

Coding Standards

Team Soury

Christian Souders

Max Towery

1. Naming Conventions

- a. Use descriptive variable/function/class names to reduce unnecessary comments and ensure our code is easy to read
 - I. If comments are needed be as succinct as possible
 - II. We permit the use of acronyms, but we must include comments indicating what the acronym stands for
- b. Classes and Interfaces will be in PascalCase
- c. Variables and Methods will be in camelCase
- d. Constants will be in UPPERCASE
- e. For loops will use the following conventions:
 - I. Start with variable i
 - II. If nested use j, then k.. y, z

2. Presentation

- a. After method declaration, return once before using curly brackets
- b. Always use curly brackets for conditional checks i.e. if/else, for, while
 - I. To prevent logic errors and make maintenance easier
- c. Align curly brackets with the start of statement declaration that corresponds to the brackets
- d. Indent each block of code following a beginning curly bracket symbol({)
- e. Ensure that each line of code is less than 100 characters long

3. Documentation

- a. Include comments at top of classes that include the following fields
 - I. Author + Team name
 - II. Running log of changes to code including what was changed and the date
- b. Include comments above any unclear methods
- c. Include comments on if/else statements if it is unclear what the statement is checking for
- d. Use comments at the end of loops if they are longer than 5 lines of code

4. Testing

- a. Ensure that we use a verbose testing file that accounts for all edge cases