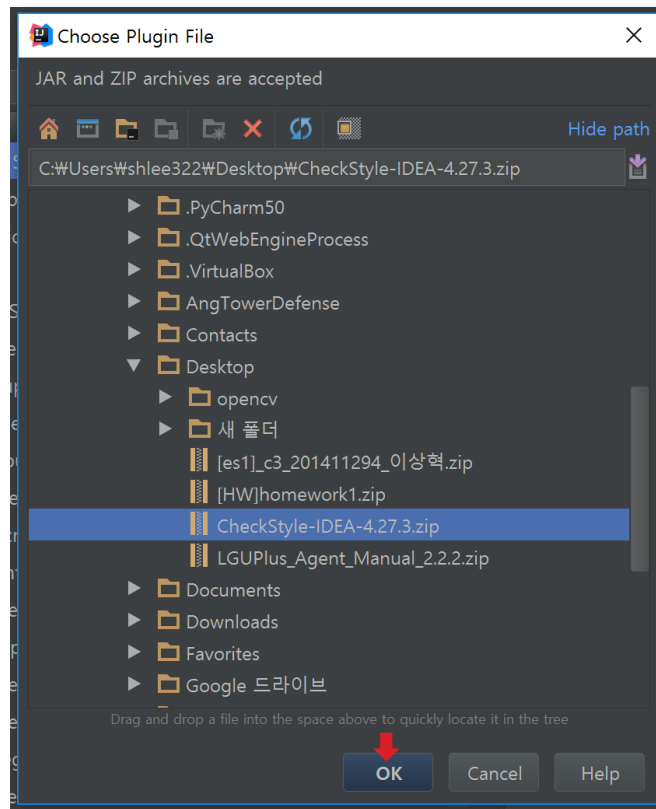
File > Settings 클릭 > 좌측 메뉴에서 Plugins 클릭 > Install plugin from disk 클릭



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

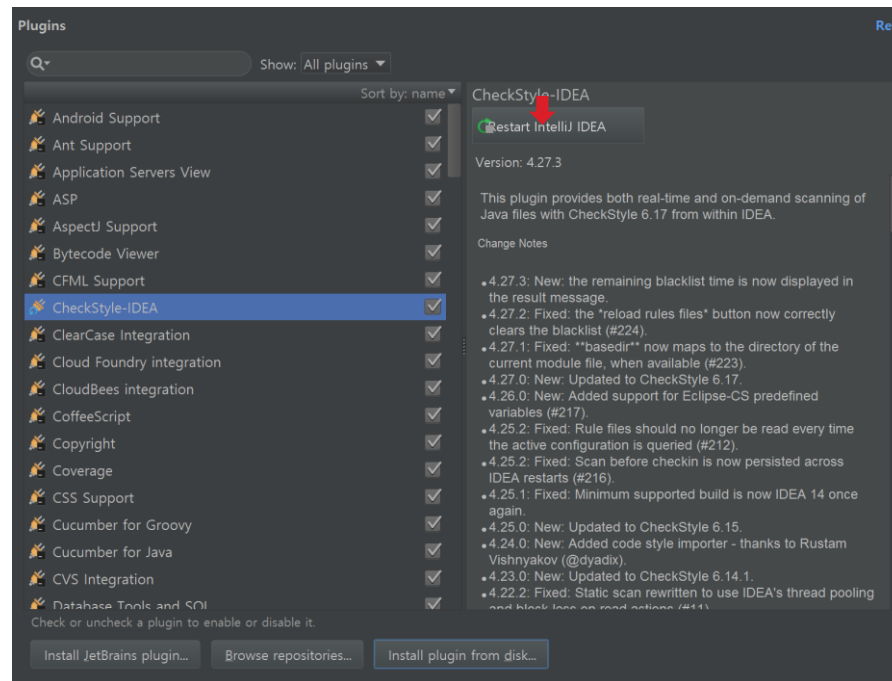
File > Settings 클릭 > 좌측 메뉴에서 Plugins 클릭 > Install plugin from disk 클릭 > 파일 선택 후 OK 클릭



