ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

SOFTWARE ENGINEERING II
WORKSHOP #2 – CODING INSPECTION

TEAM #2

Fulco Pincay

October 2024

October 28, 2024

Part 1: Peer Review

The Shopping Cart module was reviewed with a focus on code quality, adherence to standards, and functionality. The overall score for this module is **85/100**. Below is a detailed analysis and feedback for improvement:

1. **Code Formatting and Structure (Score: 90/100)**: - The code is formatted consistently, mostly adhering to PEP 8 standards. Minor adjustments, such as adding line breaks in functions like `apply_discounts`, could enhance readability. - Separating discount calculations into dedicated methods could improve modularity and clarity.

2. **Naming Conventions (Score: 85/100)**: - Variable names are clear and follow standard conventions. However, the method `apply_discounts` could be renamed to a more descriptive name, such as `apply_cart_discounts`, for clarity.

3. **Commenting (Score: 75/100)**: - Docstrings are generally present, but more detailed explanations would be beneficial in complex sections, like tax and discount calculations. Adding inline comments in areas with intricate logic would improve readability.

4. **Logic and Functionality (Score: 80/100)**: - The code functions as expected; however, there is redundancy in the application of the coupon discount within `calculate_total`. Refining the discount logic to avoid double

application would increase accuracy.

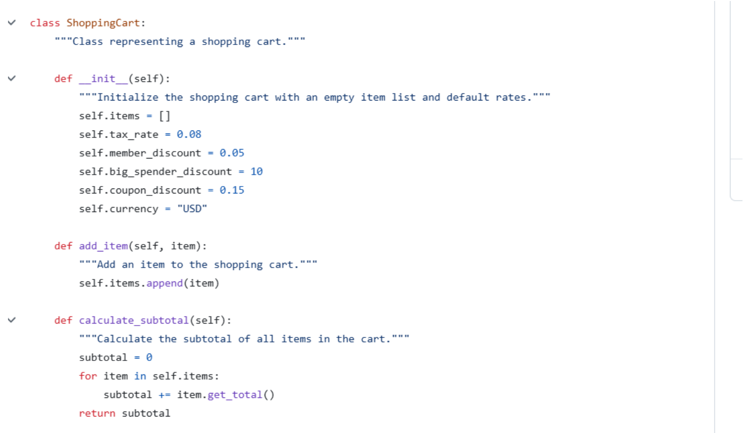
5. **Error Handling (Score: 70/100)**: - Limited error handling is present, with no validation for negative prices or quantities. Adding error handling for such cases would make the code more robust.

6. **Performance (Score: 80/100)**: - The code performs efficiently, though there is an opportunity to optimize calculations related to environmental fees by handling them outside the main loop.

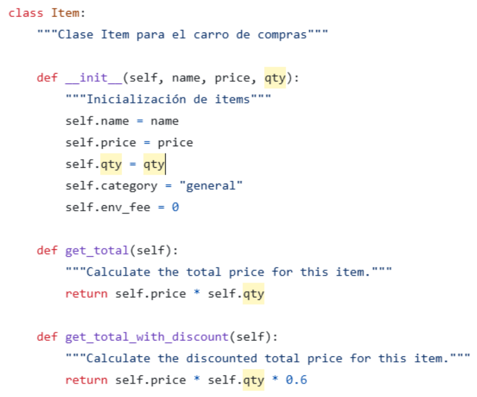
7. **Security (Score: 85/100)**: - Although no sensitive data is processed, adding basic validation for inputs (e.g., prices and quantities) would enhance security by preventing potential misuse.

Overall Score: 85/100

Final Recommendations: - Add validation in the `Item` class to handle negative inputs for price and quantity. - Refactor `apply_discounts` and `calculate_total` to prevent redundant discount applications. - Improve inline documentation for better readability, particularly in sections with complex calculations.



```python
class ShoppingCart:
    """Class representing a shopping cart."""

    def __init__(self):
        """Initialize the shopping cart with an empty item list and default rates."""
        self.items = []
        self.tax_rate = 0.08
        self.member_discount = 0.05
        self.big_spender_discount = 10
        self.coupon_discount = 0.15
        self.currency = "USD"

    def add_item(self, item):
        """Add an item to the shopping cart."""
        self.items.append(item)

    def calculate_subtotal(self):
        """Calculate the subtotal of all items in the cart."""
        subtotal = 0
        for item in self.items:
            subtotal += item.get_total()
        return subtotal
```



```python
class Item:
    """Clase Item para el carro de compras"""

    def __init__(self, name, price, qty):
        """Inicialización de items"""
        self.name = name
        self.price = price
        self.qty = qty
        self.category = "general"
        self.env_fee = 0

    def get_total(self):
        """Calculate the total price for this item."""
        return self.price * self.qty

    def get_total_with_discount(self):
        """Calculate the discounted total price for this item."""
        return self.price * self.qty * 0.6
```

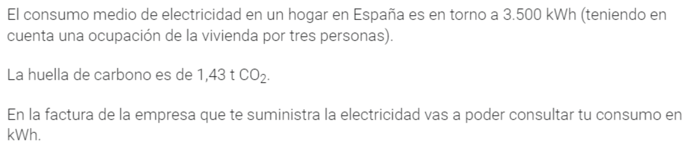
Lab Report

1 Part 2: Install Maven for Eclipse

For this workshop, all members of the team had the Maven plugin already installed. So, we skipped this part.

2 Part 3: Install the Eclipse PMD Plugin

First, we searched for the PMD plugin in Eclipse's Marketplace. The following window appeared. We chose the "pmd-eclipse-plugin 7.0.0" option, shared by PMD.



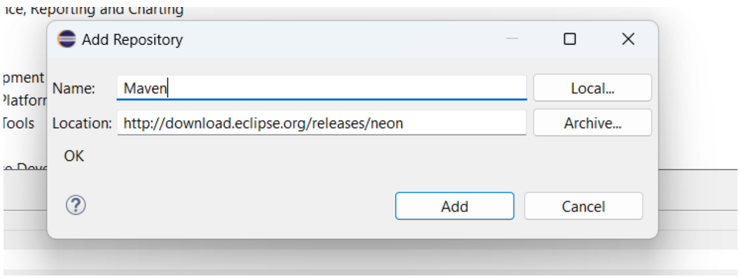
El consumo medio de electricidad en un hogar en España es en torno a 3.500 kWh (teniendo en cuenta una ocupación de la vivienda por tres personas).

La huella de carbono es de 1,43 t CO_2.

En la factura de la empresa que te suministra la electricidad vas a poder consultar tu consumo en kWh.

3 Part 4: Download and Configure the Project

After installing the plugin, we cloned the repo and configured the PMD properties of the project. Notice that the option "Enable PMD" in the PMD section is selected. We kept the set of rules provided by default and we applied the changes.

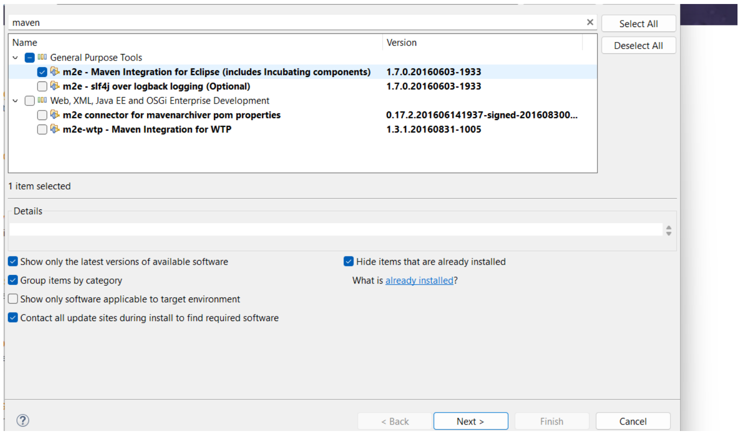


Name: Maven

Location: http://download.eclipse.org/releases/neon

4 Part 5: Using PMD

It's time to use PMD. Before checking the project, we selected the "Check code after saving" option in the Plugin Options, to get the latest advice after saving the code.



maven

Name	Version
▣ General Purpose Tools	
☑ m2e - Maven Integration for Eclipse (includes Incubating components)	1.7.0.20160603-1933
☐ m2e - slf4j over logback logging (Optional)	1.7.0.20160603-1933
☐ Web, XML, Java EE and OSGi Enterprise Development	
☐ m2e connector for mavenarchiver pom properties	0.17.2.201606141937-signed-201608300...
☐ m2e-wtp - Maven Integration for WTP	1.3.1.20160831-1005

1 item selected

Details

☑ Show only the latest versions of available software

☑ Group items by category

☐ Show only software applicable to target environment

☑ Contact all update sites during install to find required software

☑ Hide items that are already installed

What is already installed?

