

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสสนศ. 663380029-9 Section 1

Lab#12 – Analyze the quality of your software by using software metrics

วัตถุประสงค์การเรียนรู้

1. ผู้เรียนสามารถอธิบายความหมายของ Software metrics ที่วัดออกมาได้อย่างถูกต้อง
2. ผู้เรียนสามารถใช้ Static code analysis tools ในการวิเคราะห์ปัญหาคุณภาพของซอฟต์แวร์ได้
3. ผู้เรียนสามารถอธิบายความแตกต่างระหว่าง Static testing และ Dynamic testing ได้อย่างถูกต้อง

กิจกรรมที่ 12.1: การวิเคราะห์ปัญหาในโค้ดด้วย PMD

- (1) ติดตั้ง PMD plugin ลงใน Eclipse ให้เรียบร้อย
- (2) สร้าง local repository ของ git และตั้งค่าให้เรียบร้อย Workspace ให้เรียบร้อย
- (3) Clone โค้ดตั้งต้นจาก

https://github.com/ChitsuthaCSKKU/SOA/tree/2025/Assignment/Lab12_SoftwareMetrics

- (4) ศึกษาการทำงานและโครงสร้างของ Source code ที่กำหนดให้
- (5) วิเคราะห์ปัญหาใน Source code โดยใช้ PMD
- (6) Capture หน้าจอ หรือ Export รายงานผลการทดสอบ แล้วตอบคำถามต่อไปนี้

Q1: PMD พบ Error และ Warning ใน Whitboard.java กี่ตัว (Capture หน้าจอประกอบคำตอบ)

ตอบ `pmd check -d Whiteboard/src/main -R`

`category/java/bestpractices.xml,category/java/codestyle.xml,category/java/design.xml,category/java/documentation.xml,category/java/errorprone.xml,category/java/multithreading.xml,category/java/performance.xml,category/java/security.xml,category/java/errorprone.xml,category/java/complexity.xml -f html -r pmd-report.html`

เอกสารประกอบชั่วโมงปฏิบัติการ

วิชา CP353201 Software Quality Assurance

ปีการศึกษา 1/2568

ผศ. ดร. ชิตสุธา สุ่มเล็ก

สาขาวิชาวิทยาการคอมพิวเตอร์ วิทยาลัยการคอมพิวเตอร์ มข.

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสสนศ. 663380029-9 Section 1

PMD report			
Problems found			
#	File	Line	Problem
1	Whiteboard/src/main/UserWhiteboard.java	3	Class comments are required
2	Whiteboard/src/main/UserWhiteboard.java	5	Field comments are required
3	Whiteboard/src/main/UserWhiteboard.java	6	Field comments are required
4	Whiteboard/src/main/UserWhiteboard.java	8	Protected method constructor comments are required
5	Whiteboard/src/main/UserWhiteboard.java	8	Parameter 'userId' is not assigned and could be declared final
6	Whiteboard/src/main/UserWhiteboard.java	11	Usage of System.out/err
7	Whiteboard/src/main/UserWhiteboard.java	18	Public method and constructor comments are required
8	Whiteboard/src/main/UserWhiteboard.java	18	Parameter 'text' is not assigned and could be declared final
9	Whiteboard/src/main/UserWhiteboard.java	20	Usage of System.out/err
1	Whiteboard/src/main/Whiteboard.java	3	All methods are static. Consider adding a private no-args constructor to prevent instantiation.
2	Whiteboard/src/main/Whiteboard.java	3	Class comments are required
3	Whiteboard/src/main/Whiteboard.java	5	Public method and constructor comments are required
4	Whiteboard/src/main/Whiteboard.java	7	Local variable 'managerA' could be declared final
5	Whiteboard/src/main/Whiteboard.java	8	Local variable 'managerB' could be declared final
6	Whiteboard/src/main/Whiteboard.java	10	Usage of System.out/err
7	Whiteboard/src/main/Whiteboard.java	11	Usage of System.out/err
8	Whiteboard/src/main/Whiteboard.java	12	Usage of System.out/err
9	Whiteboard/src/main/Whiteboard.java	13	Comment is too large: Line too long
10	Whiteboard/src/main/Whiteboard.java	14	Usage of System.out/err
11	Whiteboard/src/main/Whiteboard.java	17	Local variable 'userId1' could be declared final
12	Whiteboard/src/main/Whiteboard.java	18	Local variable 'userId2' could be declared final
13	Whiteboard/src/main/Whiteboard.java	20	Usage of System.out/err
14	Whiteboard/src/main/Whiteboard.java	22	Local variable 'aliceWb1' could be declared final
15	Whiteboard/src/main/Whiteboard.java	25	Local variable 'bobWb' could be declared final
16	Whiteboard/src/main/Whiteboard.java	28	Local variable 'aliceWb2' could be declared final
17	Whiteboard/src/main/Whiteboard.java	32	Usage of System.out/err
18	Whiteboard/src/main/Whiteboard.java	34	Usage of System.out/err
19	Whiteboard/src/main/Whiteboard.java	35	Usage of System.out/err
20	Whiteboard/src/main/Whiteboard.java	36	Usage of System.out/err
21	Whiteboard/src/main/Whiteboard.java	37	Usage of System.out/err