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

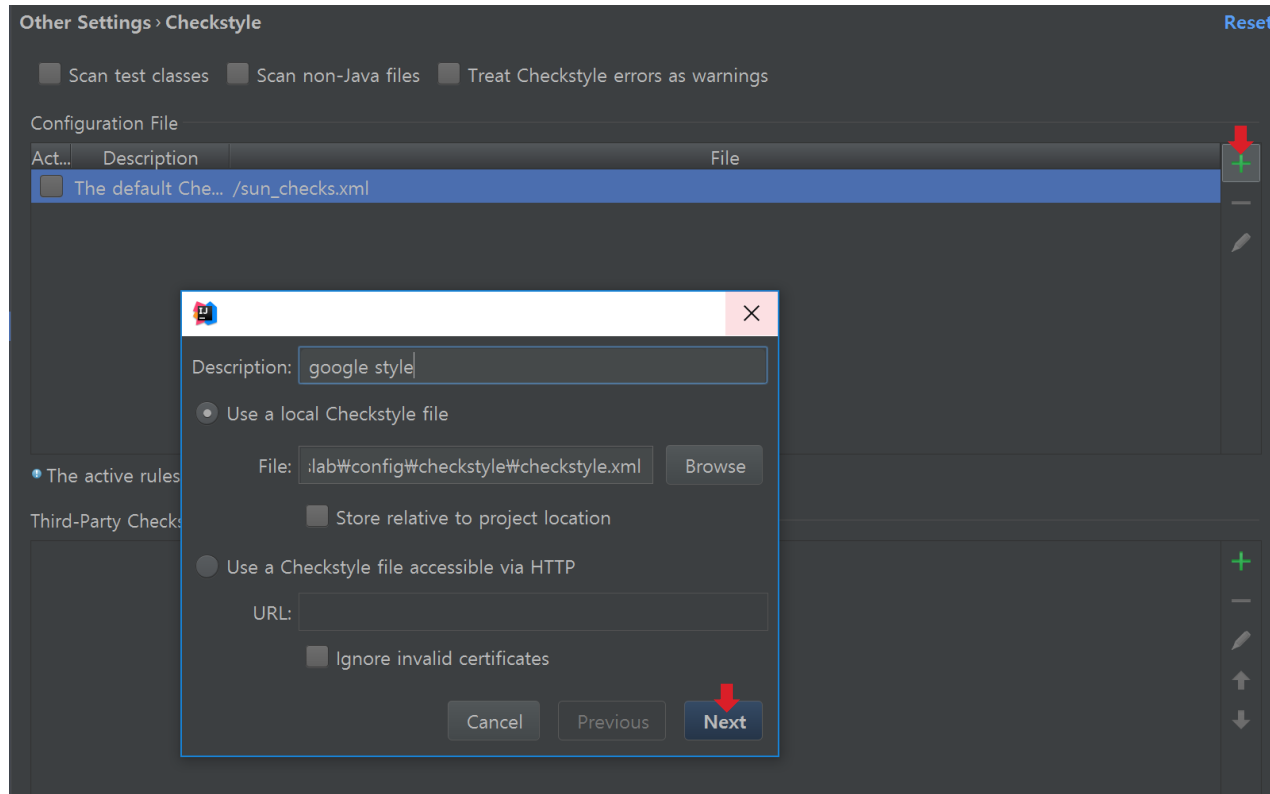
Restart IntelliJ IDEA 클릭 후 설치 확인



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

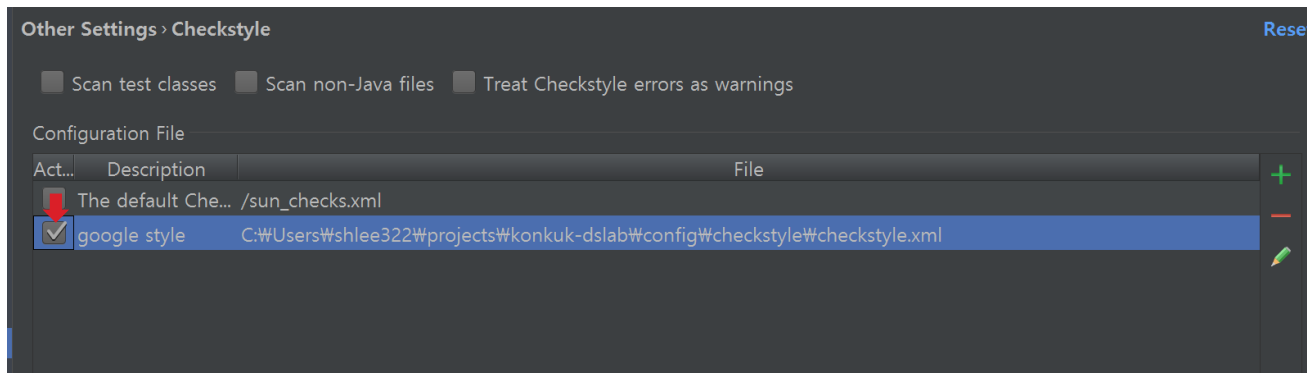
[Google Java Style](#) 다운로드 > path/config/checkstyle/checkstyle.xml 으로 저장
> Settings - Other Settings - Checkstyle에서 Config 추가