We checked the code of the repo by clicking in the project folder and selecting "PMD – Check code". After that, the following content appeared. It's important to note that most of the violations displayed are Urgent (medium priority).

If we right-click on any violation, there is a view with the description of the problem and alternative solutions.

5 Part 6: Creating PMD Rule Sets

The next step is to create the PMD rule sets. For this workshop, all members of the team had the Eclipse XML Editors and Tools plugin already installed. So, we skipped this part. Then, we created the XML file with the name: "g2_ruleset.xml".

We edited the file in the Source tab. We added a new set of rules:

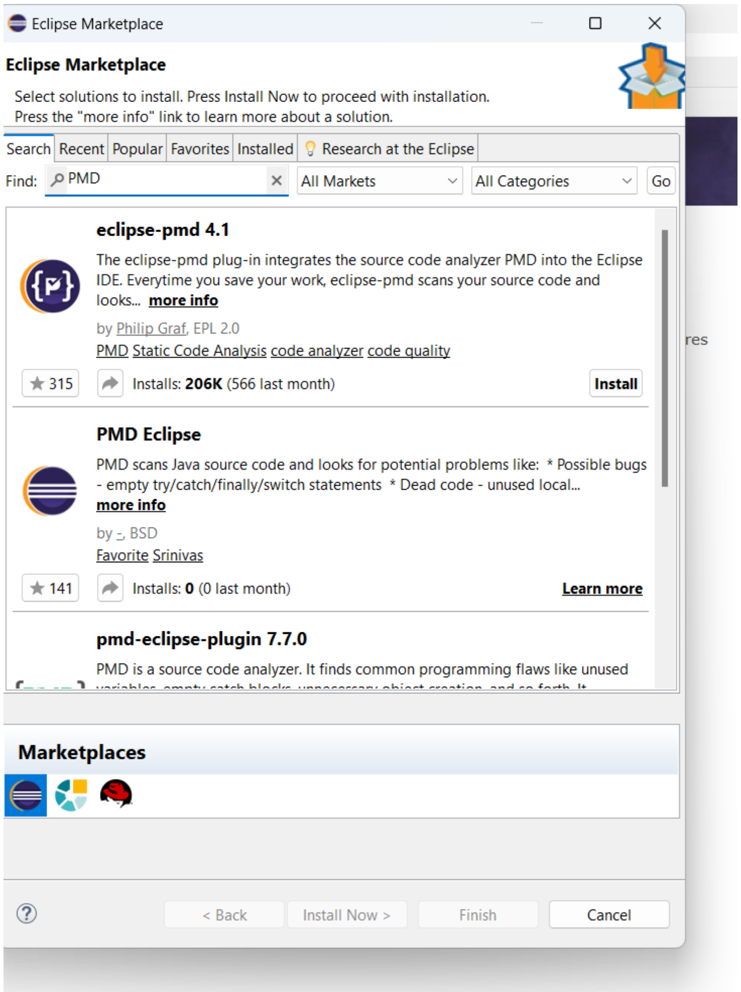
- Performance
- Best Practices
- Immutable Field
- Use Utility Class

The "Immutable Field" and "Utility Class" rules have a priority number of 1, the highest priority possible.

In case we want to use an external ruleset, we need to go to the PMD Configuration in Window – Preferences. There, we checked the "Use global configuration" option. Notice that there are no rules active.

We clicked the "Import rules" button and the following window appeared. We upload the ruleset created by clicking the Browse button. Then we clicked the Ok button.

Finally, we apply and close all the changes.



Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.
Press the "more info" link to learn more about a solution.

Find: PMD

eclipse-pmd 4.1

The eclipse-pmd plug-in integrates the source code analyzer PMD into the Eclipse IDE. Everytime you save your work, eclipse-pmd scans your source code and looks... **more info**

by Philip Graf, EPL 2.0

PMD Static Code Analysis code analyzer code quality

★ 315 Installs: **206K** (566 last month) **Install**

PMD Eclipse

PMD scans Java source code and looks for potential problems like: * Possible bugs - empty try/catch/finally/switch statements * Dead code - unused local... **more info**

by -, BSD

Favorite Srinivas

★ 141 Installs: **0** (0 last month) **Learn more**

pmd-eclipse-plugin 7.7.0

PMD is a source code analyzer. It finds common programming flaws like unused

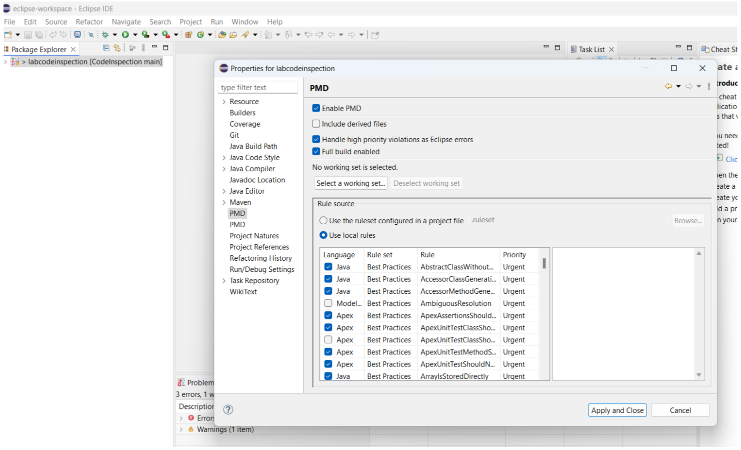
6 Part 7: Generating a PMD Report

To generate a PMD report, we opened the Preferences window and in the PMD tab, we selected the Reports category. There, we checked the HTML and text options. We applied all the changes.

After doing so, a folder named “Reports” was created in the folder’s project with our reports in HTML and TXT format.

7 Part 8: Suppressing PMD Rules

In this case, we must not respect a rule. To omit it, we add a comment `NOPMD`, and then the warning will not appear in the “Violations Overview” tab.



PMD

☑ Enable PMD

□ Include derived files

☑ Handle high priority violations as Eclipse errors

☑ Full build enabled

No working set is selected.

Select a working set... Deselect working set

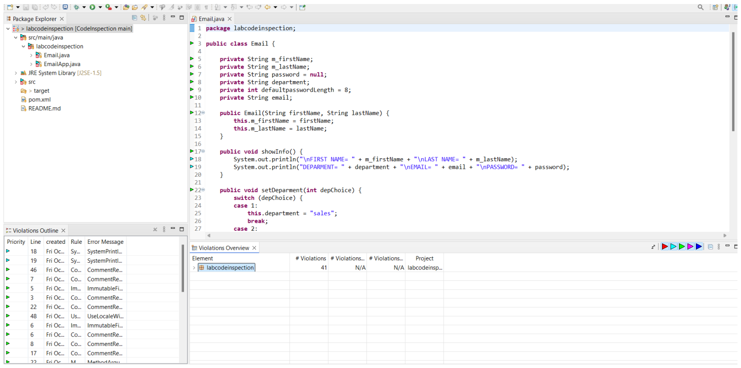
Rule source

○ Use the ruleset configured in a project file .ruleset Browse...

● Use local rules

Language	Rule set	Rule	Priority
☑ Java	Best Practices	AbstractClassWithout...	Urgent
☑ Java	Best Practices	AccessorClassGenerati...	Urgent
☑ Java	Best Practices	AccessorMethodGene...	Urgent
□ Model...	Best Practices	AmbiguousResolution	Urgent
☑ Apex	Best Practices	ApexAssertionsShould...	Urgent
☑ Apex	Best Practices	ApexUnitTestClassSho...	Urgent
□ Apex	Best Practices	ApexUnitTestClassSho...	Urgent
☑ Apex	Best Practices	ApexUnitTestMethodS...	Urgent
☑ Apex	Best Practices	ApexUnitTestShouldN...	Urgent
☑ Java	Best Practices	ArrayIsStoredDirectly	Urgent

Problems
3 errors, 1 w
Errors
Warnings (1 item)



```java
package labcodeinspection;

public class Email {

    private String m_firstName;
    private String m_lastName;
    private String password = null;
    private String department;
    private int defaultpasswordLength = 8;
    private String email;

    public Email(String firstName, String lastName) {
        this.m_firstName = firstName;
        this.m_lastName = lastName;
    }

    public void showInfo() {
        System.out.println("\nFIRST NAME= " + m_firstName + "\nLAST NAME= " + m_lastName);
        System.out.println("DEPARMENT= " + department + "\nEMAIL= " + email + "\nPASSWORD= " + password);
    }

    public void setDeparment(int depChoice) {
        switch (depChoice) {
        case 1:
            this.department = "sales";
            break;
        case 2:
```

Violations Outline

Priority	Line	created	Rule	Error Message
▷	18	Fri Oc...	Sy...	SystemPrintl...
▷	19	Fri Oc...	Sy...	SystemPrintl...
▷	46	Fri Oc...	Co...	CommentRe...
▷	7	Fri Oc...	Co...	CommentRe...
▷	5	Fri Oc...	Im...	ImmutableFi...
▷	3	Fri Oc...	Co...	CommentRe...
▷	22	Fri Oc...	Co...	CommentRe...
▷	48	Fri Oc...	Us...	UseLocaleWi...
▷	6	Fri Oc...	Im...	ImmutableFi...
▷	6	Fri Oc...	Co...	CommentRe...
▷	8	Fri Oc...	Co...	CommentRe...
▷	17	Fri Oc...	Co...	CommentRe...
▷	22	Fri Oc...	M...	MethodArgu...

