Programming Naming Rules PONS Style

Enlace video youtube: PONS Style Tutorial

VARIABLES

Common Variables Naming

Variables are declared using a letter that defines its type and underscore (letter_) followed by a descriptive name which first letter is in lower case (letter_name)

In case the descriptive name has **two** words, the second word first letter starts with upper case (letter nameDescription).

In case the descriptive name has **three or more** words, each word is separated using underscore and all start with uppercase except the first word (letter name Long Description).

EXAMPLE:

```
public bool b_moves;
public bool b_isMoving;
public bool b is Still Moving;
```

Scripts and GameObjects

Scripts are named using "script" keyword + underscore + script name
with first letter in underscore (script name)

GameObjects are named using "go" keyword + underscore + game object
name with first letter in underscore(go_name)

<u>EXAMPLE:</u>

```
public PlayerController script_playerController
public GameObject go player;
```

Components

Other components inside Game Objects are named using underscore + reference (first letter always in lower case) + underscore + componentName.

EXAMPLE:

```
SpriteRenderer _player_SpriteRenderer;
BoxCollider _enemy_BoxCollider;
```

Classes and Interfaces

Follow Unity official recommendation

https://unity.com/how-to/naming-and-code-style-tips-c-scripting-un
ity#classes-and-interfaces

Classes names are written with first letter of each word being an UpperCase:

```
public class ExampleClass : MonoBehaviour
{}
```

For interfaces we do the same as with classes but adding the prefix $\mbox{`I''}$ at the beginning.

```
// EXAMPLE: Interfaces
public interface IExampleInterface
{
    void Learn();
}
```

Methods

Follow Unity official recommendation

https://unity.com/how-to/naming-and-code-style-tips-c-scripting-un
ity#classes-and-interfaces

For methods we follow Classes naming convention:

```
public void ExampleVoid(){}
```

A good practice is to give methods descriptive names using verbs or asking questions depending on what the method is doing or which value is returning.