****

**COMSATS UNIVERSITY ISLAMABAD**

**LAHORE CAMPUS**

**Topics in Software Engineering I**

**CSE 350**

**Submitted By : Rashid Riaz – CIIT/FA29-BSE-081/LHR**

**Yamaan Obaid - CIIT/FA29-BSE-071/LHR**

**Abdullah Arshad - CIIT/FA29-BSE-104/LHR**

**Section : FA19-BSE-A**

**Submitted To : Dr. Fatima Sabir**

**Date of submission : Monday, October 3, 2022**

**Assignment 1 Marks 10**

Enable the project using any IDE for code review. It would be better to use Eclipse or NetBeans. You may use VS Code too.

1. Explore your project and find all options discussed in lecture 2 from Chapter 2. You need to prepare a google doc that highlights the potential issue that your code has. You may identify these issues with the help of

I) Identify deprecated technology or APIs. Just report the diagram.

ii) Potential issues missing in the documentation but available in the code

iii) missing of technical documentation (if applicable)

## solution:

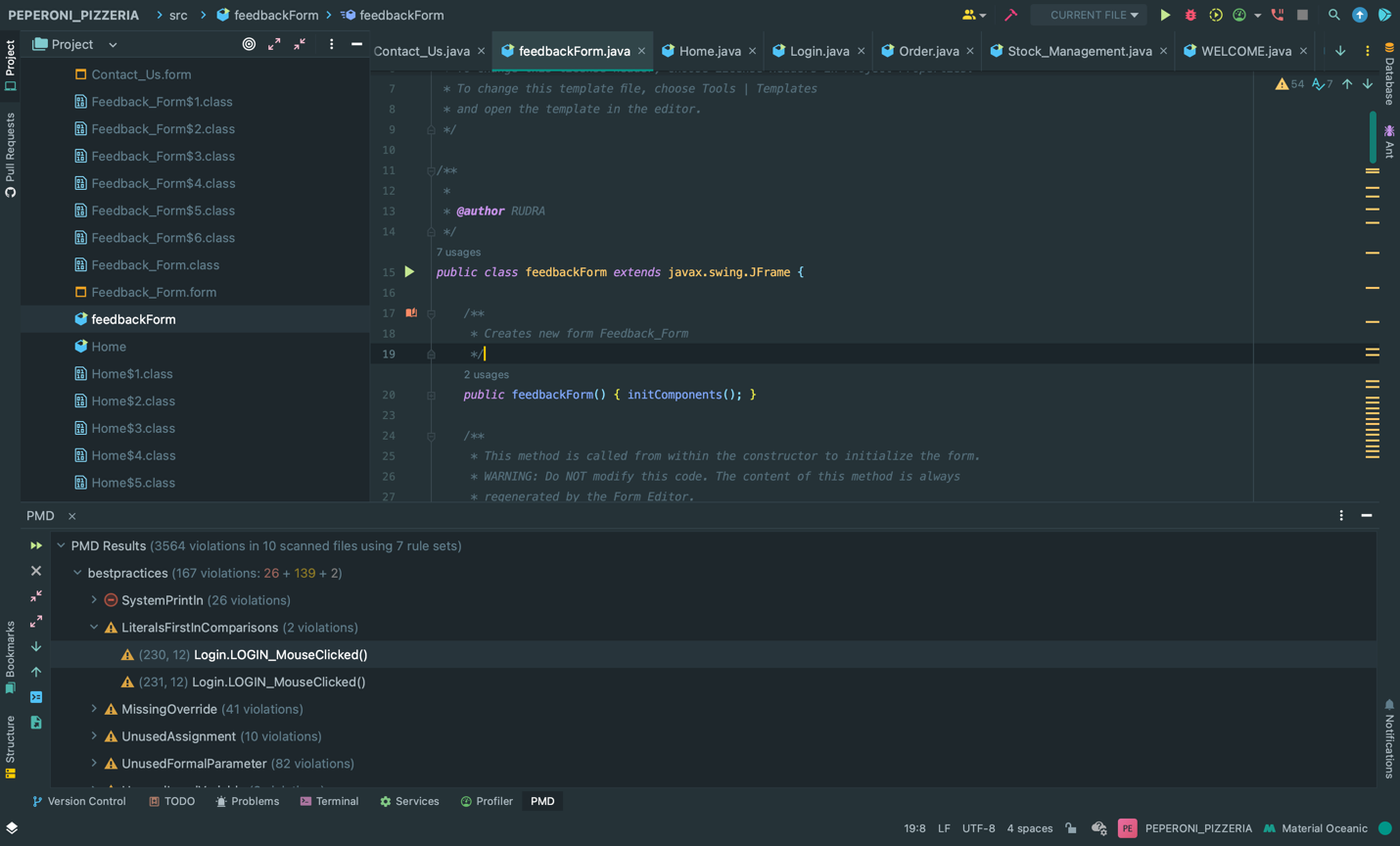
There is no documentation included in the repository, so identifying any issues in the documentation is not possible.

