UnityLibrary - Documentation

https://github.com/Luca00IT/UnityLibrary/tree/main

Description

UnityLibrary contains many different libraries for Unity to speed up the process of developing your own video game. Later you can consult the documentation in order to understand how to implement these libraries in your projects and use the functions

Prerequisites

- Unity Engine v2021.3.4f1 or higher
- Visual Studio 2022 Community + "Game Development with Unity" extension
- Initialize the library at the beginning of each script:

```
// using libraryname
```

FadeTools

The FadeTools library allows you to use functions that can create and edit a RawImage by creating a "FadeIn" and "FadeOut" effect across the entire screen. You can use this library when, for example, you need this effect for loading a layer or other occasions when you will need this effect.

Import the library

```
using FadeTools;
```

Initialize the variables

```
[SerializeField]
private RawImage rawImage;
private float duration;
```

Create RawImage

Creates a black-colored image that covers the entire camera.

```
CreateRawImage();
```

Fade In Makes the newly created image darker and darker in a time determined by the duration variable.

```
FadeIn(rawImage, duration);
```

Fade Out Makes the newly create image lighter in a time determined by the duration variable.

```
FadeOut(rawImage, duration);
```

ActionTools

The "ActionTools" library allows you to quickly manage interaction with GameObjects in the first person, using "tags" as the interaction method.

Import the library

```
using ActionTools;
```

Initialize the variables

```
[SerializeField]
private List<string> interactiveTags = new List<string>
private int tagIndex;
private float maxRaycastDistance;
```

Store the interactive tags

```
void Start()
{
    Action.SaveTags(interactiveTags);
}
```

Check if the GameObject the player is looking is interactive, then do stuff

```
void Update()
{
    if (Action.IsObjectInteractable(out tagIndex, maxRaycastDistance))
    {
        switch(tagIndex)
        {
            case 0:
                //do stuff here
                Debug.Log("The GameObject is interactive: " + interactableTags[tagIndex]);
                break;
        }
```

```
    }
}
```

License

All code written within the repository is under **GNU General Public License v3.0**