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

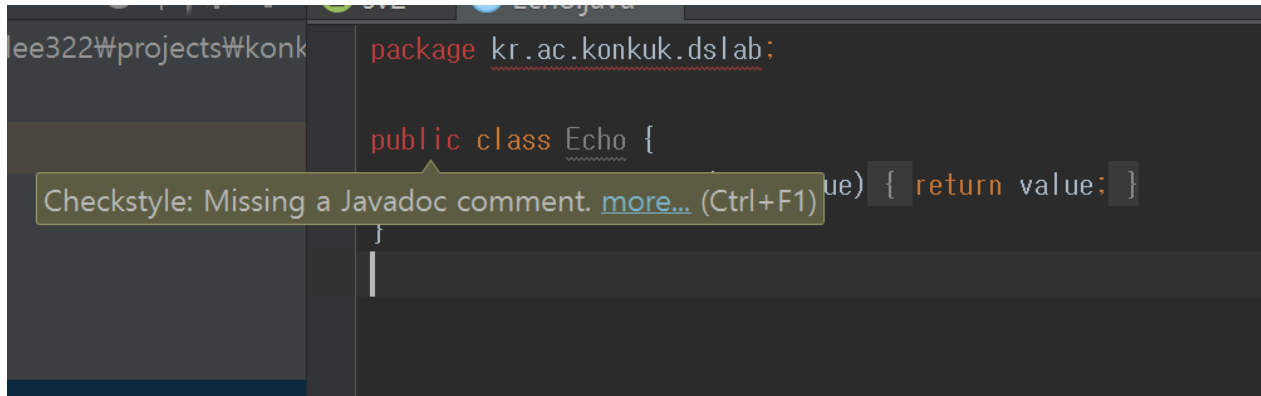
[Google Java Style](#) 다운로드 > path/config/checkstyle/checkstyle.xml 으로 저장
> Settings - Other Settings - Checkstyle에서 Config 추가 > 추가한 config로 변경 후 OK



Checkstyle – IntelliJ Plugin

IntelliJ Plugin 설치

설정 확인



The screenshot shows a code editor window with the following Java code:

```
package kr.ac.konkuk.dslab;  
  
public class Echo {  
    public String echo(String value) { return value; }  
}
```

A tooltip message is displayed over the code, stating: "Checkstyle: Missing a Javadoc comment. [more...](#) (Ctrl+F1)".

Checkstyle – Gradle Plugin

Gradle Plugin 설치

build.gradle에 `apply plugin: 'checkstyle'`, checkstyle 설정 추가

```
apply plugin: 'java'
apply plugin: 'cobertura'
apply plugin: 'checkstyle'

checkstyle {
    configFile = new File(rootDir, "config/checkstyle/checkstyle.xml")
    ignoreFailures = true
    sourceSets = [sourceSets.main]
    toolVersion = "6.17"
}
```

Checkstyle – Gradle Plugin

Gradle Plugin 설치

수정한 build.gradle과 checkstyle.xml 커밋, 푸시

```
b# git add config/checkstyle/checkstyle.xml
b# git add build.gradle
b# git commit -m "add checkstyle"
b# git push
```


Checkstyle – Gradle Plugin

Gradle Plugin 설치

CI 서버에서 동작 확인

```
274 [ant:checkstyle] [WARN] /home/travis/build/shlee322/konkuk-dslab/src/main/java/Echo.java:4: 'method def modifier' have incorrect indentation level 4,
    expected level should be 2. [Indentation]
275 [ant:checkstyle] [WARN] /home/travis/build/shlee322/konkuk-dslab/src/main/java/Echo.java:5: 'method def' child have incorrect indentation level 8,
    expected level should be 4. [Indentation]
276 [ant:checkstyle] [WARN] /home/travis/build/shlee322/konkuk-dslab/src/main/java/Echo.java:6: 'method def rcurly' have incorrect indentation level 4,
    expected level should be 2. [Indentation]
```

PMD - Program May Dependable

PMD는 자바 소스 코드 분석기로 사용하지 않는 변수, 아무 처리도 안하는 catch 블록, 불필요한 객체 생성 등등을 찾아냅니다.