Violations Overview

Element	# Violations	# Violations...	# Violations...	Project
labcodeinspection	41	N/A	N/A	labcodeinsp...

After saving one file, we can verify that the violation is not anymore on the tab.

8 Part 9: Detecting Cut-and-Paste with CPD

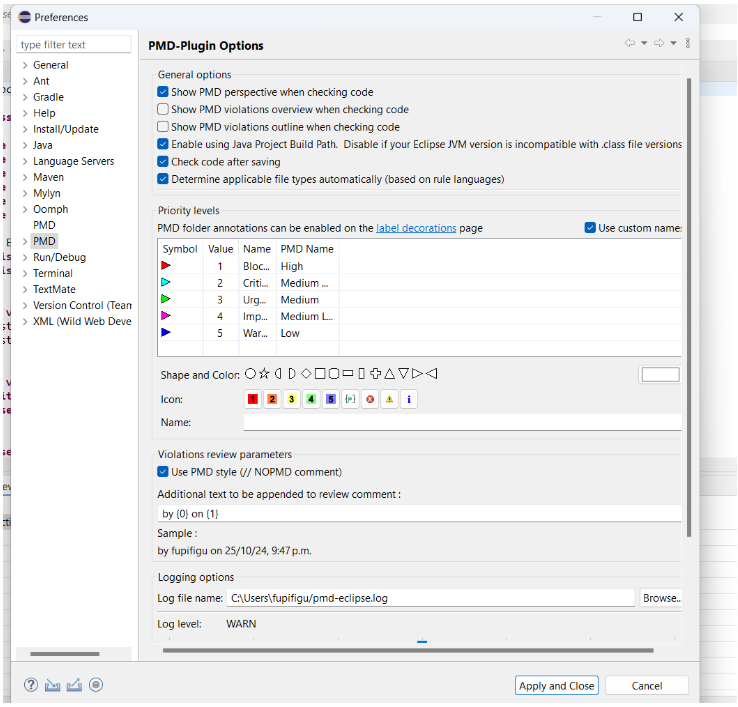
Just to make a test, we copied the method `randomPassword` from `Email.java` to `EmailApp.java`, then clicked on "Find Suspect Cut & Paste" and configured the necessary inputs.

In the /report directory, a file was generated with a clear description of the duplicated code.

9 Part 10: PMD and Maven

Following similar steps, we edited the `pom.xml` file and added the rule set from our previously created XML file.

Finally, we accessed our Lab Code Inspection with the PMD results.



PMD-Plugin Options

General options

☑ Show PMD perspective when checking code
☐ Show PMD violations overview when checking code
☐ Show PMD violations outline when checking code
☑ Enable using Java Project Build Path. Disable if your Eclipse JVM version is incompatible with .class file versions
☑ Check code after saving
☑ Determine applicable file types automatically (based on rule languages)

Priority levels

PMD folder annotations can be enabled on the label decorations page ☑ Use custom name:

Symbol	Value	Name	PMD Name
▶	1	Bloc...	High
▷	2	Criti...	Medium ...
▶	3	Urg...	Medium
▶	4	Imp...	Medium L...
▶	5	War...	Low

Shape and Color: ○ ☆ ◖ ◗ ◇ □ ▢ ▭ ▯ ✚ △ ▽ ▷ ◁

Icon: 1 2 3 4 5 {P} ⊗ ⚠ i

Name:

Violations review parameters

☑ Use PMD style (// NOPMD comment)

Additional text to be appended to review comment :

by {0} on {1}

Sample :

by fupifigu on 25/10/24, 9:47 p.m.

Logging options

Log file name: C:\Users\fupifigu/pmd-eclipse.log Browse..

Log level: WARN

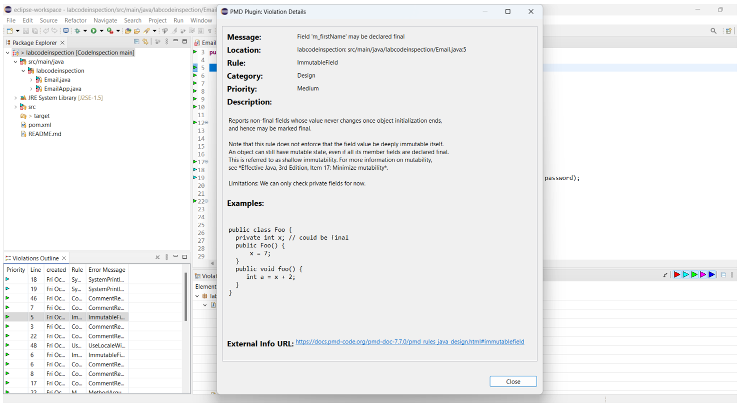
Challenge

Following the previous steps, we generated the PMD report.

Repository

The project code is available on GitHub:

- `https://github.com/Fulcopin/taller_2.1.git` - Main branch: Laboratory code
- Coding Standards - II TERM 2024 branch: Challenge code



PMD Plugin: Violation Details

Message: Field 'm_firstName' may be declared final

Location: labcodeinspection: src/main/java/labcodeinspection/Email.java:5

Rule: ImmutableField

Category: Design

Priority: Medium

Description:

Reports non-final fields whose value never changes once object initialization ends,
and hence may be marked final.

Note that this rule does not enforce that the field value be deeply immutable itself.
An object can still have mutable state, even if all its member fields are declared final.
This is referred to as shallow immutability. For more information on mutability,
see *Effective Java, 3rd Edition, Item 17: Minimize mutability*.

Limitations: We can only check private fields for now.

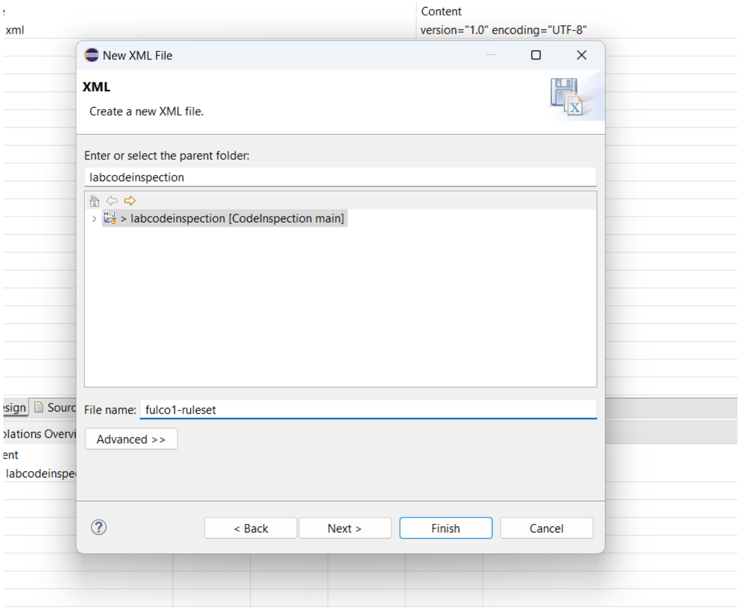
Examples:

```
public class Foo {
  private int x; // could be final
  public Foo() {
      x = 7;
  }
  public void foo() {
     int a = x + 2;
  }
}
```

External Info URL: https://docs.pmd-code.org/pmd-doc-7.7.0/pmd_rules_java_design.html#immutablefield

Violations Outline

Priority	Line	created	Rule	Error Message
▶	18	Fri Oc...	Sy...	SystemPrintl...
▶	19	Fri Oc...	Sy...	SystemPrintl...
▶	46	Fri Oc...	Co...	CommentRe...
▶	7	Fri Oc...	Co...	CommentRe...
▶	5	Fri Oc...	Im...	ImmutableFi...
▶	3	Fri Oc...	Co...	CommentRe...
▶	22	Fri Oc...	Co...	CommentRe...
▶	48	Fri Oc...	Us...	UseLocaleWi...
▶	6	Fri Oc...	Im...	ImmutableFi...
▶	6	Fri Oc...	Co...	CommentRe...
▶	8	Fri Oc...	Co...	CommentRe...
▶	17	Fri Oc...	Co...	CommentRe...
▶	22	Fri Oc...	M...	MethodArgu...