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสสนศ. 663380029-9 Section 1

4	Whiteboard/src/main/Whiteboard.java	7	Local variable 'managerA' could be declared final
5	Whiteboard/src/main/Whiteboard.java	8	Local variable 'managerB' could be declared final
6	Whiteboard/src/main/Whiteboard.java	10	Usage of System.out/err
7	Whiteboard/src/main/Whiteboard.java	11	Usage of System.out/err
8	Whiteboard/src/main/Whiteboard.java	12	Usage of System.out/err
9	Whiteboard/src/main/Whiteboard.java	13	Comment is too large: Line too long
10	Whiteboard/src/main/Whiteboard.java	14	Usage of System.out/err
11	Whiteboard/src/main/Whiteboard.java	17	Local variable 'userId1' could be declared final
12	Whiteboard/src/main/Whiteboard.java	18	Local variable 'userId2' could be declared final
13	Whiteboard/src/main/Whiteboard.java	20	Usage of System.out/err
14	Whiteboard/src/main/Whiteboard.java	22	Local variable 'aliceWb1' could be declared final
15	Whiteboard/src/main/Whiteboard.java	25	Local variable 'bobWb' could be declared final
16	Whiteboard/src/main/Whiteboard.java	28	Local variable 'aliceWb2' could be declared final
17	Whiteboard/src/main/Whiteboard.java	32	Usage of System.out/err
18	Whiteboard/src/main/Whiteboard.java	34	Usage of System.out/err
19	Whiteboard/src/main/Whiteboard.java	35	Usage of System.out/err
20	Whiteboard/src/main/Whiteboard.java	36	Usage of System.out/err
21	Whiteboard/src/main/Whiteboard.java	37	Usage of System.out/err
22	Whiteboard/src/main/Whiteboard.java	40	Usage of System.out/err
23	Whiteboard/src/main/Whiteboard.java	41	Usage of System.out/err
24	Whiteboard/src/main/Whiteboard.java	42	Usage of System.out/err
25	Whiteboard/src/main/Whiteboard.java	44	Usage of System.out/err
1	Whiteboard/src/main/WhiteboardManager.java	6	Class comments are required
2	Whiteboard/src/main/WhiteboardManager.java	8	Field comments are required
3	Whiteboard/src/main/WhiteboardManager.java	20	Field comments are required
4	Whiteboard/src/main/WhiteboardManager.java	20	Fields should be declared at the top of the class, before any method declarations, constructors, initializers or inner classes.
5	Whiteboard/src/main/WhiteboardManager.java	22	Public method and constructor comments are required
6	Whiteboard/src/main/WhiteboardManager.java	22	Parameter 'userId' is not assigned and could be declared final
7	Whiteboard/src/main/WhiteboardManager.java	29	Public method and constructor comments are required