Checkstyle와 동일한 방식으로 설치 (IntelliJ, gradle 모두 동일)

```
apply plugin: 'pmd'

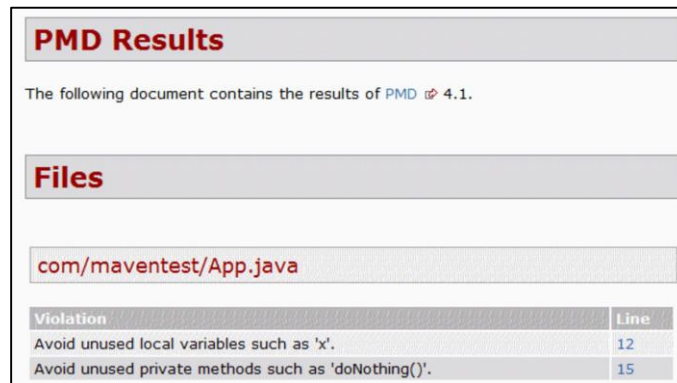
pmd {
    ignoreFailures = true
    sourceSets = [sourceSets.main]
    ruleSets = ["java-basic", "java-strings", "java-braces"]
}

tasks.withType(Pmd) {
    reports {
        xml.enabled false
        html.enabled true
    }
}
```

build.gradle 일부

PMD - Program May Dependable

PMD는 자바 소스 코드 분석기로 사용하지 않는 변수, 아무 처리도 안하는 catch 블록, 불필요한 객체 생성 등등을 찾아냅니다.



PMD Results

The following document contains the results of [PMD 4.1](#).

Files

[com/maventest/App.java](#)

Violation	Line
Avoid unused local variables such as 'x'.	12
Avoid unused private methods such as 'doNothing()'.	15

PMD results

Findbugs

Findbugs는 버그 패턴을 찾아 자동으로 알려주는 정적 분석 툴입니다. (byte code기반으로 분석 진행)

```
apply plugin: 'findbugs'

findbugs {
    ignoreFailures = true
    toolVersion = "2.0.1"
    sourceSets = [sourceSets.main]
    effort = "max"
}

tasks.withType(FindBugs) {
    reports {
        xml.enabled = false
        html.enabled = true
    }
}
```

build.gradle 일부

Findbugs

Findbugs는 버그 패턴을 찾아 자동으로 알려주는 정적 분석 툴입니다. (byte code기반으로 분석 진행)

FindBugs Report

Project Information

Project:
FindBugs version: 2.0.1
Code analyzed:

- /home/shlee322/projects/konkuk-dslab/build/classes/main/Echo.class
- /home/shlee322/projects/konkuk-dslab/build/classes/main/kr/ac/konkuk/dslab/Echo.class

Metrics

10 lines of code analyzed, in 2 classes, in 2 packages.

Metric	Total	Density*
High Priority Warnings		0.00
Medium Priority Warnings		0.00
Total Warnings	0	0.00

FindBugs results

JDepend

패키지 의존성과 관련된 설계 품질의 객체지향식 측정을 통해,
시스템 디자인을 효과적으로 분석하고 관리할 수 있도록 지원하는 도구

- 패키지별로 의존성 측정 가능
- 패키지 의존성과 관련된 데이터 품질을 수치화하여 표현
- 수치화된 데이터 품질을 그래프로 표현

```
apply plugin: 'jdepend'

task jdependReport(dependsOn: jdependMain, group: 'verification', description: 'Generates a HTML report after executing jdepend checks.') << {
    ant.xslt(
        in: "$reporting.baseDir/jdepend/main.xml",
        style: "$rootDir/config/jdepend/jdepend-report.xslt",
        out: "$reporting.baseDir/jdepend/main.html")
    }

check {
    dependsOn += ['jdependReport']
}
```

build.gradle 일부 ([style 다운로드](#))

JDepend

패키지 의존성과 관련된 설계 품질의 객체지향식 측정을 통해,
시스템 디자인을 효과적으로 분석하고 관리할 수 있도록 지원하는 도구

- 패키지별로 의존성 측정 가능
- 패키지 의존성과 관련된 데이터 품질을 수치화하여 표현
- 수치화된 데이터 품질을 그래프로 표현

JDepend Analysis Designed for use with [JDepend](#) and [Ant](#).

Summary [\[summary\]](#) [\[packages\]](#) [\[cycles\]](#) [\[explanations\]](#)

Package	Total Classes	Abstract Classes	Concrete Classes	Afferent Couplings	Efferent Couplings	Abstractness	Instability	Distance
Default	1	0	1	0	1	0	1	0
kr.ac.konkuk.dslab	1	0	1	0	1	0	1	0
java.lang	No stats available: package referenced, but not analyzed.							