New XML File

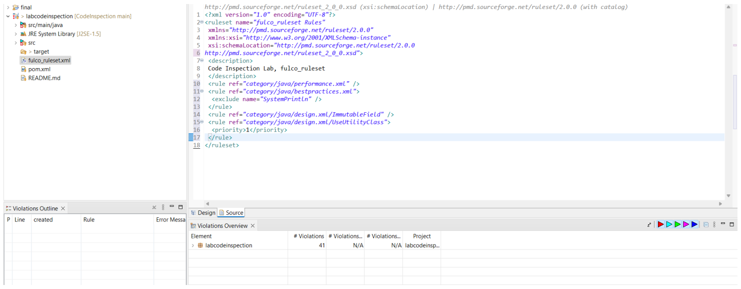
XML

Create a new XML file.

Enter or select the parent folder: labcodeinspection

> labcodeinspection [CodeInspection main]

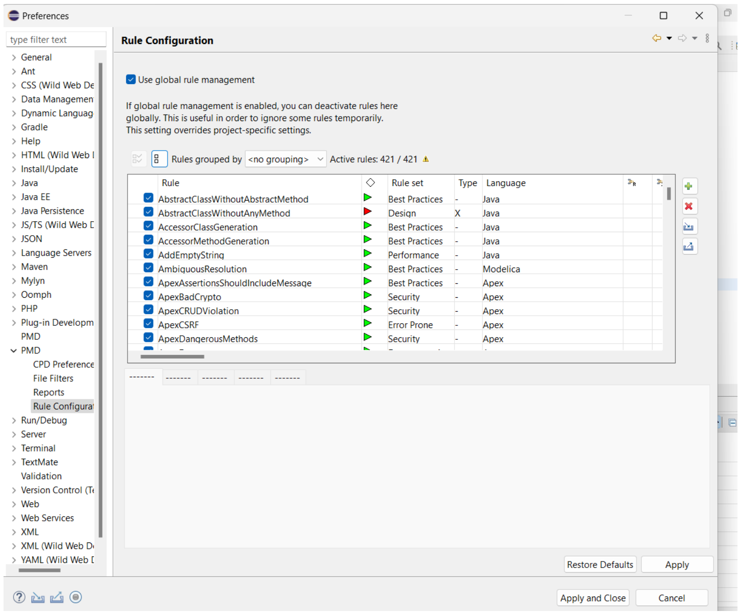
File name: fulco1-ruleset



```
http://pmd.sourceforge.net/ruleset_2_0_0.xsd (xsi:schemaLocation) | http://pmd.sourceforge.net/ruleset/2.0.0 (with catalog)
<?xml version="1.0" encoding="UTF-8"?>
<ruleset name="fulco_ruleset Rules"
 xmlns="http://pmd.sourceforge.net/ruleset/2.0.0"
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xsi:schemaLocation="http://pmd.sourceforge.net/ruleset/2.0.0
http://pmd.sourceforge.net/ruleset_2_0_0.xsd">
 <description>
 Code Inspection Lab, fulco_ruleset
 </description>
 <rule ref="category/java/performance.xml" />
 <rule ref="category/java/bestpractices.xml">
  <exclude name="SystemPrintln" />
 </rule>
 <rule ref="category/java/design.xml/ImmutableField" />
 <rule ref="category/java/design.xml/UseUtilityClass">
  <priority>1</priority>
 </rule>
</ruleset>
```

Violations Overview

Element	# Violations	# Violations...	# Violations...	Project
labcodeinspection	41	N/A	N/A	labcodeinsp...



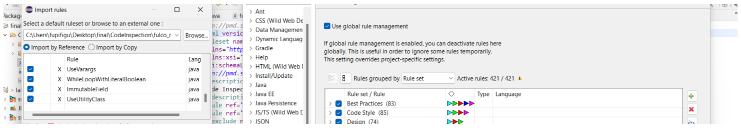
Rule Configuration

☑ Use global rule management

If global rule management is enabled, you can deactivate rules here
globally. This is useful in order to ignore some rules temporarily.
This setting overrides project-specific settings.

Rules grouped by <no grouping> Active rules: 421 / 421 ⚠

	Rule	◇	Rule set	Type	Language
☑	AbstractClassWithoutAbstractMethod	▶	Best Practices	-	Java
☑	AbstractClassWithoutAnyMethod	▶	Design	X	Java
☑	AccessorClassGeneration	▶	Best Practices	-	Java
☑	AccessorMethodGeneration	▶	Best Practices	-	Java
☑	AddEmptyString	▶	Performance	-	Java
☑	AmbiguousResolution	▶	Best Practices	-	Modelica
☑	ApexAssertionsShouldIncludeMessage	▶	Best Practices	-	Apex
☑	ApexBadCrypto	▶	Security	-	Apex
☑	ApexCRUDViolation	▶	Security	-	Apex
☑	ApexCSRF	▶	Error Prone	-	Apex
☑	ApexDangerousMethods	▶	Security	-	Apex



Select a default ruleset or browse to an external one :

C:\Users\fupifigu\Desktop\final\CodeInspection\fulco_r

● Import by Reference ○ Import by Copy

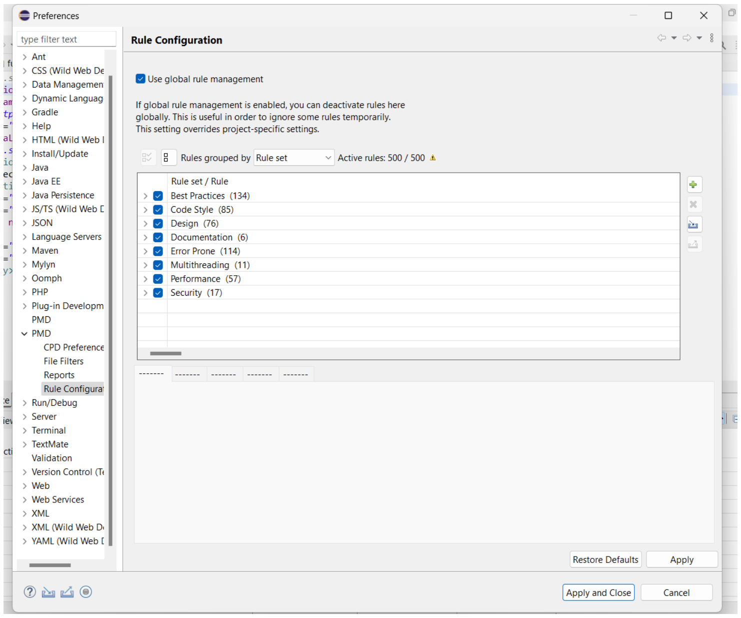
		Rule	Lang
☑	X	UseVarargs	java
☑	X	WhileLoopWithLiteralBoolean	java
☑	X	ImmutableField	java
☑	X	UseUtilityClass	java

☑ Use global rule management

If global rule management is enabled, you can deactivate rules here
globally. This is useful in order to ignore some rules temporarily.
This setting overrides project-specific settings.

Rules grouped by Rule set Active rules: 421 / 421 ⚠

	Rule set / Rule	◇	Type	Language
☑	Best Practices (83)	▶▶▶▶▶		
☑	Code Style (85)	▶▶▶▶		
☑	Design (74)	▶▶▶		



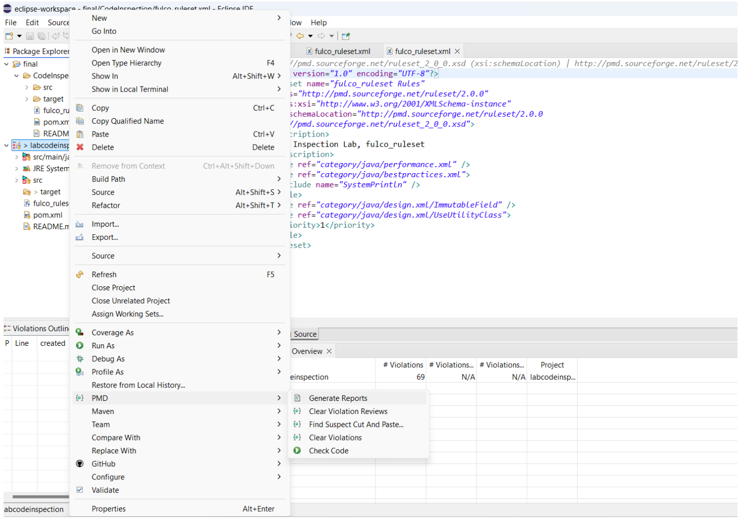
Rule Configuration

☑ Use global rule management

If global rule management is enabled, you can deactivate rules here
globally. This is useful in order to ignore some rules temporarily.
This setting overrides project-specific settings.

