Description:
This assignment requires working/using the project (software) that needs to be compiled (Make sure you choose a good project and resolve any dependencies using its installation manual or you can use compilable projects that I have provided in the previous assignment folder).

Using PMD:

Locating and fixing software defects (bugs) is one of the most expensive tasks involved in software development. Because of this, software projects benefit greatly from tools that can automate the process of locating them. This is often accomplished using a technique called "static analysis", which examines the software and extracts useful knowledge from it. One such popular static analysis tool for Java programs is PMD. PMD is a source code analyzer. It finds common programming flaws like unused variables, empty catch blocks, unnecessary object creation, and so forth. It supports Java, JavaScript, Salesforce.com Apex, PLSQL, Apache Velocity, XML, XSL. Additionally it includes CPD, the copy-paste-detector. CPD finds duplicated code in Java, C, C++, C#, PHP, Ruby, Fortran, JavaScript, PLSQL, Apache Velocity, Ruby, Scala, Objective C, Matlab, Python, Go, Swift and Salesforce.com Apex.

Task:
For this exercise we will have PMD analyze one version of a software of your choice. You will then be asked to provide some detailed analysis of the generated report and to offer up a proposed fix for the identified bug:

Task:

1. Install the Eclipse plug-in for PMD
2. Run PMD on the same input code (preferably) from Part 2.
3. Pick 10 issues of different types.
4. Create a report and, for each issue, add to the report:
    1. The type of issue
    2. Whether it is a false or a true positive
    3. If it is a true positive, what are the necessary steps to fix it?
    4. How long it took you to check it / fix it.
5. Add to the report a concise comment about your experience with PMD (positives, negatives, other comments).