Packages [\[summary\]](#) [\[packages\]](#) [\[cycles\]](#) [\[explanations\]](#)

Default

[Afferent Couplings](#): 0 [Efferent Couplings](#): 1 [Abstractness](#): 0 [Instability](#): 1 [Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	Echo	None	java.lang

kr.ac.konkuk.dslab

[Afferent Couplings](#): 0 [Efferent Couplings](#): 1 [Abstractness](#): 0 [Instability](#): 1 [Distance](#): 0

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
None	kr.ac.konkuk.dslab.Echo	None	java.lang

Cycles [\[summary\]](#) [\[packages\]](#) [\[cycles\]](#) [\[explanations\]](#)

There are no cyclic dependencies.

JDepend results

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

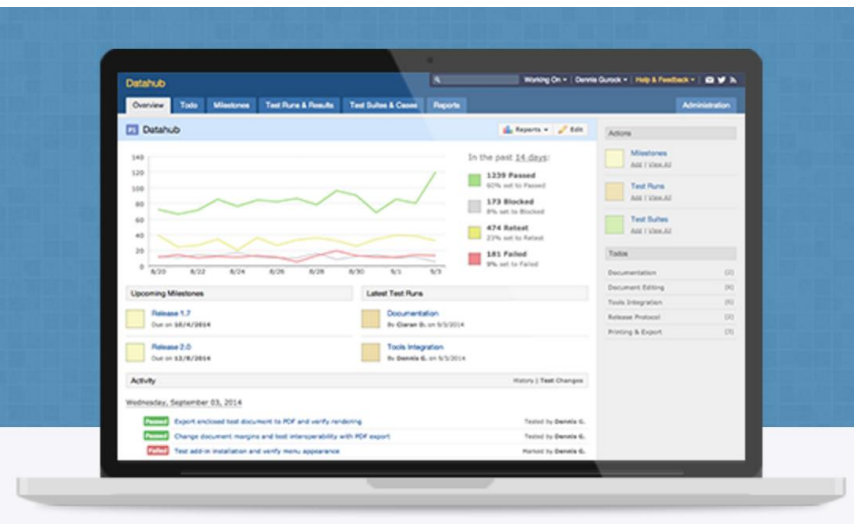
GitHub Test Management with TestRail

Integrate our test case management software with GitHub

- Manage test cases, plans and runs
- Start test runs and capture test results
- Push bug reports to GitHub and link issue IDs
- Track test results, activity and progress

Try TestRail For Free

See Features



TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

Get started with your free TestRail trial in minutes

- TestRail Cloud - Try the online edition of TestRail**
We will instantly set up your account. You can switch to a local server later.
- TestRail Server - Download TestRail for your server**
Download TestRail and easily install it on a Windows Server or Linux.

Trial Details

Web Address *
Your account will be available under this address.

Your Name *

Email Address *

Organization

Organization

Expected Users

Phone Number
Highly recommended for optional phone support.

Create TestRail Trial **I accept the Terms of Service**

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

Get started with your free TestRail trial in minutes

- TestRail Cloud - Try the online edition of TestRail**
We will instantly set up your account. You can switch to a local server later.
- TestRail Server - Download TestRail for your server**
Download TestRail and easily install it on a Windows Server or Linux.

Trial Details

Web Address *
Your account will be available under this address.

Your Name *

Email Address *

Organization

Organization

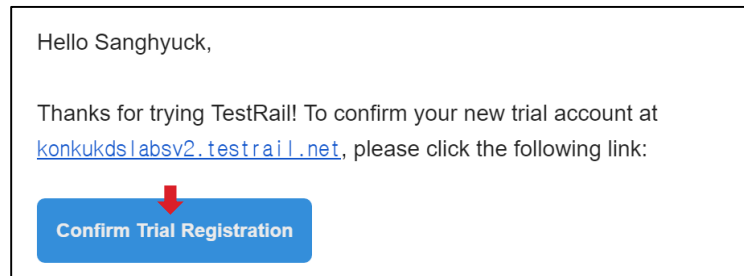
Expected Users

Phone Number
Highly recommended for optional phone support.

Create TestRail Trial **I accept the Terms of Service**

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.



