

Unity

Platformer 2D:
Trabalhando com
Scriptable Objects

Atualizando interface
com scriptable objects



Atualizando a interface com Scriptable Objects

Seguindo o exemplo mostrado em aula, podemos atualizar a interface de uma maneira diferente: mais funcional e desacoplada, ou seja, sem estar presa ao código do jogo, e podendo ser reaproveitada.

Criamos um *scriptable object* que recebe uma valor. Ele pode ser qualquer valor que você queira usar: *float*, *int*, *string*, etc.

Esse *scriptable object* vai servir pra guardarmos a variável, e acessarmos ela de diversos lugares necessários.

No exemplo abaixo, fizemos um *scriptable object* que guarda uma variável *int*.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[CreateAssetMenu]
public class SOInt : ScriptableObject
{
    public float value;
```

Atualizando a interface com Scriptable Objects

É importante colocar o componente **[CreateAssetMenu]** em cima da declaração da classe. Ele fará com que esse menu abaixo apareça, clicando com o botão direito na pasta do projeto, e assim, seja possível criar os Scriptable Objects.

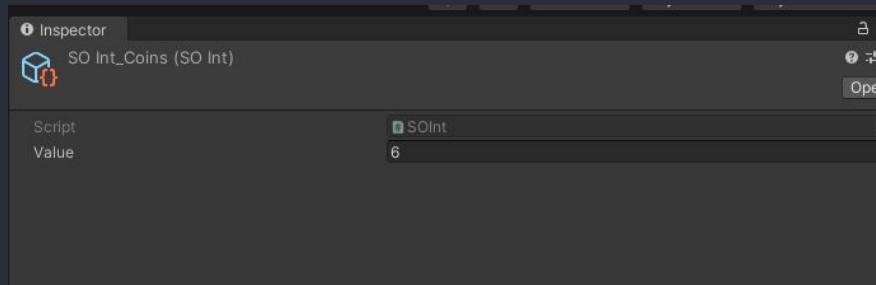
```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

[CreateAssetMenu]
@ Unity Script | 2 references
public class SOInt : ScriptableObject
{
    public float value;
}
```



Atualizando a interface com Scriptable Objects

Uma vez criado, o arquivo será desta maneira:



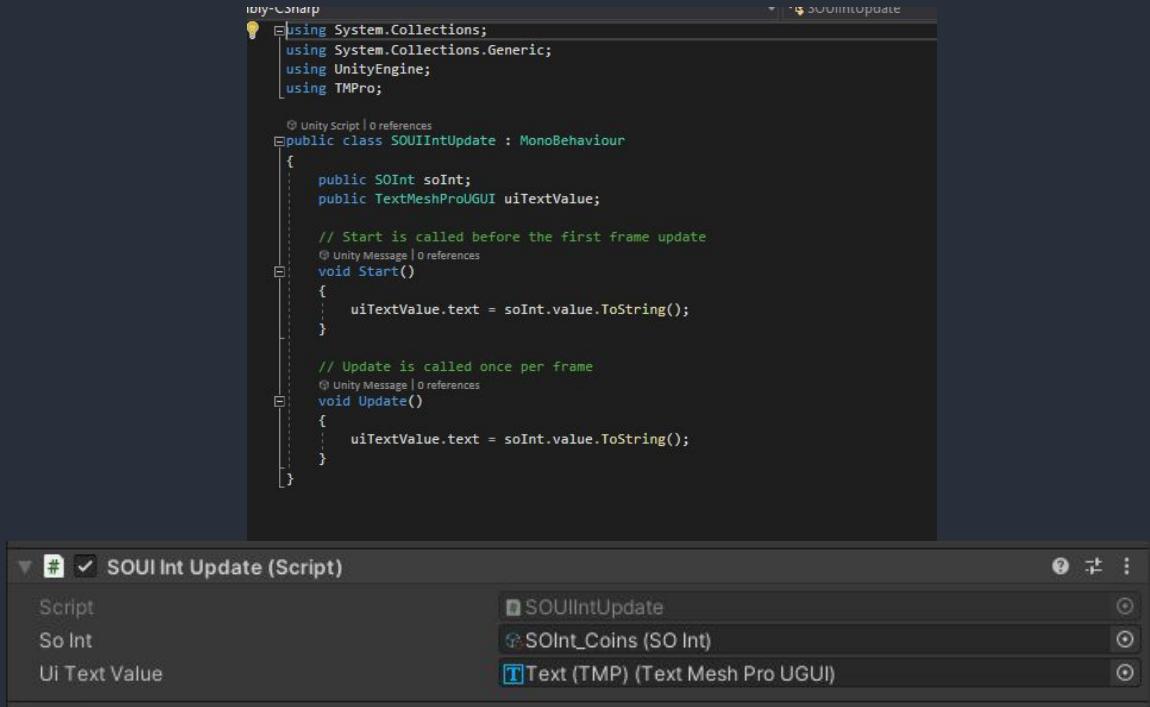
Atualizando a interface com Scriptable Objects

Podemos acessar esse arquivo pelo código da seguinte maneira:

```
public SOInt coins;  
  
I reference  
public void AddCoins(int amount = 1)  
{  
    coins.value += amount;  
    UpdateUI();  
}
```

Atualizando a interface com Scriptable Objects

Com essa técnica, podemos criar classes que vão nos ajudar a atualizar as UIs, como no exemplo abaixo, deixando o nosso código bem desacoplado, e bastante reutilizável.



```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using TMPro;

public class SOUIIntUpdate : MonoBehaviour
{
    public SOInt soInt;
    public TextMeshProUGUI uiTextValue;

    // Start is called before the first frame update
    void Start()
    {
        uiTextValue.text = soInt.value.ToString();
    }

    // Update is called once per frame
    void Update()
    {
        uiTextValue.text = soInt.value.ToString();
    }
}
```

SOUINIntUpdate (Script)

Script	SO Int	Ui Text Value
SOUINIntUpdate	SOInt_Coins (SO Int)	Text (TMP) (Text Mesh Pro UGUI)