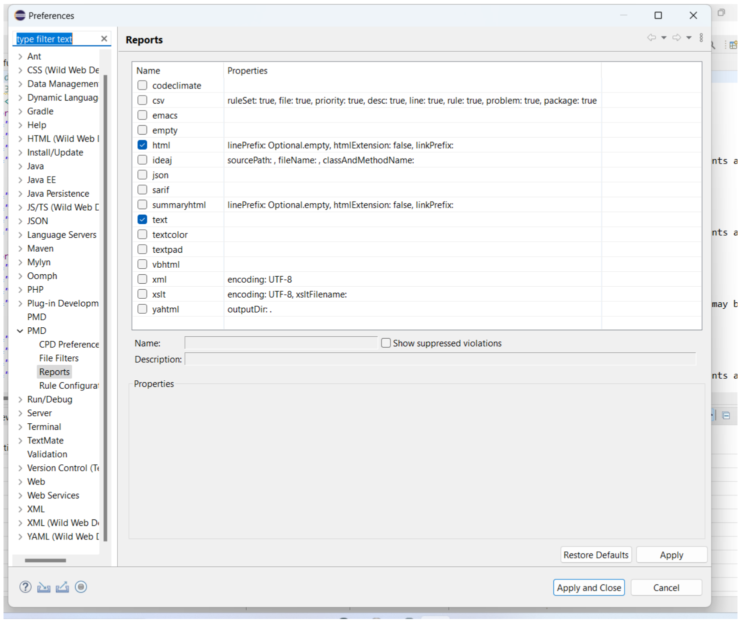
Rules grouped by Rule set Active rules: 500 / 500 ⚠

	Rule set / Rule
☑	Best Practices (134)
☑	Code Style (85)
☑	Design (76)
☑	Documentation (6)
☑	Error Prone (114)
☑	Multithreading (11)
☑	Performance (57)
☑	Security (17)



```
//pmd.sourceforge.net/ruleset_2_0_0.xsd (xsi:schemaLocation) | http://pmd.sourceforge.net/ruleset/2
 version="1.0" encoding="UTF-8"?>
set name="fulco_ruleset Rules"
s="http://pmd.sourceforge.net/ruleset/2.0.0"
s:xsi="http://www.w3.org/2001/XMLSchema-instance"
schemaLocation="http://pmd.sourceforge.net/ruleset/2.0.0
//pmd.sourceforge.net/ruleset_2_0_0.xsd">
cription>
 Inspection Lab, fulco_ruleset
scription>
e ref="category/java/performance.xml" />
e ref="category/java/bestpractices.xml">
clude name="SystemPrintln" />
le>
e ref="category/java/design.xml/ImmutableField" />
e ref="category/java/design.xml/UseUtilityClass">
iority>1</priority>
le>
eset>
```

	# Violations	# Violations...	# Violations...	Project
einspection	69	N/A	N/A	labcodeinsp...

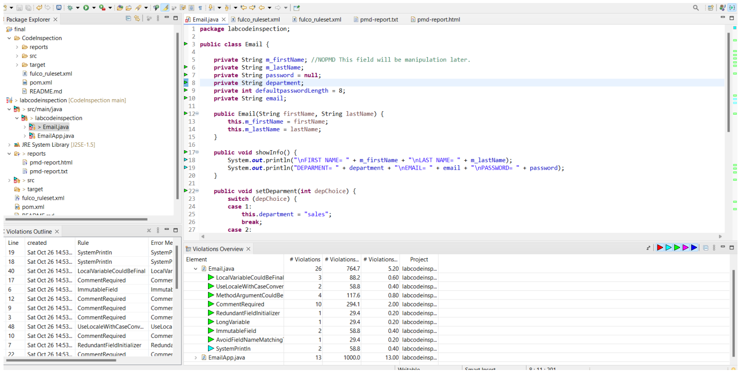


Name	Properties
☐ codeclimate	
☐ csv	ruleSet: true, file: true, priority: true, desc: true, line: true, rule: true, problem: true, package: true
☐ emacs	
☐ empty	
☑ html	linePrefix: Optional.empty, htmlExtension: false, linkPrefix:
☐ ideaj	sourcePath: , fileName: , classAndMethodName:
☐ json	
☐ sarif	
☐ summaryhtml	linePrefix: Optional.empty, htmlExtension: false, linkPrefix:
☑ text	
☐ textcolor	
☐ textpad	
☐ vbhtml	
☐ xml	encoding: UTF-8
☐ xslt	encoding: UTF-8, xsltFilename:
☐ yahtml	outputDir: .

Name: ☐ Show suppressed violations

Description:

Properties



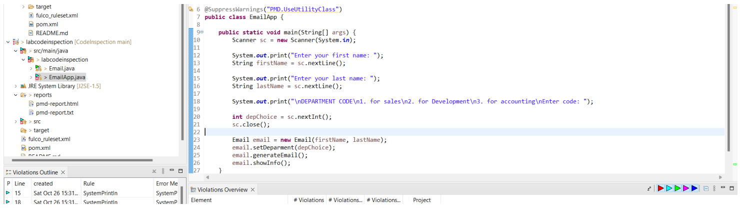
```java
package labcodeinspection;

public class Email {

    private String m_firstName; //NOPMD This field will be manipulation later.
    private String m_lastName;
    private String password = null;
    private String department;
    private int defaultpasswordLength = 8;
    private String email;

    public Email(String firstName, String lastName) {
        this.m_firstName = firstName;
        this.m_lastName = lastName;
    }

    public void showInfo() {
        System.out.println("\nFIRST NAME= " + m_firstName + "\nLAST NAME= " + m_lastName);
        System.out.println("DEPARMENT= " + department + "\nEMAIL= " + email + "\nPASSWORD= " + password);
    }

    public void setDeparment(int depChoice) {
        switch (depChoice) {
        case 1:
            this.department = "sales";
            break;
        case 2:
```

Violations Outline

Line	created	Rule	Error Me
19	Sat Oct 26 14:53...	SystemPrintln	SystemP
18	Sat Oct 26 14:53...	SystemPrintln	SystemP
40	Sat Oct 26 14:53...	LocalVariableCouldBeFinal	LocalVar
17	Sat Oct 26 14:53...	CommentRequired	Commer
6	Sat Oct 26 14:53...	ImmutableField	Immutat
12	Sat Oct 26 14:53...	CommentRequired	Commer
9	Sat Oct 26 14:53...	CommentRequired	Commer
3	Sat Oct 26 14:53...	CommentRequired	Commer
48	Sat Oct 26 14:53...	UseLocaleWithCaseConv...	UseLoca
10	Sat Oct 26 14:53...	CommentRequired	Commer
7	Sat Oct 26 14:53...	RedundantFieldInitializer	Redunda
22	Sat Oct 26 14:53...	CommentRequired	Commer

Violations Overview

Element	# Violations	# Violations...	# Violations...	Project
Email.java	26	764.7	5.20	labcodeinsp...
LocalVariableCouldBeFinal	3	88.2	0.60	labcodeinsp...
UseLocaleWithCaseConver	2	58.8	0.40	labcodeinsp...
MethodArgumentCouldBe	4	117.6	0.80	labcodeinsp...
CommentRequired	10	294.1	2.00	labcodeinsp...
RedundantFieldInitializer	1	29.4	0.20	labcodeinsp...
LongVariable	1	29.4	0.20	labcodeinsp...
ImmutableField	2	58.8	0.40	labcodeinsp...
AvoidFieldNameMatching	1	29.4	0.20	labcodeinsp...
SystemPrintln	2	58.8	0.40	labcodeinsp...
EmailApp.java	13	1000.0	13.00	labcodeinsp...



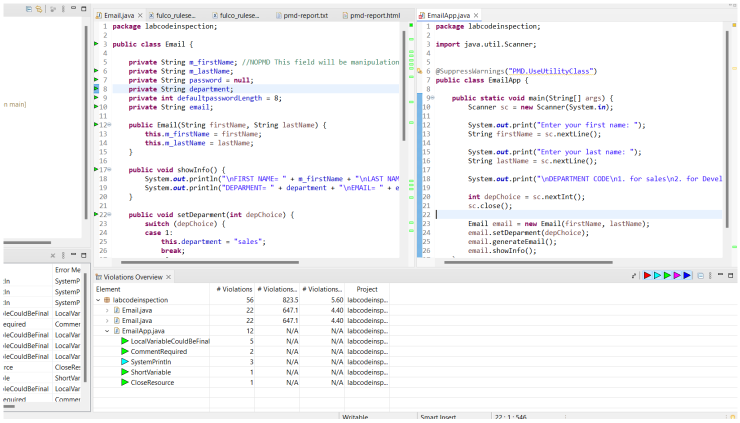
```java
@SuppressWarnings("PMD.UseUtilityClass")
public class EmailApp {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter your first name: ");
        String firstName = sc.nextLine();

        System.out.print("Enter your last name: ");
        String lastName = sc.nextLine();

        System.out.print("\nDEPARTMENT CODE\n1. for sales\n2. for Development\n3. for accounting\nEnter code: ");

        int depChoice = sc.nextInt();
        sc.close();

        Email email = new Email(firstName, lastName);
        email.setDeparment(depChoice);
        email.generateEmail();
        email.showInfo();
    }
```

