Team M.E.M.E Official Coding Style Guide
A mixture of the Google C++ and Java Style Guides for C# programming in the Unity Game Engine.
Written by Judson James.

_____

____

General Styling

- **80 character limit per line**   -- this is so it prints on paper nicely.
- Tabs are **two spaces**   -- this is so it looks compact.
- **In-Line** Bracing   -- again, makes it look compact.
- Do **NOT** use "magic numbers"   -- it makes the code look more defined.

Code Styling

- Variables
  - All **primitive** variables will use the **lower camel-case** naming convention.
  - All **object** variables will use the **upper camel-case** naming convention.
  - If using a **private variable**, assign a "p" at the beginning of the name and start the rest of the name in camel-casing.
  - If using a **global variable**, assign a "g" at the beginning of the name and start the rest of the name in camel-casing.
  - If using a **constant variable**, use the "CONSTANT_CASE" naming convention.
  - All parameter variables will follow their primitive/object variable naming conventions.
- Functions
  - All **user-made** functions will use **upper camel-case** to differentiate from built-in functions.

Documentation

- Inside Documentation
  - Inside every script there should be a section at the top under the "using" Unity base inclusions, there should be a documented header containing:
    - Name of Author(s)
    - Date the Script was last modified
    - General Purpose of the script
  - Inside the class, all private variables should be listed after the class declaration
  - Before every function, comment one line such that it includes :
    - [function declaration] : [brief description of function]
    - This includes the "Start" and "Update" functions
  - Before every loop, complex boolean, or switch:
    - If it does four or more statements, explain what it is doing in one or two lines.
  - For every block of code that is associated with an action, separate it with an empty line before moving to another action in the function.

- Outside Documentation
  - Script's External Documentation
    - There won't be a README.md for every script, but there will be a separate documented file that will be used to explain everything within a script.