**Welcome to the RISK wiki!**

**Coding conventions:**

1. **Naming Conventions for:**

Package: package names start with lower case letters.

Interface and Classes: Class names must start with the Capital Letters. In case, If class names has multiple words, inner words must start with capital Letters

Example: String, StringBuffer

Global Variables: Using meaningful names for global variables instead of just cryptic abbreviations makes the code easier to read and understanding. When variable names contains more than two words, we should use camel case.

Example: name instead of n and computerScience instead of cs.

Local Variables: First letter should start with lower case and then follow camel case rule.

Example: noOfPlayers

Static Variables: Every subsequent letters are capitalized and every other word is separated with an underscore sign.

Example: NUM\_PLAYERS = 5

1. **Other Conventions:**

Parentheses: Opening parentheses “{” should be placed at the end of the same line as the declaration statement whereas the closing parentheses “}” starts a line indented to match its corresponding opening statement.

Example: public class Parentheis extends Object {

int noOfPlayers;

Parentheis (int i) {

noOfPlayers = i;

}

int emptyMethod() {}

if(true) {

return true;

}

else {

return false;

}

}

Comments: Use // to explain variable

Example: Player[] player ; // array of players

Block comments: /\* \*/ for method description

Example: /\*

\* Here is a block comment.

\*/

Javadoc comments: As per mentioned in Oracle documentation.

Example: /\*\*

\* Here is a block comment.

\*/

The difference between Block comments and Javadoc comments is asterisk(\*). In Javadoc, first line contains two asterisk i.e. /\*\*.

1. **Testing Conventions:**

For testing, follow the rules defined below:

Testing Class name should start according to the tested class and at the should suffix test.

Example: if you are testing class GameController then its corresponding test class would be GameControllerTest.

Methods name should start according to the tested method and at the should suffix test.

Example: If we have to test intializeReinforcementArmies() method then

@Test

public void intializeReinforcementArmiesTest()

Note: do include @test annotation before the test method.

Without a single assert class the testing is not relevant. So use either of the available assert classes: assertArrayEquals, assertEquals, assertTrue, assertFalse, assertNull, assertNotNull, assertSame, assertNotSame, assertThat and fail.

1. **Indentation:**

Using proper indentation makes the program more human readable and gives an indication of scope without affecting the program directly.

Example:

if (a < 10 && b < 20 || c < 60) {

return true;

}

else {

System.out.println(“Correct”) ;

return false;

}