Violations Outline

P	Line	created	Rule	Error Me
▷	15	Sat Oct 26 15:31...	SystemPrintln	SystemP
▷	18	Sat Oct 26 15:31...	SystemPrintln	SystemP

Violations Overview

Element	# Violations	# Violations...	# Violations...	Project



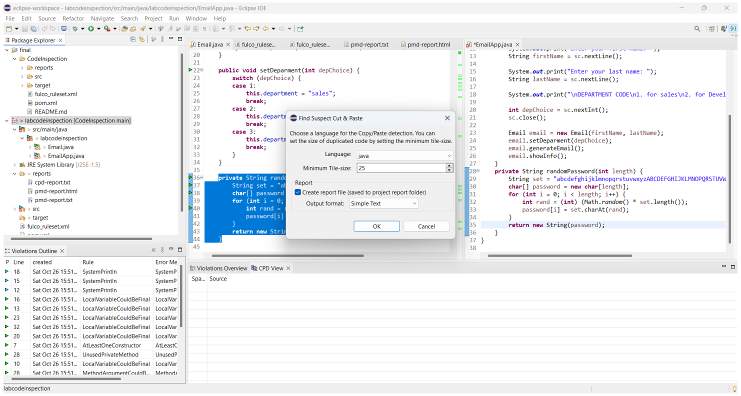
```java
package labcodeinspection;

public class Email {

    private String m_firstName; //NOPMD This field will be manipulation
    private String m_lastName;
    private String password = null;
    private String department;
    private int defaultpasswordLength = 8;
    private String email;

    public Email(String firstName, String lastName) {
        this.m_firstName = firstName;
        this.m_lastName = lastName;
    }

    public void showInfo() {
        System.out.println("\nFIRST NAME= " + m_firstName + "\nLAST NAM
        System.out.println("DEPARMENT= " + department + "\nEMAIL= " + e
    }

    public void setDeparment(int depChoice) {
        switch (depChoice) {
        case 1:
            this.department = "sales";
            break;
```

```java
package labcodeinspection;

import java.util.Scanner;


@SuppressWarnings("PMD.UseUtilityClass")
public class EmailApp {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter your first name: ");
        String firstName = sc.nextLine();

        System.out.print("Enter your last name: ");
        String lastName = sc.nextLine();

        System.out.print("\nDEPARTMENT CODE\n1. for sales\n2. for Devel

        int depChoice = sc.nextInt();
        sc.close();

        Email email = new Email(firstName, lastName);
        email.setDeparment(depChoice);
        email.generateEmail();
        email.showInfo();
```

	Error Me
tln	SystemP
tln	SystemP
tln	SystemP
leCouldBeFinal	LocalVar
equired	Commer
leCouldBeFinal	LocalVar
leCouldBeFinal	LocalVar
leCouldBeFinal	LocalVar
rce	CloseRe:
le	ShortVar
leCouldBeFinal	LocalVar
equired	Commer

Violations Overview

Element	# Violations	# Violations...	# Violations...	Project
labcodeinspection	56	823.5	5.60	labcodeinsp...
Email.java	22	647.1	4.40	labcodeinsp...
Email.java	22	647.1	4.40	labcodeinsp...
EmailApp.java	12	N/A	N/A	labcodeinsp...
LocalVariableCouldBeFinal	5	N/A	N/A	labcodeinsp...
CommentRequired	2	N/A	N/A	labcodeinsp...
SystemPrintln	3	N/A	N/A	labcodeinsp...
ShortVariable	1	N/A	N/A	labcodeinsp...
CloseResource	1	N/A	N/A	labcodeinsp...



```java
    }

    public void setDeparment(int depChoice) {
        switch (depChoice) {
        case 1:
            this.department = "sales";
            break;
        case 2:
            this.departm
            break;
        case 3:
            this.departm
            break;
        }
    }

    private String rando
        String set = "ab
        char[] password
        for (int i = 0;
            int rand =
            password[i]
        }
        return new Strin
    }
```

Find Suspect Cut & Paste

Choose a language for the Copy/Paste detection. You can set the size of duplicated code by setting the minimum tile-size.

Language: java

Minimum Tile-size: 25

Report

☑ Create report file (saved to project report folder)

Output format: Simple Text

```java
        String firstName = sc.nextLine();

        System.out.print("Enter your last name: ");
        String lastName = sc.nextLine();

        System.out.print("\nDEPARTMENT CODE\n1. for sales\n2. for Devel

        int depChoice = sc.nextInt();
        sc.close();

        Email email = new Email(firstName, lastName);
        email.setDeparment(depChoice);
        email.generateEmail();
        email.showInfo();
    }
    private String randomPassword(int length) {
        String set = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVW
        char[] password = new char[length];
        for (int i = 0; i < length; i++) {
            int rand = (int) (Math.random() * set.length());
            password[i] = set.charAt(rand);
        }
        return new String(password);
    }
}
```

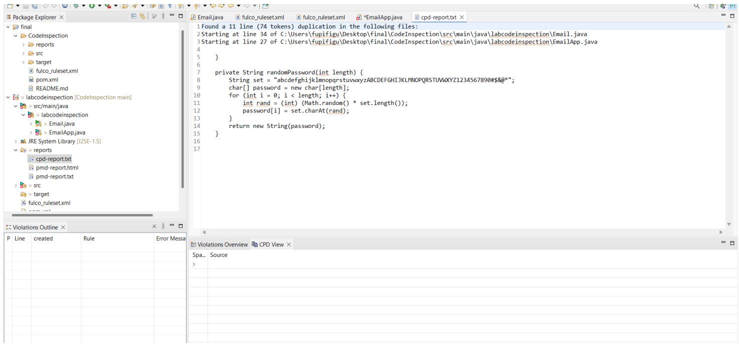
Violations Outline

P	Line	created	Rule	Error Me
▷	18	Sat Oct 26 15:51...	SystemPrintln	SystemP
▷	15	Sat Oct 26 15:51...	SystemPrintln	SystemP
▷	12	Sat Oct 26 15:51...	SystemPrintln	SystemP
▶	16	Sat Oct 26 15:51...	LocalVariableCouldBeFinal	LocalVar
▶	13	Sat Oct 26 15:51...	LocalVariableCouldBeFinal	LocalVar
▶	23	Sat Oct 26 15:51...	LocalVariableCouldBeFinal	LocalVar
▶	32	Sat Oct 26 15:51...	LocalVariableCouldBeFinal	LocalVar
▶	20	Sat Oct 26 15:51...	LocalVariableCouldBeFinal	LocalVar
▶	7	Sat Oct 26 15:51...	AtLeastOneConstructor	AtLeastC
▶	28	Sat Oct 26 15:51...	UnusedPrivateMethod	UnusedP
▶	10	Sat Oct 26 15:51...	LocalVariableCouldBeFinal	LocalVar
▶	28	Sat Oct 26 15:51...	MethodArgumentCouldB...	Method/

