Coding Standards

File Formatting

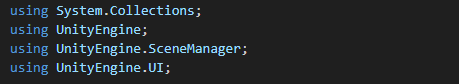
Indentation

Each new level of code signaled by the presence of a { on the previous line should be indented using 3 spaces and decremented by 3 spaces at the } character. Tabs should be avoided.

Libraries

Libraries should be ordered alphabetically at the head of a file. If multiple sub libraries are being used, the top-level library should be at the head of this list.

**Example:**

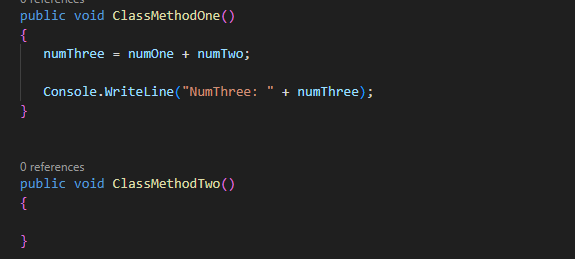


Whitespaces

Developers should follow the spacing guidelines below:

* All functions should be separated by at least 2 empty lines in code
* All logic/arithmetic operations done in code should have variables and operators separated by one space
* Place additional blank lines to show separation between unique sections of code

**Example:**



Naming Conventions

Folders

Folders should follow the PascalCasing format, where the first letter of every word is capitalized. No spaces, underscores, numbers, or special characters are used in the name.

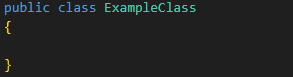
Files

File names should always be named like the class they contain. If more than one class is present in a file, the more relevant class should be used to name the file. File names should follow PascalCasing format, where the first letter of every word is capitalized. There should be no spaces, underscores, numbers, or special characters used in the name.

Classes

All class names should follow the PascalCasing format, where the first letter of every word is capitalized. There should be no spaces, underscores, numbers, or special characters used in the name. The name of the class should match the name of the file it resides in.

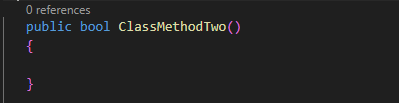
**Example:**



Functions

All function names should follow the PascalCasing format. There should be no spaces, underscores, numbers, or special characters used in the name. The starting ‘{‘ should appear on the line after the function name.

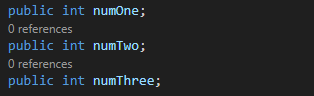
**Example:**



Variables

All variables should follow the camelCasing format, where the first word of the variable is lower case, and all other words capitalized. There should be no spaces, underscores, numbers, or special characters used in the name.

**Example:**

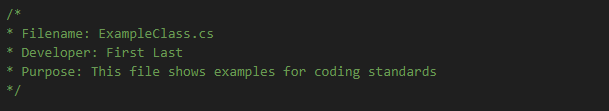


Documentation

Files

At the beginning of a file before any code or libraries, a comment should be formatted should be written as shown below.

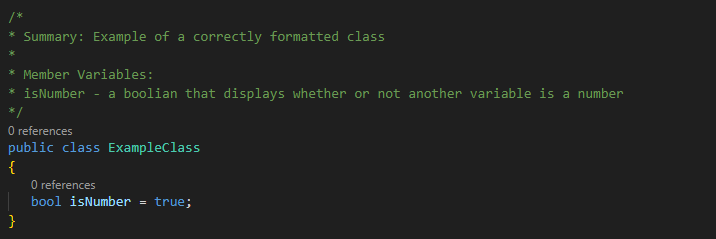
**Example:**



Classes

Class comments should be put right above the class declaration. As shown below, the first line should contain a summary of the class. Member variables should also be included with a space between the summary and itself as shown below.

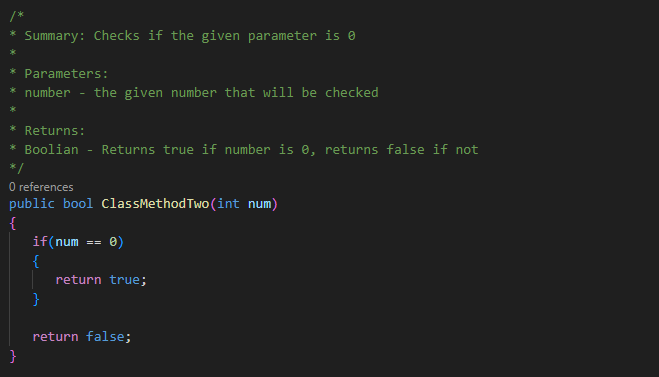
**Example:**



Functions

Function comments should be put right above the function declaration with the following format. Comments should include a summary, parameters, and returns. If the function returns void, then the returns section is not needed.

**Example:**



Prefabs

Comment Prefabs  
Name: Semi-generic  
Summary: include keywords  
Description: more detail of prefab, troubleshooting guide, integration instructions, etc. videos (to make BC happy) (will be graded)

Other Comments

Any additional comments should be placed above the line of code as shown below in the example. Comments should answer the question likely to be asked and are there to increase the readability of the code.

**Example:**