이메일 확인

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

The screenshot displays the TestRail QA dashboard. At the top, the navigation bar includes 'Working On', 'Sanghyuck Lee', and 'Help & Feedback'. The main header shows 'TestRail QA' and 'Dashboard' (with 'Administration' also visible). The 'All Projects' section features a yellow informational box titled 'Add your first project to TestRail' with an 'Add Project' button. To the right, a 'New to TestRail?' section provides a link to the 'Getting Started' tutorial. A sidebar on the right shows 'Add Project', '0 active and 0 completed projects.', and a 'Todos' list. A blue arrow points from a 'Welcome to TestRail!' message box to the first step of a bottom navigation bar: 'Create a project' (Datahub icon). The other steps are 'Add cases' (Cases icon), 'Start a run' (Run icon), 'Add results' (Results icon), 'Invite users' (Users icon), and 'Set up integration' (Integrations icon).

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

Add Project

Project Access Defects References

Name *

Ex: *New Widget, Intranet or Payroll Software*

Announcement

|

You can post an announcement to the project overview page. This could include links to the project's issue tracker or knowledge base, for example.

Show the announcement on the overview page

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

The screenshot shows the TestRail interface for a project named 'sv2'. The top navigation bar includes tabs for Overview, Todo, Milestones, Test Runs & Results, Test Cases, Reports, and Administration. A 'P1 sv2' label is visible on the left, and 'Reports' and 'Edit' options are on the right.

A central yellow notification box reads: "Congratulations! You have created your first project. Now that you have created your first project in TestRail, you can add your project milestones and create your first test cases. If you are new to TestRail, it is highly recommended to take a look at TestRail's [user guide and manual](#)." To the right of this box, "Next recommended steps" are listed: Read [Getting Started](#), Add [project milestones](#), Enter [test cases](#), and Add [additional users](#).

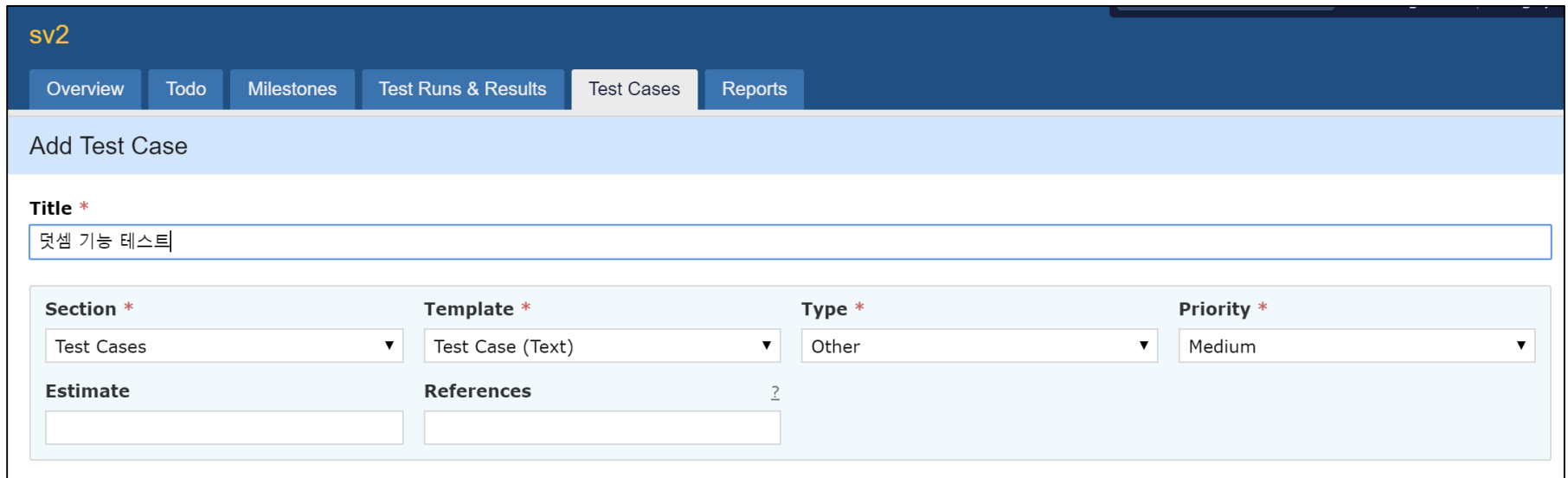
Below the notification is a line chart showing progress over time from 4/1 to 4/15. The y-axis ranges from 0 to 6. To the right of the chart, a summary for the past 14 days shows: 0 Passed (0% set to Passed), 0 Blocked (0% set to Blocked), 0 Retest (0% set to Retest), and 0 Failed (0% set to Failed).