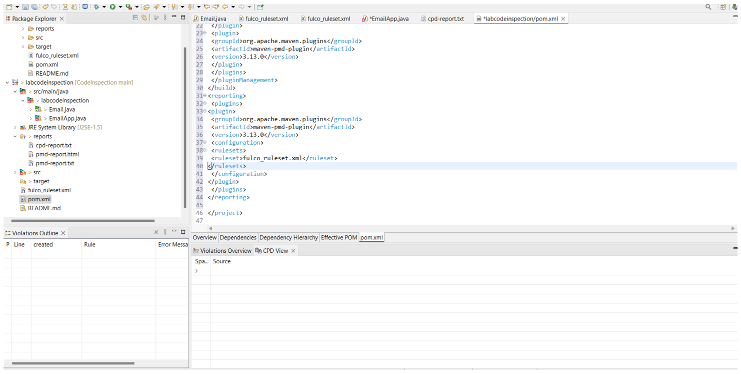
Violations Overview

CPD View

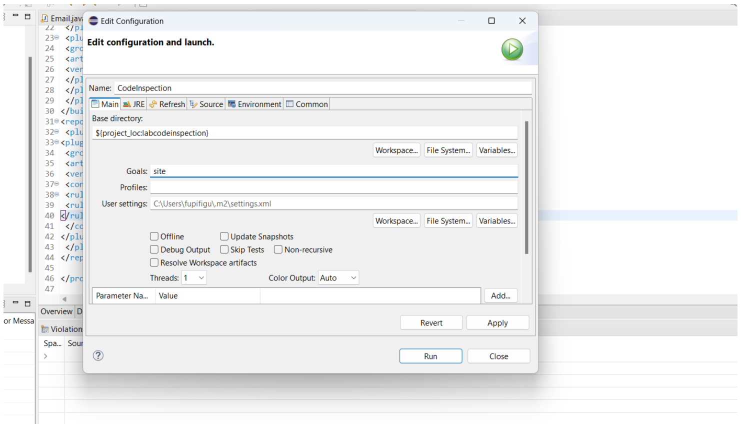
Spa...	Source



```
Found a 11 line (74 tokens) duplication in the following files:
Starting at line 34 of C:\Users\fupifigu\Desktop\final\CodeInspection\src\main\java\labcodeinspection\Email.java
Starting at line 27 of C:\Users\fupifigu\Desktop\final\CodeInspection\src\main\java\labcodeinspection\EmailApp.java

    }

    private String randomPassword(int length) {
        String set = "abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ1234567890#$&@*";
        char[] password = new char[length];
        for (int i = 0; i < length; i++) {
            int rand = (int) (Math.random() * set.length());
            password[i] = set.charAt(rand);
        }
        return new String(password);
    }
```



```
 </plugin>
 <plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-pmd-plugin</artifactId>
 <version>3.13.0</version>
 </plugin>
 </plugins>
 </pluginManagement>
</build>
<reporting>
 <plugins>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-pmd-plugin</artifactId>
 <version>3.13.0</version>
 <configuration>
 <rulesets>
 <ruleset>fulco_ruleset.xml</ruleset>
</rulesets>
 </configuration>
</plugin>
 </plugins>
</reporting>

</project>
```



Edit Configuration

Edit configuration and launch.

Name: CodeInspection

Base directory: ${project_loc:labcodeinspection}

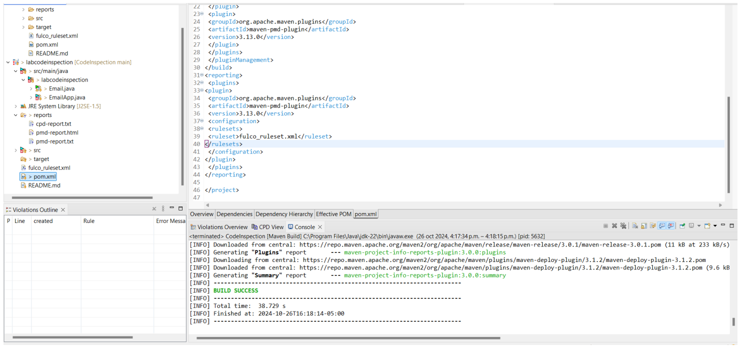
Goals: site

Profiles:

User settings: C:\Users\fupifigu\.m2\settings.xml

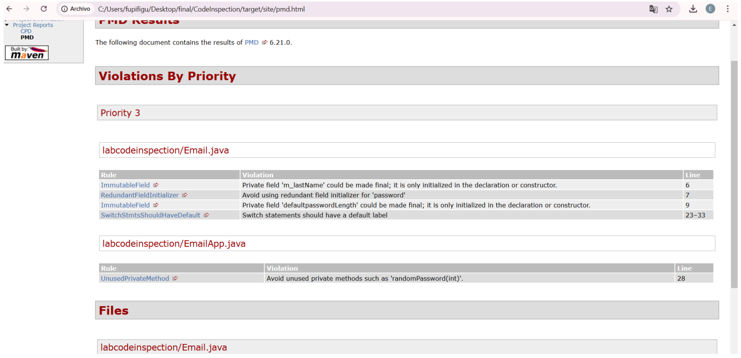
□ Offline □ Update Snapshots
□ Debug Output □ Skip Tests □ Non-recursive
□ Resolve Workspace artifacts

Threads: 1 Color Output: Auto



```
 </plugin>
 <plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-pmd-plugin</artifactId>
 <version>3.13.0</version>
 </plugin>
 </plugins>
 </pluginManagement>
</build>
<reporting>
 <plugins>
<plugin>
 <groupId>org.apache.maven.plugins</groupId>
 <artifactId>maven-pmd-plugin</artifactId>
 <version>3.13.0</version>
 <configuration>
 <rulesets>
 <ruleset>fulco_ruleset.xml</ruleset>
</rulesets>
 </configuration>
</plugin>
 </plugins>
</reporting>

</project>

```

```
<terminated> CodeInspection [Maven Build] C:\Program Files\Java\jdk-22\bin\javaw.exe  (26 oct 2024, 4:17:34 p.m. – 4:18:15 p.m.) [pid: 5632]
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/release/maven-release/3.0.1/maven-release-3.0.1.pom (11 kB at 233 kB/s)
[INFO] Generating "Plugins" report       --- maven-project-info-reports-plugin:3.0.0:plugins
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/3.1.2/maven-deploy-plugin-3.1.2.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/3.1.2/maven-deploy-plugin-3.1.2.pom (9.6 kB
[INFO] Generating "Summary" report       --- maven-project-info-reports-plugin:3.0.0:summary
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  38.729 s
[INFO] Finished at: 2024-10-26T16:18:14-05:00
[INFO] ------------------------------------------------------------------------
```


Archivo
C:/Users/fupifigu/Desktop/final/CodeInspection/target/site/pmd.html
Project Reports
CPD
PMD
Built by: maven

The following document contains the results of PMD 6.21.0.

Violations By Priority

Priority 3

labcodeinspection/Email.java

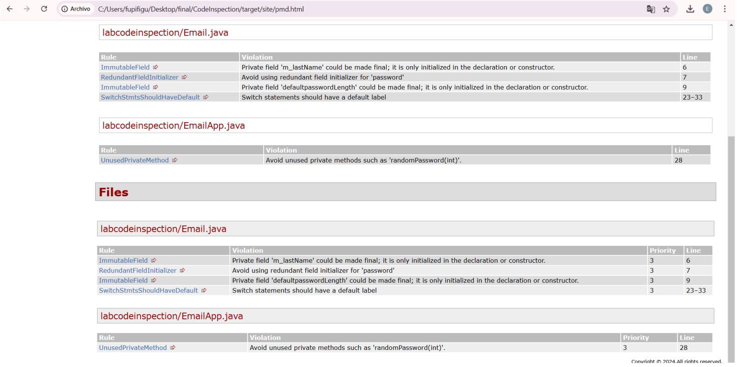
Rule	Violation	Line
ImmutableField	Private field 'm_lastName' could be made final; it is only initialized in the declaration or constructor.	6
RedundantFieldInitializer	Avoid using redundant field initializer for 'password'	7
ImmutableField	Private field 'defaultpasswordLength' could be made final; it is only initialized in the declaration or constructor.	9
SwitchStmtsShouldHaveDefault	Switch statements should have a default label	23–33

labcodeinspection/EmailApp.java

Rule	Violation	Line
UnusedPrivateMethod	Avoid unused private methods such as 'randomPassword(int)'.	28

Files

labcodeinspection/Email.java


Archivo
C:/Users/fupifigu/Desktop/final/CodeInspection/target/site/pmd.html

labcodeinspection/Email.java

Rule	Violation	Line
ImmutableField	Private field 'm_lastName' could be made final; it is only initialized in the declaration or constructor.	6
RedundantFieldInitializer	Avoid using redundant field initializer for 'password'	7
ImmutableField	Private field 'defaultpasswordLength' could be made final; it is only initialized in the declaration or constructor.	9
SwitchStmtsShouldHaveDefault	Switch statements should have a default label	23–33

labcodeinspection/EmailApp.java

Rule	Violation	Line
UnusedPrivateMethod	Avoid unused private methods such as 'randomPassword(int)'.	28

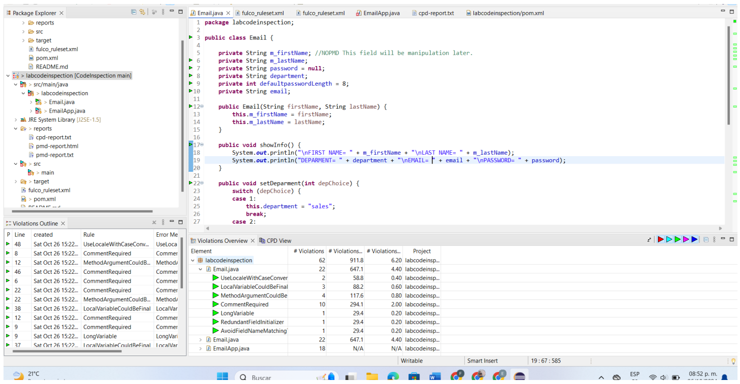
Files

labcodeinspection/Email.java

Rule	Violation	Priority	Line
ImmutableField	Private field 'm_lastName' could be made final; it is only initialized in the declaration or constructor.	3	6
RedundantFieldInitializer	Avoid using redundant field initializer for 'password'	3	7
ImmutableField	Private field 'defaultpasswordLength' could be made final; it is only initialized in the declaration or constructor.	3	9
SwitchStmtsShouldHaveDefault	Switch statements should have a default label	3	23–33

labcodeinspection/EmailApp.java

Rule	Violation	Priority	Line
UnusedPrivateMethod	Avoid unused private methods such as 'randomPassword(int)'.	3	28



```
> reports
> src
> target
  fulco_ruleset.xml
  pom.xml
  README.md
v > labcodeinspection [CodeInspection main]
  v > src/main/java
    v > labcodeinspection
      > > Email.java
      > > EmailApp.java
  > JRE System Library [J2SE-1.5]
  v > reports
      cpd-report.txt
      pmd-report.html
      pmd-report.txt
  v > src
      > main
  > > target
    fulco_ruleset.xml
    > pom.xml
```