1. Use PMD to help identify potential coding errors and customize the rules you use to make sure only pertinent rules are applied to your source code.
   1. Create & view code issues directly from your editor
   2. Track & prioritize code improvements like technical debt
   3. Check your code quality
   4. Apply at least 3 PMD rules with the help of tool
   5. Generate the Abstract Syntax Tree of your source code using PMD.

## Solution:

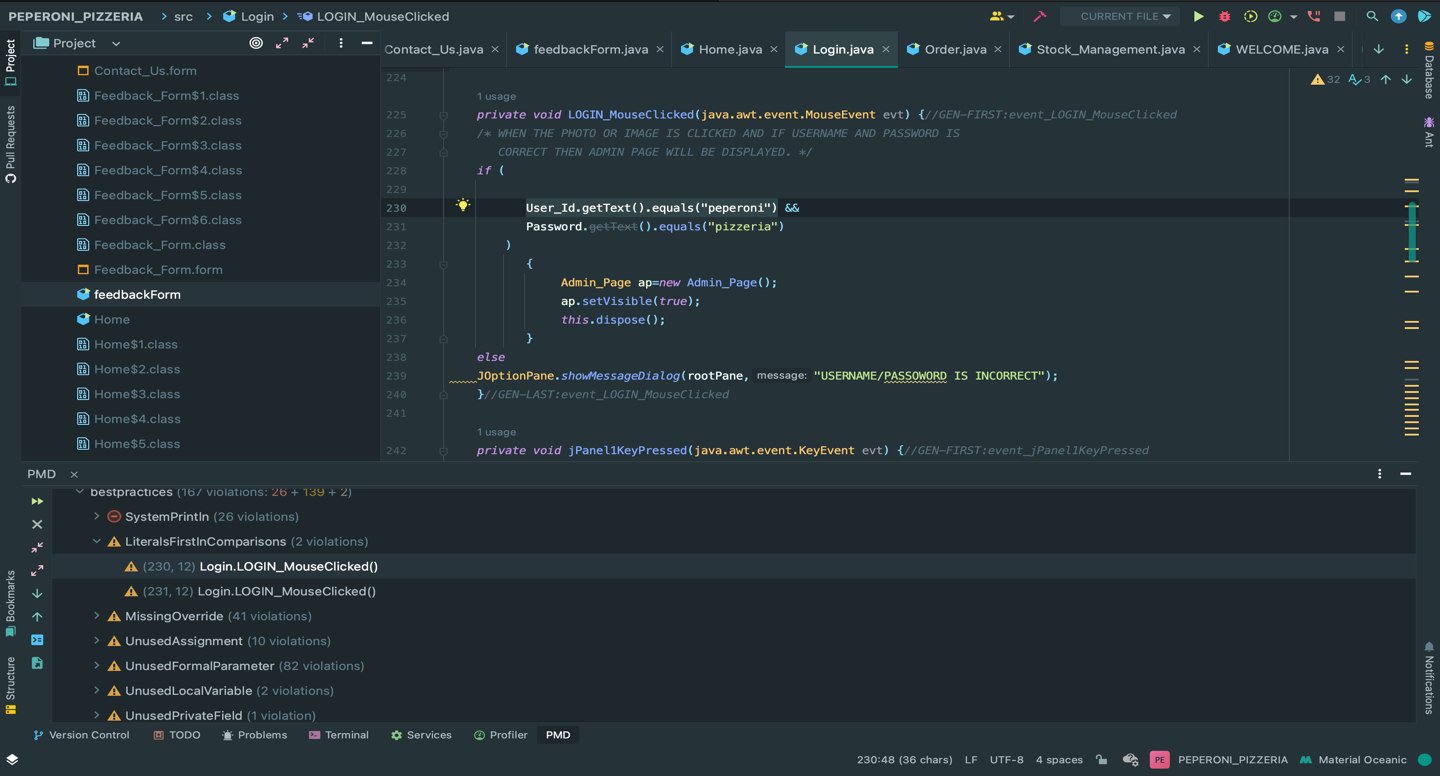
**1. Class names are not as per the standard naming convention**