At the bottom, there are two sections: "Upcoming Milestones" and "Latest Test Runs". Both sections indicate that the project does not contain any active milestones or test runs, respectively, and provide an "Add Milestone" and "Add Test Run" button.

On the right side, there is an "Actions" sidebar with links for "Milestones" (Add | View All), "Test Runs" (Add | View All), and "Test Cases" (View All). Below this is a "Todos" section which is currently empty, showing "None."

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.



The screenshot shows the 'Add Test Case' form in TestRail. The interface has a dark blue header with the project name 'sv2' and a navigation bar with tabs: Overview, Todo, Milestones, Test Runs & Results, Test Cases (selected), and Reports. The form itself is light blue and contains the following fields:

- Title ***: A text input field containing '덧셈 기능 테스트'.
- Section ***: A dropdown menu with 'Test Cases' selected.
- Template ***: A dropdown menu with 'Test Case (Text)' selected.
- Type ***: A dropdown menu with 'Other' selected.
- Priority ***: A dropdown menu with 'Medium' selected.
- Estimate**: An empty text input field.
- References**: An empty text input field with a help icon (?) to its right.

TestRail

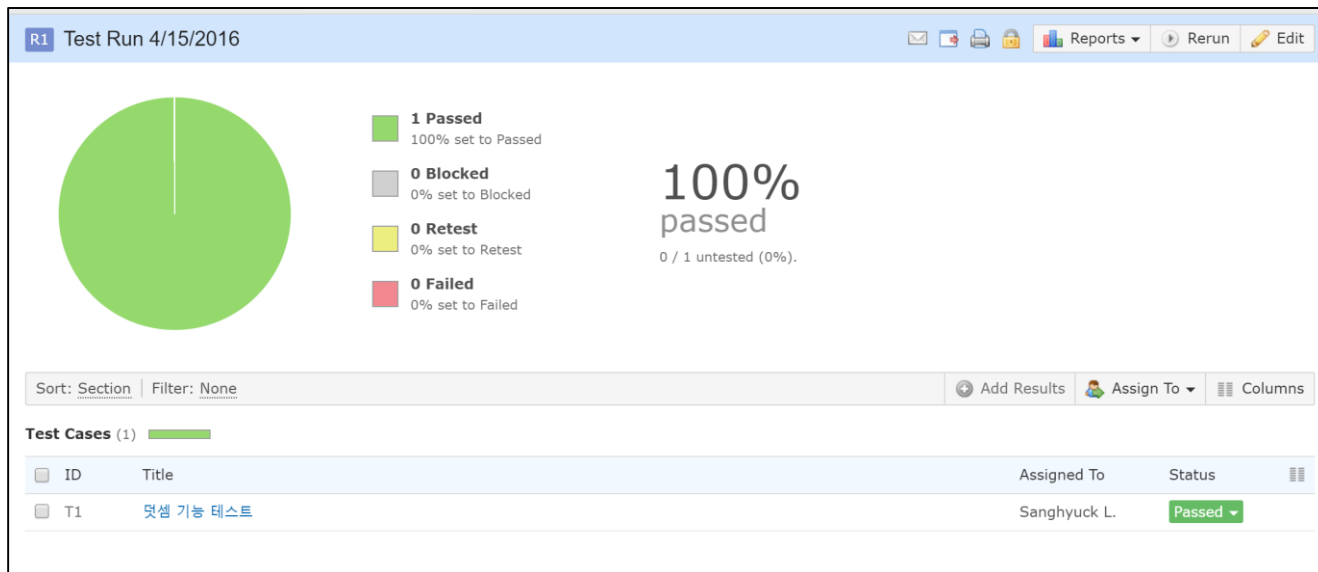
TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.

The screenshot displays the TestRail interface for a specific test run. At the top, there are navigation tabs: Overview, Todo, Milestones, Test Runs & Results (selected), Test Cases, and Reports. The main content area shows a summary for 'Test Run 4/15/2016'. A pie chart indicates that 100% of the test cases are 'Untested'. To the right of the chart, a list shows the status counts: 0 Passed, 0 Blocked, 0 Retest, and 0 Failed. A large '0% passed' text is prominently displayed, with a sub-note '1 / 1 untested (100%)'. Below the summary, there are controls for sorting and filtering, and a table of test cases. The table has columns for ID, Title, Assigned To, and Status. One test case is listed with ID 'T1' and title '맞춤 기능 테스트', assigned to 'Sanghyuck L.', and its status is currently 'Untested'. A dropdown menu is open for the 'Status' column, showing options: Passed, Blocked, Retest, and Failed.