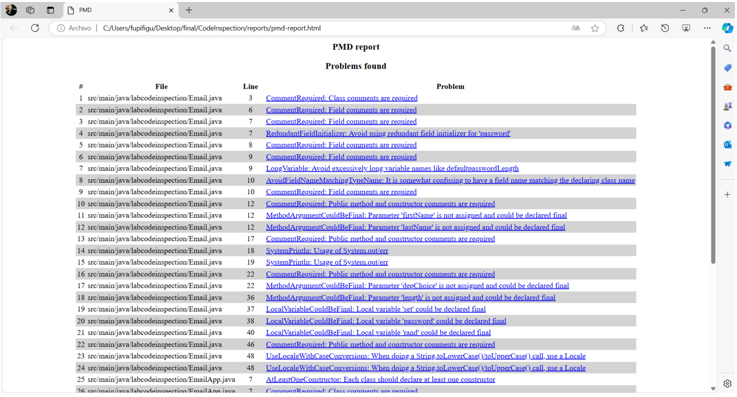
```java
package labcodeinspection;

public class Email {

    private String m_firstName; //NOPMD This field will be manipulation later.
    private String m_lastName;
    private String password = null;
    private String department;
    private int defaultpasswordLength = 8;
    private String email;

    public Email(String firstName, String lastName) {
        this.m_firstName = firstName;
        this.m_lastName = lastName;
    }

    public void showInfo() {
        System.out.println("\nFIRST NAME= " + m_firstName + "\nLAST NAME= " + m_lastName);
        System.out.println("DEPARMENT= " + department + "\nEMAIL= " + email + "\nPASSWORD= " + password);
    }

    public void setDeparment(int depChoice) {
        switch (depChoice) {
        case 1:
            this.department = "sales";
            break;
        case 2:
```

Violations Outline

P	Line	created	Rule	Error Me
▶	48	Sat Oct 26 15:22...	UseLocaleWithCaseConv...	UseLoca
▶	8	Sat Oct 26 15:22...	CommentRequired	Commer
▶	12	Sat Oct 26 15:22...	MethodArgumentCouldB...	Method/
▶	46	Sat Oct 26 15:22...	CommentRequired	Commer
▶	6	Sat Oct 26 15:22...	CommentRequired	Commer
▶	22	Sat Oct 26 15:22...	CommentRequired	Commer
▶	22	Sat Oct 26 15:22...	MethodArgumentCouldB...	Method/
▶	38	Sat Oct 26 15:22...	LocalVariableCouldBeFinal	LocalVar
▶	12	Sat Oct 26 15:22...	CommentRequired	Commer
▶	9	Sat Oct 26 15:22...	CommentRequired	Commer
▶	9	Sat Oct 26 15:22...	LongVariable	LongVari
▶	37	Sat Oct 26 15:22...	LocalVariableCouldBeFinal	LocalVar

Violations Overview | CPD View

Element	# Violations	# Violations...	# Violations...	Project
labcodeinspection	62	911.8	6.20	labcodeinsp...
Email.java	22	647.1	4.40	labcodeinsp...
UseLocaleWithCaseConver	2	58.8	0.40	labcodeinsp...
LocalVariableCouldBeFinal	3	88.2	0.60	labcodeinsp...
MethodArgumentCouldBe	4	117.6	0.80	labcodeinsp...
CommentRequired	10	294.1	2.00	labcodeinsp...
LongVariable	1	29.4	0.20	labcodeinsp...
RedundantFieldInitializer	1	29.4	0.20	labcodeinsp...
AvoidFieldNameMatching	1	29.4	0.20	labcodeinsp...
Email.java	22	647.1	4.40	labcodeinsp...
EmailApp.java	18	N/A	N/A	labcodeinsp...

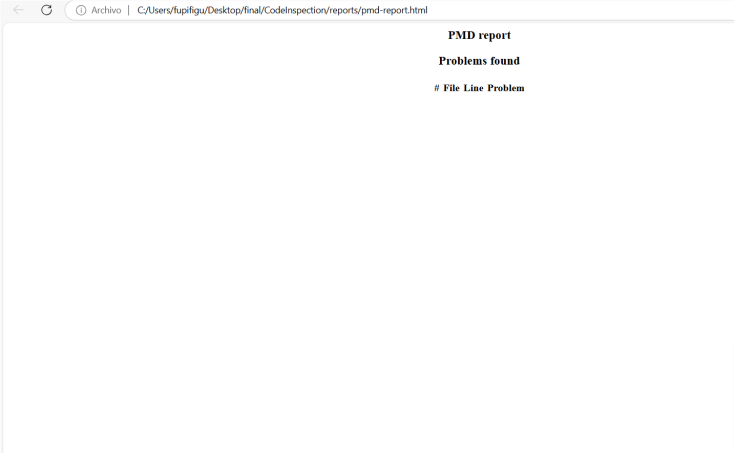


Archivo | C:/Users/fupifigu/Desktop/final/CodeInspection/reports/pmd-report.html

PMD report

Problems found

#	File	Line	Problem
1	src/main/java/labcodeinspection/Email.java	3	CommentRequired: Class comments are required
2	src/main/java/labcodeinspection/Email.java	6	CommentRequired: Field comments are required
3	src/main/java/labcodeinspection/Email.java	7	CommentRequired: Field comments are required
4	src/main/java/labcodeinspection/Email.java	7	RedundantFieldInitializer: Avoid using redundant field initializer for 'password'
5	src/main/java/labcodeinspection/Email.java	8	CommentRequired: Field comments are required
6	src/main/java/labcodeinspection/Email.java	9	CommentRequired: Field comments are required
7	src/main/java/labcodeinspection/Email.java	9	LongVariable: Avoid excessively long variable names like defaultpasswordLength
8	src/main/java/labcodeinspection/Email.java	10	AvoidFieldNameMatchingTypeName: It is somewhat confusing to have a field name matching the declaring class name
9	src/main/java/labcodeinspection/Email.java	10	CommentRequired: Field comments are required
10	src/main/java/labcodeinspection/Email.java	12	CommentRequired: Public method and constructor comments are required
11	src/main/java/labcodeinspection/Email.java	12	MethodArgumentCouldBeFinal: Parameter 'firstName' is not assigned and could be declared final
12	src/main/java/labcodeinspection/Email.java	12	MethodArgumentCouldBeFinal: Parameter 'lastName' is not assigned and could be declared final
13	src/main/java/labcodeinspection/Email.java	17	CommentRequired: Public method and constructor comments are required
14	src/main/java/labcodeinspection/Email.java	18	SystemPrintln: Usage of System.out/err
15	src/main/java/labcodeinspection/Email.java	19	SystemPrintln: Usage of System.out/err
16	src/main/java/labcodeinspection/Email.java	22	CommentRequired: Public method and constructor comments are required
17	src/main/java/labcodeinspection/Email.java	22	MethodArgumentCouldBeFinal: Parameter 'depChoice' is not assigned and could be declared final
18	src/main/java/labcodeinspection/Email.java	36	MethodArgumentCouldBeFinal: Parameter 'length' is not assigned and could be declared final
19	src/main/java/labcodeinspection/Email.java	37	LocalVariableCouldBeFinal: Local variable 'set' could be declared final
20	src/main/java/labcodeinspection/Email.java	38	LocalVariableCouldBeFinal: Local variable 'password' could be declared final
21	src/main/java/labcodeinspection/Email.java	40	LocalVariableCouldBeFinal: Local variable 'rand' could be declared final
22	src/main/java/labcodeinspection/Email.java	46	CommentRequired: Public method and constructor comments are required
23	src/main/java/labcodeinspection/Email.java	48	UseLocaleWithCaseConversions: When doing a String.toLowerCase()/toUpperCase() call, use a Locale
24	src/main/java/labcodeinspection/Email.java	48	UseLocaleWithCaseConversions: When doing a String.toLowerCase()/toUpperCase() call, use a Locale
25	src/main/java/labcodeinspection/EmailApp.java	7	AtLeastOneConstructor: Each class should declare at least one constructor
26	src/main/java/labcodeinspection/EmailApp.java	7	CommentRequired: Class comments are required


Archivo | C:/Users/fupifigu/Desktop/final/CodeInspection/reports/pmd-report.html

PMD report

Problems found

# File Line Problem