Remove unused members, don’t declare a variable and not use it.

Always close streams

Keep classes small and concise, they should only have 1 reason to change.

Classes should be structured in the following way (Instance variable, constructors, methods).

Remove old comments – most comments are unnecessary.

Code should be comment light and self-documenting – most comments are unnecessary.

Approach returning null or returning nothing from a method call with extreme caution – how can I test a method that returns void.

Don’t repeat yourself (DRY) Refactor code instead of copy paste.

If we have more than a few nested logic statements refactor out appropriate methods

Constructors should never normally have more than five parameters – think **Builder** pattern if you approaching this number.

Ensure encapsulation is properly followed (instance variables private with public getters and setters)

Members should be properly named (A variable called a or b is not self-documenting, you will not understand it in 2 weeks time)

Proper catch blocks should be added for exception handling instead of single Exception object handler.

Uniform coding standards for braces, loops, if-else, switch should be used across the application. This will usually be company specific.

Similar logic should be declared in a helper class and called from multiple places, this can be a static method.

A single statement should not go beyond the viewable area of the editor or IDE and should be split across multiple lines.

The best way to check if the String object is neither null nor empty string is to use the following code: if(“”.equals(str)).