Configuration errors	
Rule	Problem
LoosePackageCoupling	No packages or classes specified

Q2: จงอธิบายเกี่ยวกับปัญหา “Local variable 'managerB' could be declared final” ว่าเป็นปัญหาอะไร และมีแนวทางในการแก้ไขอย่างไร

ตอบปัญหานี้เกิดขึ้นเพราะตัวแปร managerB เป็นตัวแปรภายในเมธอด (local variable) ที่ถูกกำหนดค่าเพียงครั้งเดียว และไม่มีมีการเปลี่ยนแปลงค่าหลังจากนั้น PMD จึงแจ้งเตือนว่า ตัวแปรนี้สามารถประกาศเป็น final ได้ เพื่อบอกให้ชัดเจนว่า ค่าของตัวแปรจะไม่ถูกแก้ไขภายหลัง ซึ่งช่วยเพิ่มความปลอดภัยของโค้ดและทำให้โค้ดอ่านง่ายขึ้น

แนวทางการแก้ไข:

เพิ่มคีย์เวิร์ด final ขณะประกาศตัวแปร เช่น

```
final WhiteboardManager managerB = new WhiteboardManager();
```

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสศนศ. 663380029-9 Section 1

Q3: PMD ใช้ Metrics อะไรบ้างในการระบุ God class และแต่ละตัวมีความหมายว่าอะไรบ้าง

ตอบ

ในการตรวจหาคลาสที่เข้าข่าย “God Class” (คลาสที่มีหน้าที่มากเกินไปและละเมิดหลักการ Single Responsibility) PMD จะใช้ **Metrics หลัก 3 ตัว** ในการวิเคราะห์ ได้แก่:

1. WMC (Weighted Methods per Class)

เป็นตัววัด จำนวนและความซับซ้อน ของเมธอดทั้งหมดในคลาส

ค่าที่สูง บ่งบอกว่าคลาสมีพฤติกรรมมากเกินไป อาจรับผิดชอบหลายหน้าที่เกินควร

2. ATFD (Access to Foreign Data)

- วัดจำนวนครั้งที่คลาสเข้าถึงข้อมูลจากคลาสอื่น (เช่น การเรียกใช้ field หรือ method ของคลาสอื่น)
- ถ้าค่าสูง แสดงว่าคลาสนี้พึ่งพาข้อมูลภายนอกมากเกินไป มีแนวโน้มเป็น “ตัวควบคุมใหญ่” ที่รู้เรื่องทุกอย่าง

3. TCC (Tight Class Cohesion)

- วัดระดับความสัมพันธ์ระหว่างเมธอดภายในคลาสเอง (เช่น เมธอดใช้ฟิลด์เดียวกันบ่อยแค่ไหน)
- ถ้าค่าต่ำ หมายถึงเมธอดในคลาสแทบไม่เกี่ยวข้องกัน เป็นสัญญาณว่าเป็นคลาสที่รวมหลายหน้าที่ไว้ในทีเดียว

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสสนศ. 663380029-9 Section 1

กิจกรรมที่ 12.2: การวิเคราะห์ปัญหาในโค้ดด้วย Checkstyle

- (1) ติดตั้ง checkstyle plugin ลงใน Eclipse ให้เรียบร้อย
- (2) สร้าง local repository ของ git และตั้งค่าให้เรียบร้อย Workspace ให้เรียบร้อย
- (3) Clone โค้ดตั้งต้นจาก

https://github.com/ChitsuthaCSKKU/SOA/tree/2025/Assignment/Lab12_SoftwareMetrics

- (4) ศึกษาการทำงานและโครงสร้างของ Source code ที่กำหนดให้
- (5) วิเคราะห์ปัญหาใน Source code โดยใช้ checkstyle
- (6) Capture หน้าจอ หรือ Export รายงานผลการทดสอบ แล้วตอบคำถามต่อไปนี้