ID	Title	Assigned To	Status
T1	맞춤 기능 테스트	Sanghyuck L.	Untested

TestRail

TestRail은 Test Management 툴로 전체적인 테스트를 관리하는 툴입니다.



추가 : Github Issue Tracker

추가적으로 간단하게 Github Issue Tracker의 Label 기능을 이용하면 이를 흉내 낼 수 있습니다.

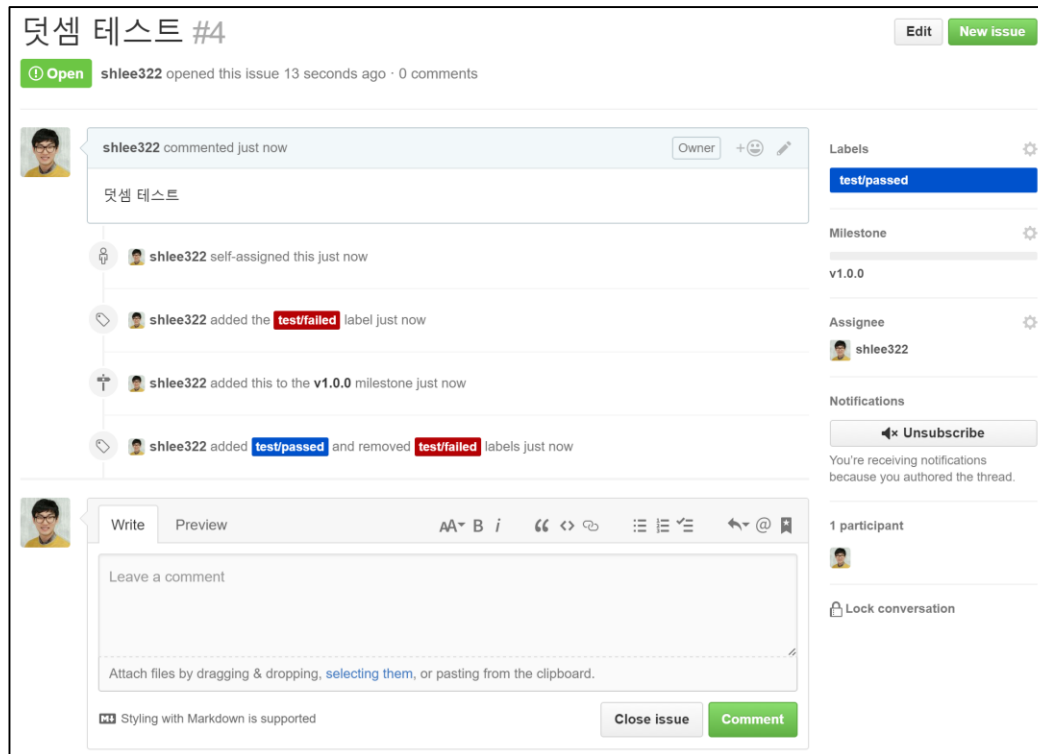


11 labels Sort ▾

	0 open issues	 Edit	 Delete
	0 open issues	 Edit	 Delete
	0 open issues	 Edit	 Delete
	0 open issues	 Edit	 Delete

추가 : Github Issue Tracker

추가적으로 간단하게 Github Issue Tracker의 Label 기능을 이용하면 이를 흉내 낼 수 있습니다.



The screenshot displays a GitHub issue titled "덧셈 테스트 #4". The issue is currently open and was created 13 seconds ago by user shlee322. The issue description is "덧셈 테스트".

The comment history shows the following actions:

- shlee322 commented just now: 덧셈 테스트
- shlee322 self-assigned this just now
- shlee322 added the **test/failed** label just now
- shlee322 added this to the **v1.0.0** milestone just now
- shlee322 added **test/passed** and removed **test/failed** labels just now

The right sidebar contains the following settings:

- Labels: **test/passed**
- Milestone: **v1.0.0**
- Assignee: **shlee322**
- Notifications: **Unsubscribe** (You're receiving notifications because you authored the thread.)
- 1 participant
- Lock conversation

The bottom of the page features a comment input area with a "Write" tab, a "Preview" tab, and a rich text editor. The input area contains the text "Leave a comment" and a note: "Attach files by dragging & dropping, selecting them, or pasting from the clipboard." The "Close issue" button is disabled, and the "Comment" button is active.