

CS2 Style Guide (Java)

Dr. Szumlanski

Spring 2021

- ★ Capitalize the first letter of all class names. Use lowercase for the first letter of all method names.
- ★ Any time you open a curly brace, that curly brace should start on a new line.
- ★ Any time you open a new code block, indent all the code within that code block one level deeper than you were already indenting.
- ★ Be consistent with the amount of indentation you're using, and be consistent in using either spaces or tabs for indentation throughout your source file. If you're using spaces for indentation, please use at least two spaces for each new level of indentation, because trying to read code that uses just a single space for each level of indentation is downright painful.
- ★ Please avoid block-style comments: `/* comment */`
- ★ Instead, please use inline-style comments: `// comment`
- ★ Always include a space after the `"/"` in your comments: `"// comment"` instead of `"//comment"`
- ★ The header comments introducing your source file (including the comment(s) with your name, course number, semester, NID, and so on), should always be placed above your import statements.
- ★ Use end-of-line comments sparingly. Comments longer than three words should always be placed above the lines of code to which they refer. Furthermore, such comments should be indented to properly align with the code to which they refer. For example, if line 16 of your code is indented with two tabs, and line 15 contains a comment referring to line 16, then line 15 should also be intended with two tabs.
- ★ Please do not write excessively long lines of code. Lines must be no longer than 100 characters wide.
- ★ Avoid excessive consecutive blank lines. In general, you should never have more than one or two consecutive blank lines.
- ★ Please leave a space on both sides of any binary operators you use in your code (i.e., operators that take two operands). For example, use `(a + b) - c` instead of `(a+b)-c`. (The only place you do not have to follow this restriction is within the square brackets used to access an array index, as in: `array[i+j]`.)
- ★ When defining or calling a method, do not leave a space before its opening parenthesis. For example: use `System.out.println("Hi!")` instead of `System.out.println ("Hi!")`.
- ★ Do leave a space before the opening parenthesis in an *if* statement or a loop. For example, use `use for (i = 0; i < n; i++)` instead of `for(i = 0; i < n; i++)`, and use `if (condition)` instead of `if(condition)` or `if(condition)`.
- ★ Use meaningful variable names that convey the purpose of your variables. (The exceptions here are when using variables like *i*, *j*, and *k* for looping variables or *m* and *n* for the sizes of some inputs.)
- ★ Do not use `var` to declare variables.