Class name should be in pascal case i.e., FeedbackForm



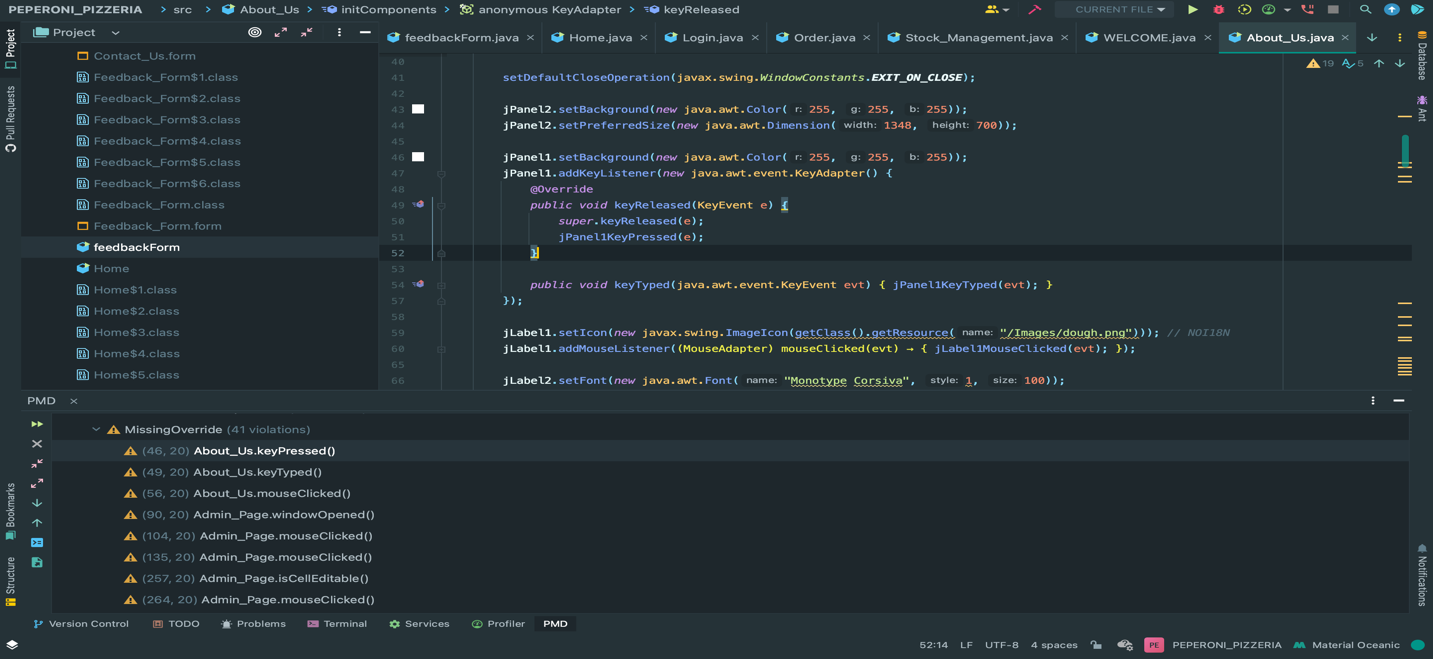
**2. Using Literals on right side in comparisons is not a good practice.**

While making comparison, placing literal on the left side is always a good practice as it won’t give a null pointer exception in case the variable is null. i.e. “peperoni”.equals(user\_id.getText())

