Documentation
Essentials

Summary

Guillem Poy
@guplem
TriunityStudios.com

This file is meant to describe the main features included in the asset.

All the code is commented with XML documentation so this file does not go in deep in every class implementation and neither how to use them.

If you want to see examples, know what each method does, most of the class functionality and how to use most of the asset's features, I recommend you to look into the "Examples" folder (and read the comments inside the "Essentials/Scripts" folder).

Essentials' Settings and Modifications

This window can be found under the "Window/Essentials/Settings" menu. In there you can apply recommended modifications.

Additionally, you can add custom modification simply by programming them in a class that inherits from the abstract lass "Modification".
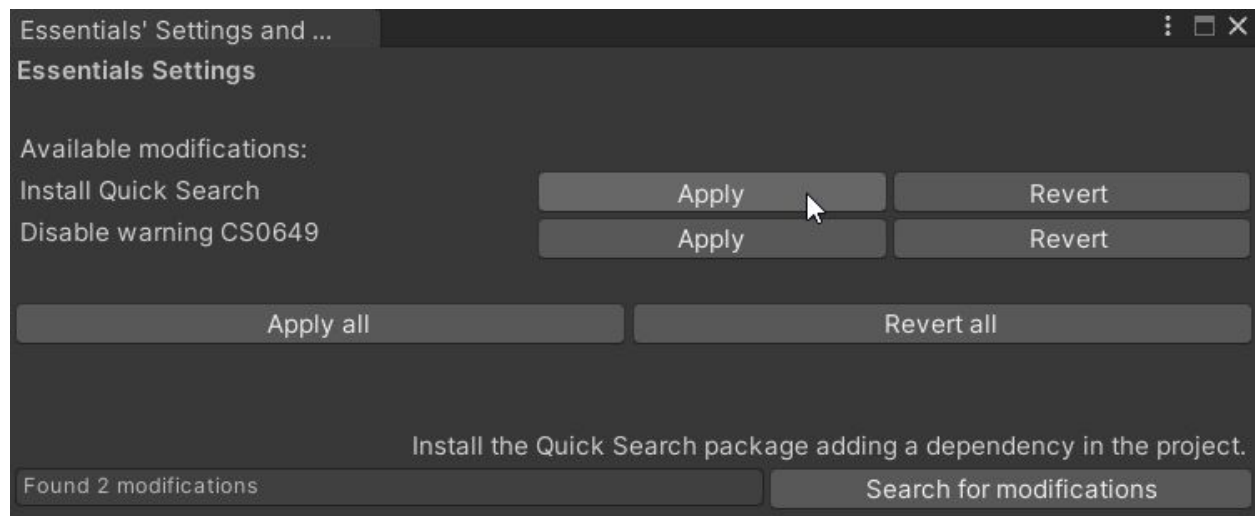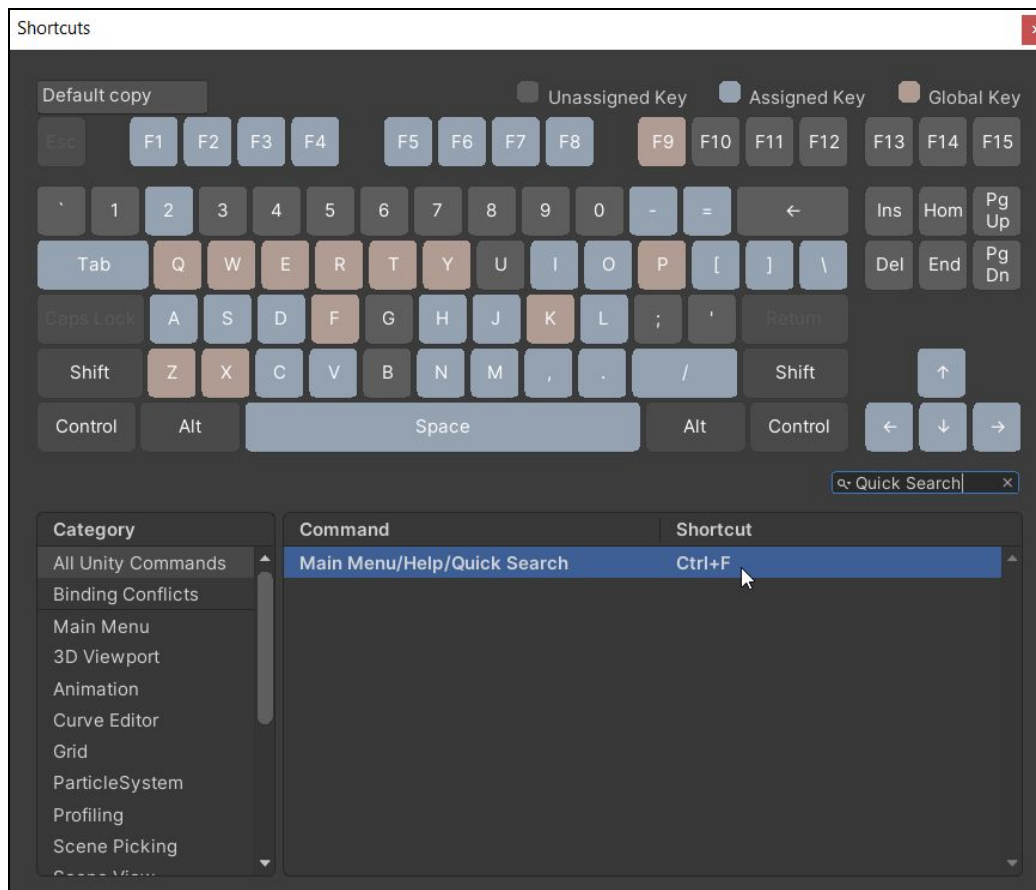

