SuperNova Games Coding Standards

1. **Comments and Headers**

Begin a File with a header comment indicating the file’s purpose

* /\* Playermovement.cs script maps keys to inputs...

Install xxx in package manager to implement\*/

Function comments should preceed the function declaration, while shorter inline comments are reasonable at the end of lines.

* //Returns the value health of when mouse hovers over health bar for visual appeal
* OnMouseEnter(); // tracks mouse hover

Place comments denoting the ‘why’ of the code, not ‘what it does’

* // Imrpoves throughput using ‘\_\_\_’ function

Leave a comment denoting the purpose of a function if not immediately clear

* // reverses user input
* static string ProcessString(string input)
* {
* char[] charArray = input.ToCharArray();
* Array.Reverse(charArray);
* string reversed = new string(charArray);

1. **Naming Conventions**

Private variables should be written in camelCase

* private int myVariable
* private int enemySpawn

Public variables should be written in camelCase

* public int publicVariable
* public int gameObject

Functions should

Class names should be written in PascalCase, as well as all public members

* class GameManager
* class UserInput

Don’t use Hungarian notation, where you denote the type of variable in the name

* Do – private int length; Don’t - private int i\_Length; private int iLength;

1. **Principles**

No more than 100 characters per line, to avoid side scrolling and truncated lines

Follow ‘Don't repeat yourself’ principles to avoid duplicate logic

* DO
  + static void PerformSum(int num1, int num2)
  + {
  + PerformSum(5, 10);
  + PerformSum(20, 8);
* DON’T
  + int a = 5, b = 10;
  + Console.WriteLine("Sum: " + (a + b));
  + int x = 20, y = 8;
  + Console.WriteLine("Sum: " + (x + y));

No more than 3 nested if statements, this reduces clarity and performance

* DONT
  + If (isUser)
  + {
  + If (isMod)
  + {
  + If(isAdmin)
  + {

1. **Indentation and Spacing**

Do not have an opening and closing curly bracket on the same line, curly braces on their own line, not at the end of the function, class, iteration, etc.

* static void ExampleFunction()
* {
* for (int i = 0; i < 3; i++) // Loop from 0 to 2
* {
* if (i % 2 == 0) // Check if even

Standard 4 space tab / indents when changing scope

* static void ExampleFuntion()
* {

Add spaces between variables and operators.

* DO DON’T
  + Multiply(a \* b); Multiply(a\*b);