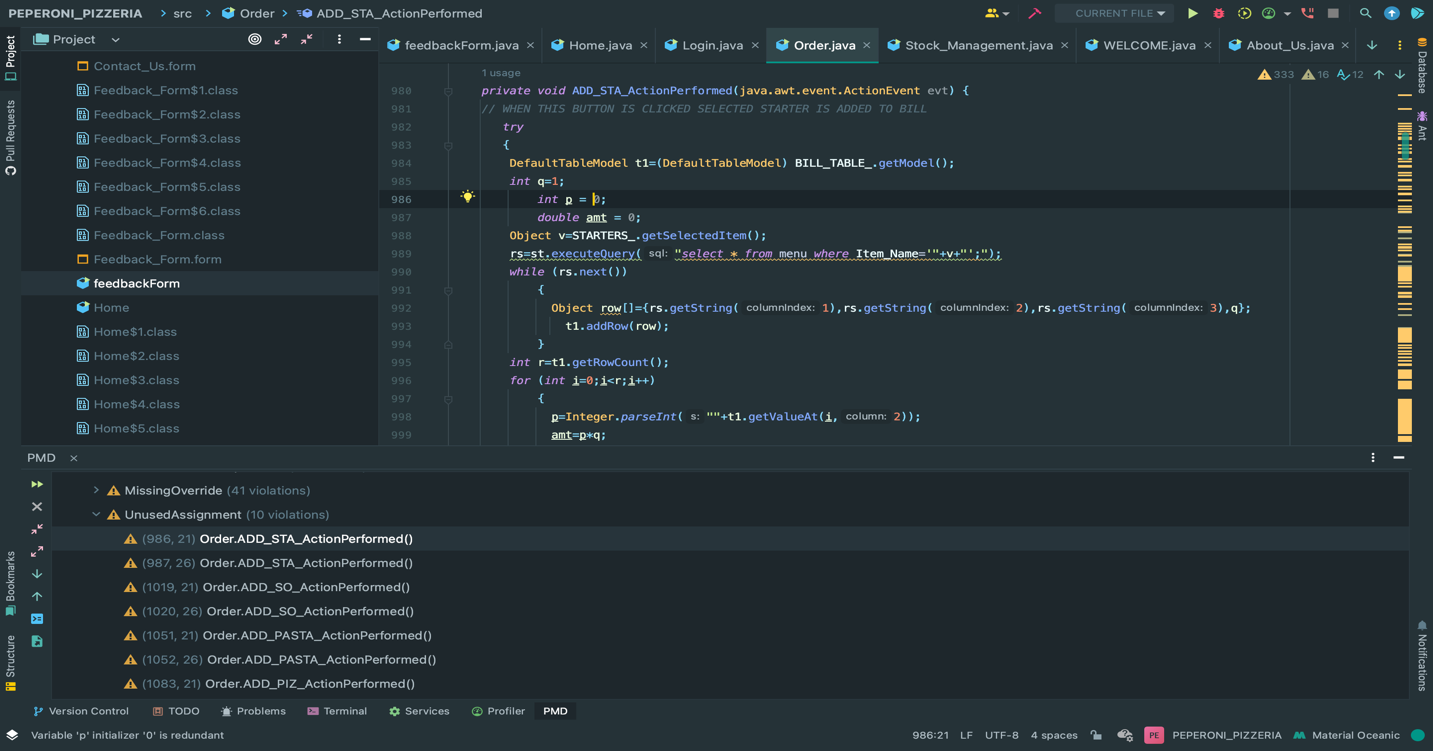


**3. Missing @override annotation**

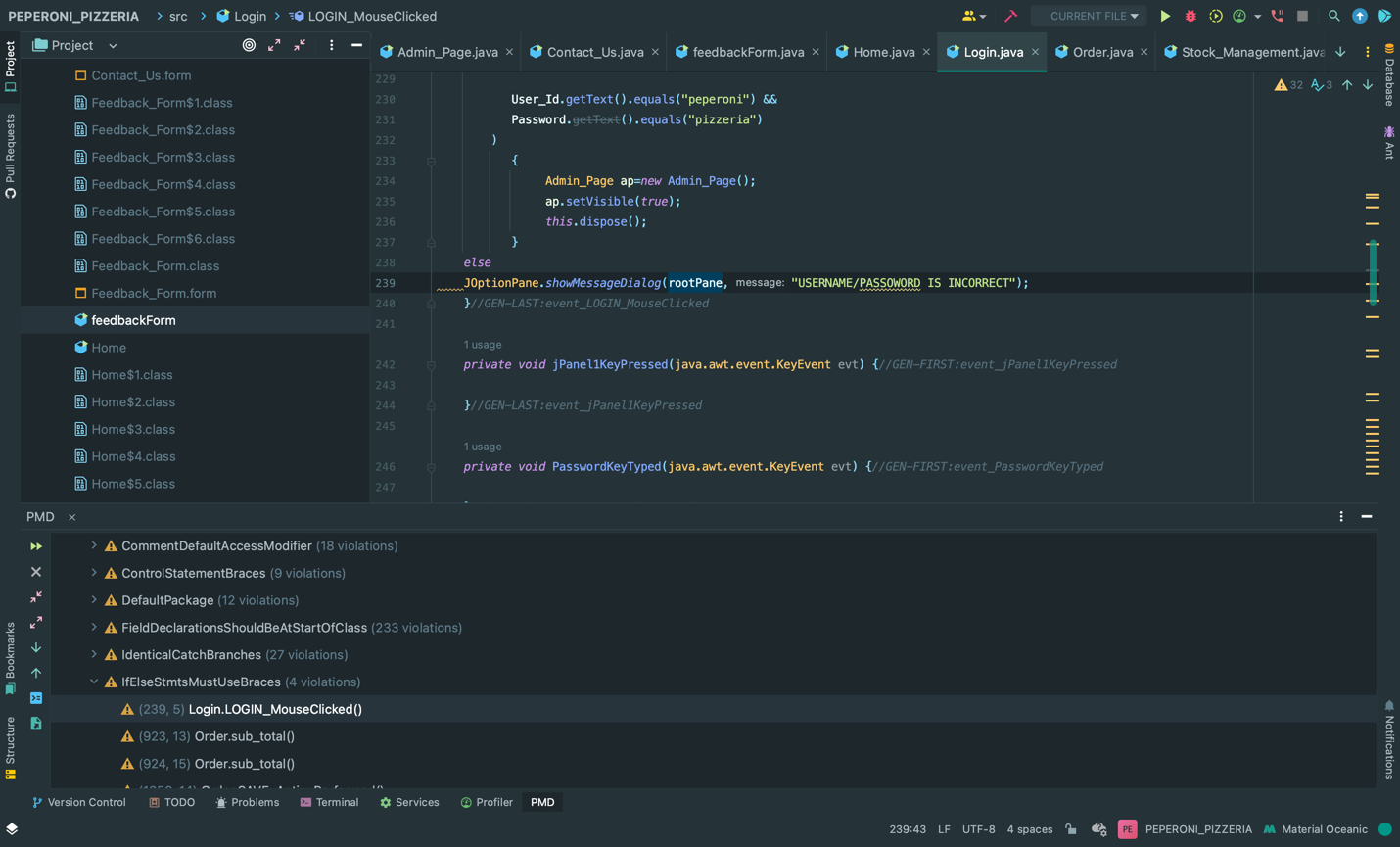
@Override annotation provides some extra information while generating documentations and for informing the developer that the code intends to override the method from a superclass.