The available modifications by default are:

Quick Search

Easily allows you to install and uninstall the latest version of the package "Quick Search" (identified as *com.unity.quicksearch*). Changing the "Quick Search" shortcut to your preference is recommended[1].

---

[1] I personally recommend using "Ctrl+F" to trigger Quick Search

Warning C60649

Allows you to enable or disable the warning CS0649: "Field 'var' is never assigned to, and will always have its default value null." Which makes no sense most of the time in Unity projects.

Essentials Menu

At the top bar of Unity's editor you can find the "Essentials" menu. In there, some actions can be performed.

Console

- **Clear Console**: Clears the Unity Editor's console.

Improved Debug

Using "DebugPro" you can easily debug IEnumerables (lists, arrays, …) using the method "DebugPro.LogEnumerable(…)". Example: DebugPro.LogEnumerable(Array).

In addition, it implements all the functionality from the Unity Engine's "Debug" class.

You can see an example of its use in the scene "DebugProExample" which uses the code "DebugProExample.cs".

Pool System

The Pool class allow you to easily improve the performance of your game avoiding the constant instantiation and destruction of objects by automatically enabling and disabling them with a limited number of instances.

You can see an example of its use in the scene "PoolExample" which uses the code inside the file "PoolExample.cs".

Simple Animations

The SimpleAnimation class in conjunction with the SimpleAnimationsManager allows you to easily animate GameObject's components without having to deal with the extensive world of Unity's animations.

You can see an example of how an animation of a GameObject's transform could work in the scene "CubeAnimationExample" which uses the code "CubeAnimationExample.cs" and in the scene "UiAnimationsExample" which uses the code "UiAnimationsExample.cs". This examples uses the class "TransformAnimation" and "RectTransformAnimation", which can be taken as an example to create animations of any component following the same patterns shown in it.

Easy Random

The EasyRandom class allows you to easily work with randomness. It inherits the "System.Random" class, so it contains the same functionality plus some methods that are easier to understand than the original ones and some more that add new functionalities.

You can see an example of its use in the scene "EasyRandomExample" which uses the code inside the file "EasyRandomExample.cs".

Extensions

Multiple class extensions have been added to the asset to add some commonly-needed functionality in video games into some already implemented classes.

Some of the extended classes are:

- Component
- Float
- GameObject
- ICollection
- IEnumerable
- Int
- LayerMask
- RectTransform
- String
- Transform
- Vector

Other Features

Shortcuts

- Save Scene and Project (Ctrl + Shift + Alt + S)
- Save project (Ctrl + Alt + S)
- Clear Console (Ctrl + SPACE)

Menu Items

- File/Save Scene and Project