Q4: จงระบุปัญหาการเขียน Code ที่ไม่ตรงกับ Coding standard ของ Java ที่ Checkstyle ระบุมา 3 รายการ พร้อมคำอธิบายว่า Code ส่วนนี้ไม่สอดคล้องกับมาตรฐานอย่างไร (Capture หน้าจอประกอบด้วย)

ตอบ

#	File	Line	Problem
1	Whiteboard/src/main/UserWhiteboard.java	3	Class comments are required
2	Whiteboard/src/main/UserWhiteboard.java	5	Field comments are required
3	Whiteboard/src/main/UserWhiteboard.java	6	Field comments are required

ปัญหา: Class comments are required

คำอธิบาย:

มาตรฐานการเขียนโค้ดของ Java (Java Coding Conventions) แนะนำให้มี Javadoc comment ด้านบนคลาสทุกครั้ง เพื่ออธิบายหน้าที่และการใช้งานของคลาสนั้น การละเว้นการเขียน comment ทำให้โค้ดขาดความชัดเจนและยากต่อการดูแลในระยะยาว

6	Whiteboard/src/main/Whiteboard.java	10	Usage of System.out/err
7	Whiteboard/src/main/Whiteboard.java	11	Usage of System.out/err
8	Whiteboard/src/main/Whiteboard.java	12	Usage of System.out/err

ปัญหา: Usage of System.out/err

คำอธิบาย:

มาตรฐานของ Java ไม่แนะนำให้ใช้ System.out.println สำหรับการ log ข้อมูลในระบบจริง ควรใช้ logging framework (เช่น java.util.logging, Log4j, SLF4J) เพื่อให้ควบคุมระดับ log และเก็บบันทึกได้เหมาะสม

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสสนศ. 663380029-9 Section 1

2	Whiteboard/src/main/WhiteboardManager.java	8	Field comments are required
3	Whiteboard/src/main/WhiteboardManager.java	20	Field comments are required
4	Whiteboard/src/main/WhiteboardManager.java	20	Fields should be declared at the top of the class, before any method declarations, constructors, initializers or inner classes.

ปัญหา: Fields should be declared at the top of the class, before any method declarations, constructors, initializers or inner classes.

คำอธิบาย:

ตามมาตรฐาน Java ตัวแปรฟิลด์ (field) ของคลาสควรถูกประกาศไว้ ด้านบนสุดของคลาส ก่อนเมธอดและคอนสตรัคเตอร์ เพื่อให้อ่านและบำรุงรักษาได้ง่าย แต่ในโค้ดนี้มีการประกาศ field ไว้หลังเมธอด จึงไม่เป็นไปตามมาตรฐาน

Q5: จาก Q4 หากต้องการแก้ไข Source code เพื่อให้ผ่านมาตรฐานการเขียนโค้ดและการตรวจสอบโดย Checkstyle จะต้องแก้ไข Code ในบรรทัดนั้น ๆ อย่างไร

ตอบ

ปัญหา: Class comments are required

เพิ่ม Javadoc comment ด้านบนคลาส

```
/**
 * คลาสนี้ใช้สำหรับจัดการข้อมูลของผู้ใช้ในระบบ Whiteboard
 */
public class UserWhiteboard {
    ...
}
```

ชื่อ-นามสกุล แก้วเพชรรัตน์ สีสันต์ รหัสสนศ. 663380029-9 Section 1

ปัญหา: Usage of System.out/err

เปลี่ยนจาก

```
System.out.println("xxxxxxxxxxxxx");
```

เป็น

```
Logger logger = Logger.getLogger(Whiteboard.class.getName());
```

```
logger.info("xxxxxxxxx");
```

ปัญหา: Fields should be declared at the top of the class, before any method declarations, constructors, initializers or inner classes.

แนวทางแก้ไข:

ย้าย field ขึ้นไปไว้ด้านบน เช่น

```
public class WhiteboardManager {
```

```
    private String userId; // ควรอยู่ตรงนี้
```

```
    public WhiteboardManager(String userId) {
```

```
        this.userId = userId;
```

```
    }
```

```
    // methods...
```

```
}
```