**Coding Standards for Team #2**

1. **Comments**

We will be using /\* \*/ for long explanations (like explaining what a method/class does) and for the smaller comments we will use //.

1. **Decleration**

We will be declaring our variables as needed.

1. **Statements per line**

We will be writing only one statement per line for more readability.

1. **Return Statements**

The returned statement should contain parenthesis when it returns more than one value.

1. **If-statements, else-if statements, if-else-if statements**

We'll follow the Allman indentation style.

The if-statements should be like this:

if(condition)

{

Statements;

}

The else-if statements should be like this:

else if(condition)

{

Statements;

}

And the if-else-if statements should be like this:

if(condition)

{

Statements;

}

else if(condition)

{

Statements;

}

else

{

Statements;

}

1. **White spaces**

For this, we will follow the Java Code Convention document (from Oracle). We will use the following conventions:

1. Blank lines.
2. Two blank lines will be used:

* Between sections of a source file.
* Between class and interface definitions.

1. One blank line will be used:

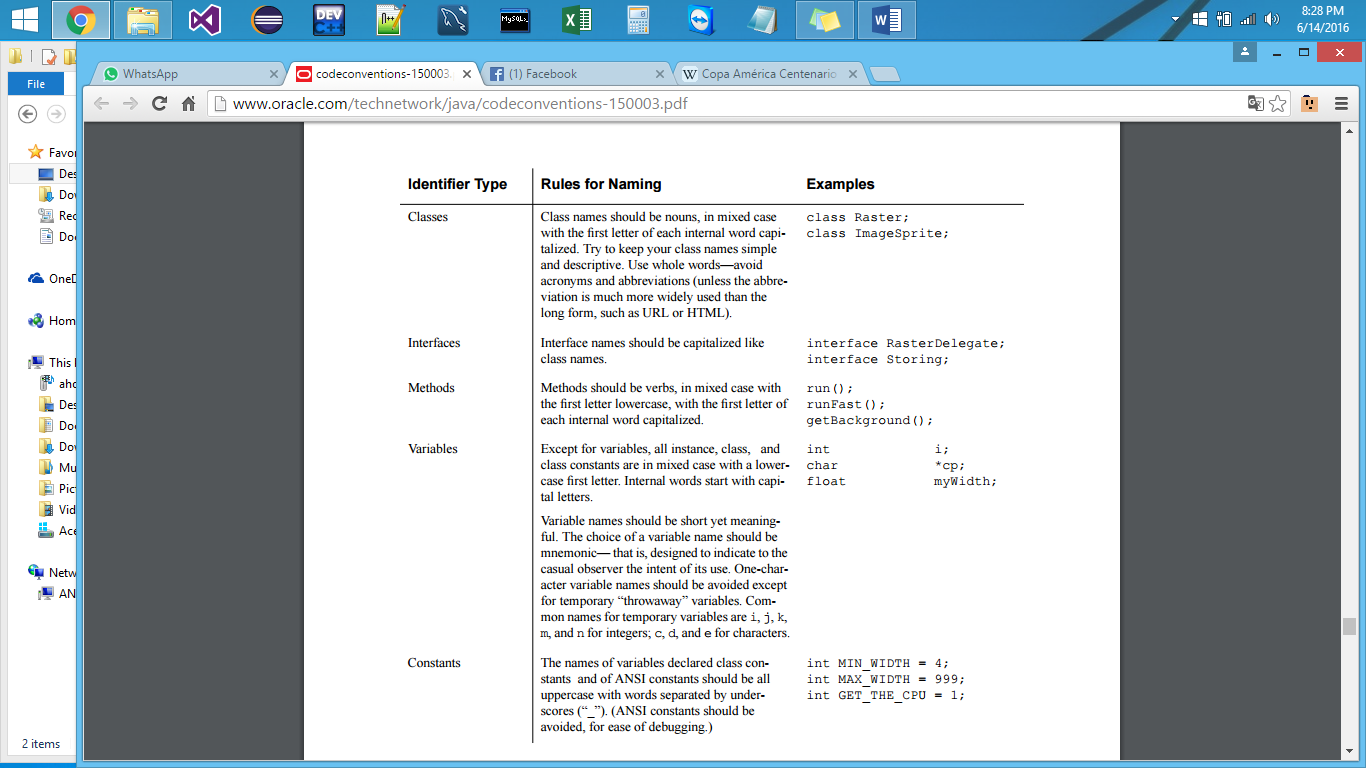
* Between methods.
* Between a local variable in a method and its first statement.
* Before a block or single lines comment.
* Between logical sections inside a method to improve readability.

1. Blank spaces.
2. When a keyword is followed by a parenthesis, it should be separated by a space; except by methods.
3. A blank space should appear after commas in argument lists.
4. All binary operators except . should be separated from their operands by spaces. Blank spaces should never separate unary operators such as unary minus, increment (“++”), and decrement (“--”) from their operands.
5. The expressions in a *for* statement should be separated by blank spaces.
6. Casts should be followed by a blank.
7. **Tabs**

For this particular, the use of the oracle standard will not be used. Tab will be the standard for indentation and line wrapping.

1. **Naming**

Naming convention for classes, interfaces, methods and variables are detailed in the following table:



1. **References**
2. *Java Code Convention*. <http://www.oracle.com/technetwork/java/codeconventions-150003.pdf>