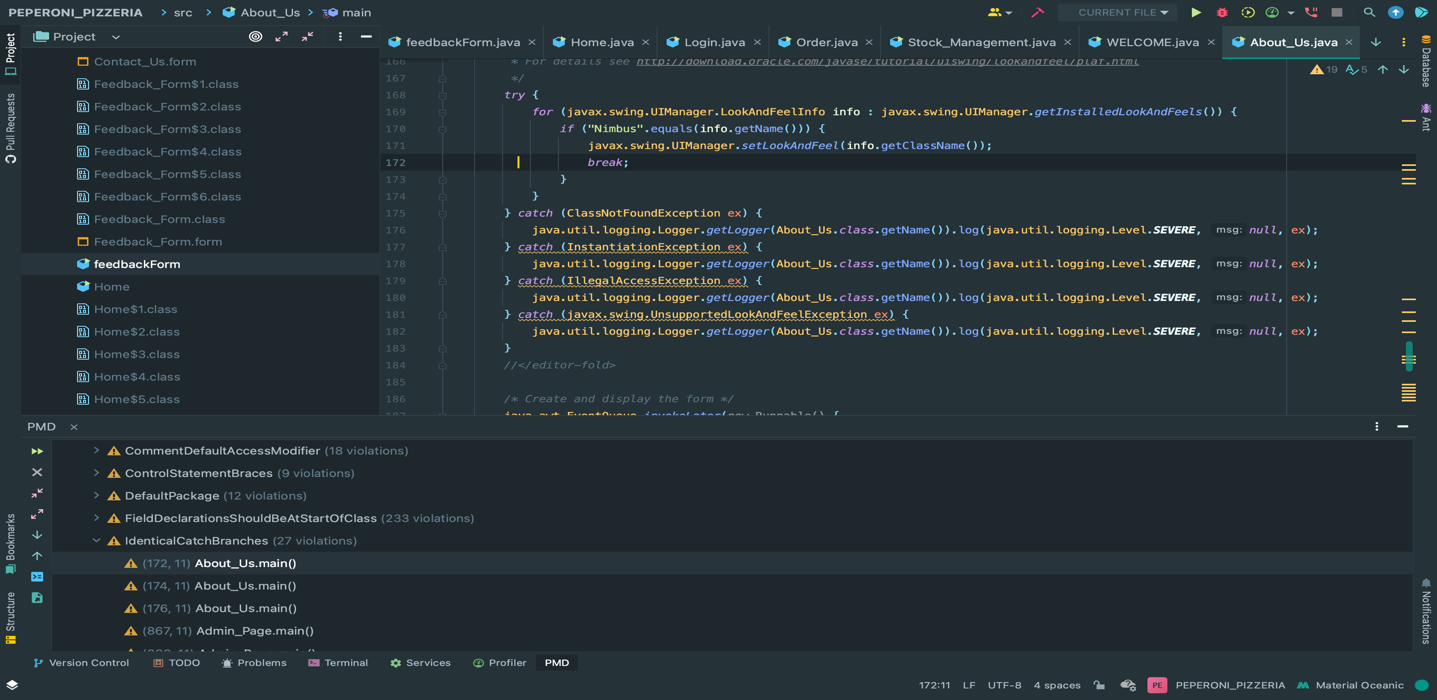
**4. Unused assignments**



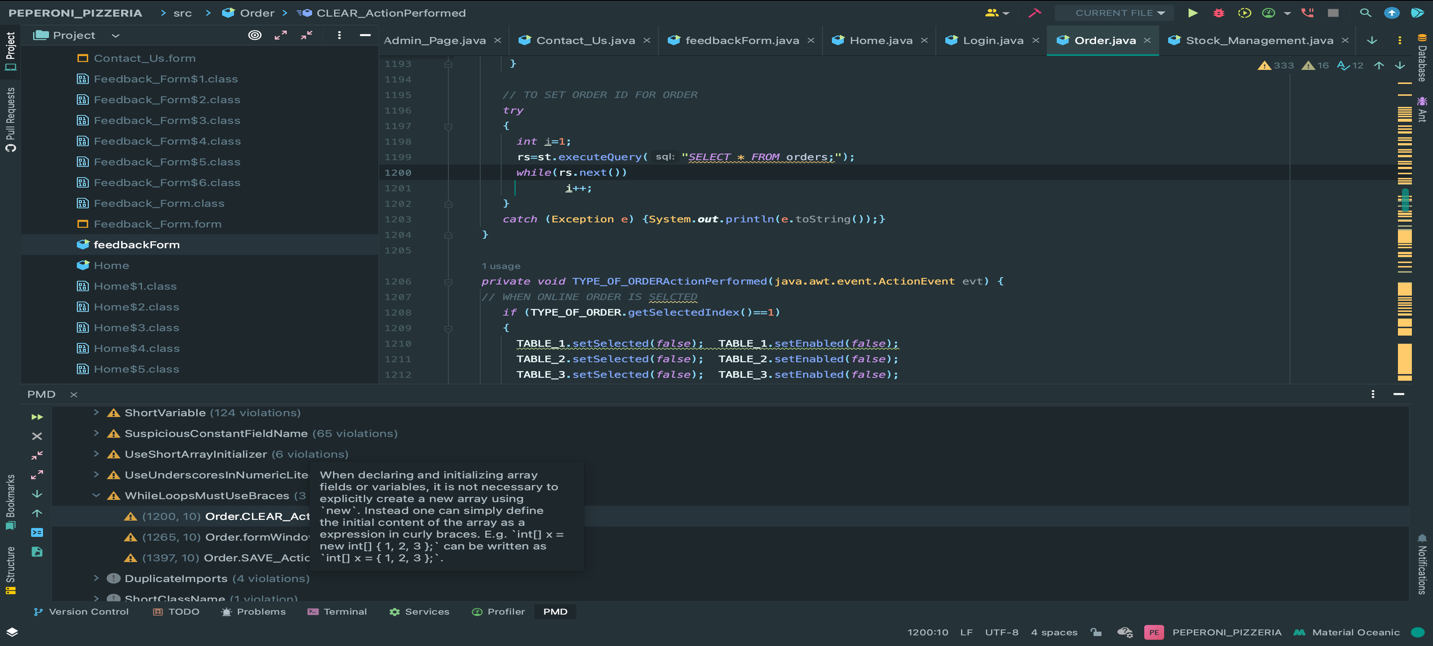
1. Use the Check Style tool to review your code. At least apply 2 rules on your code using Check Style
   1. If/Else statements must use braces:



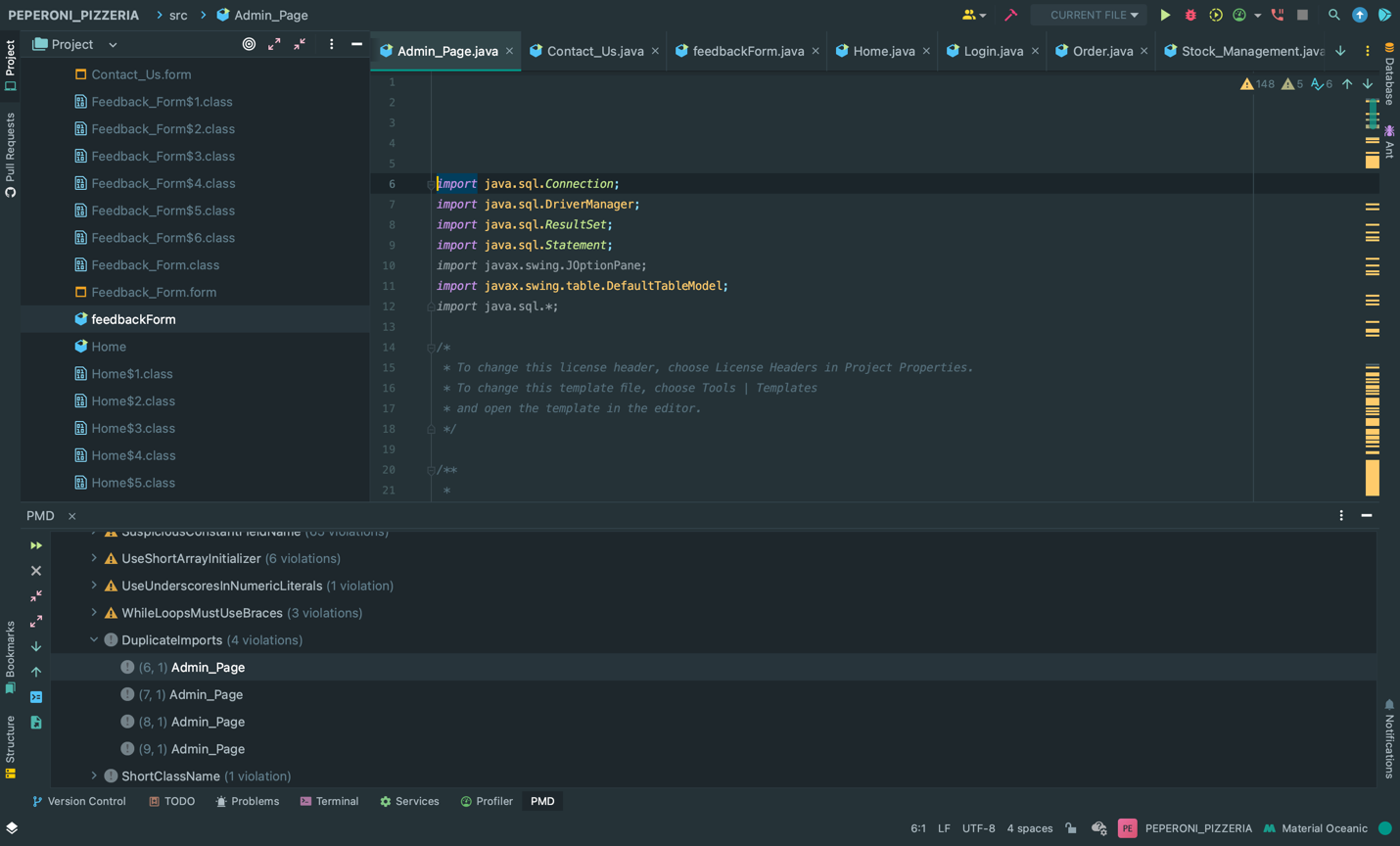
* 1. Identical catch branches



* 1. Loop must user braces



* 1. Duplicate imports



* 1. Unnecessary fully qualified names

