Curso Técnico

de Programação de Jogos Digitais

Aula 14

índice

[1. Lógica do jogo 3](#_Toc169511525)

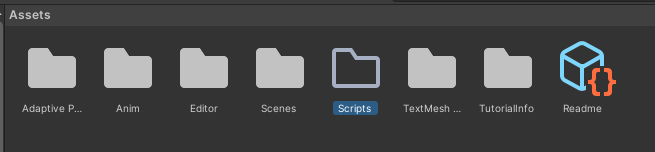
[2. Referencias 8](#_Toc169511526)

# Lógica do jogo

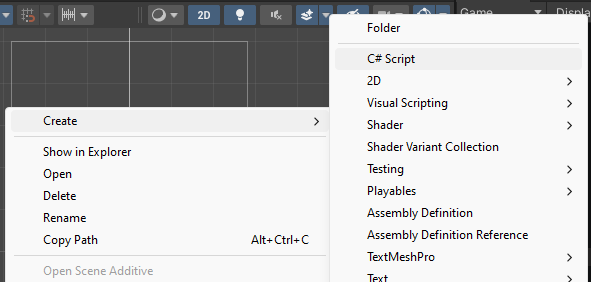
Agora vamos partir para a lógica do jogo. Teremos a lógica do movimento horizontal da bolinha que servira para coletar os quadrados que caem e assim criar o movimento no obstáculo onde ao encostar na bolinha o jogo chegará ao fim.

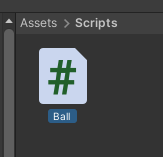
Devemos também criar o movimento no colectable ao encostar na bolinha acrescentar valor a pontuação, ou seja, adicionar a lógica do score para mostrar a pontuação do jogo.

Para organizar nosso projeto vamos criar uma pasta script onde iremos colocar nossos scripts do jogo.



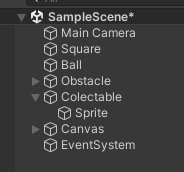
Dentro da pasta devemos criar um arquivo csharp chamado Ball. Para criar um script clique com botão direito na hierarquia do projeto e escolha C# Script, renomeie e salve na nova pasta.



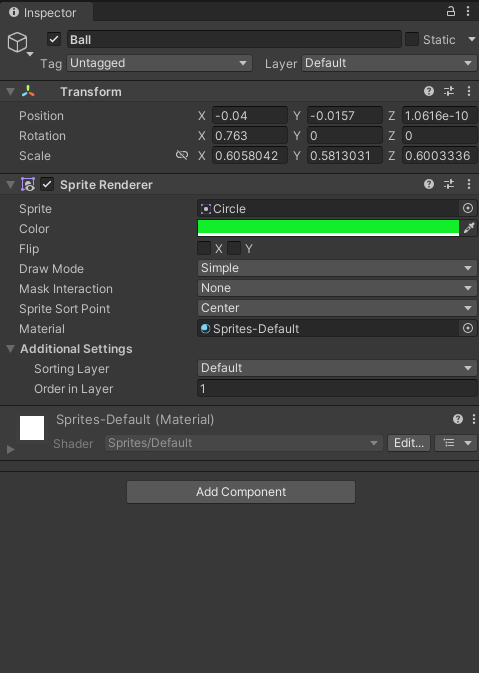


Renomeie o Circle para Ball

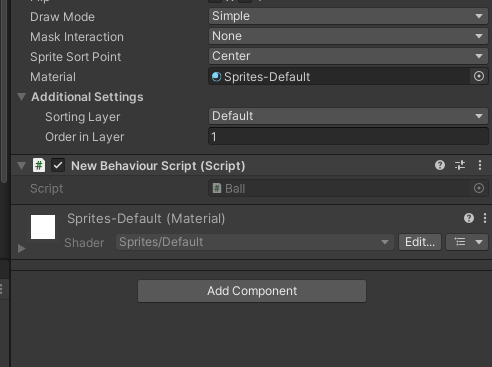
Agora devemos arrastar o script Ball para dentro dos componentes do objeto. Escolha na hierarquia o objeto Ball.



Veja na janela de inspector suas configurações.



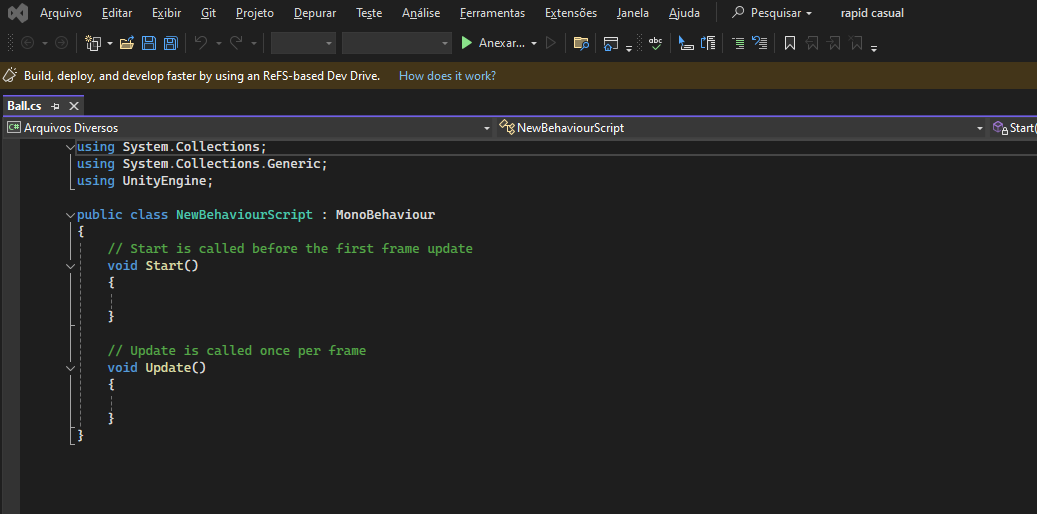
Arraste o script para o component



Clique duas vezes em seu script e o visual studio irá abrir( ou outro programa editor de csharp que esteja em sua máquina)



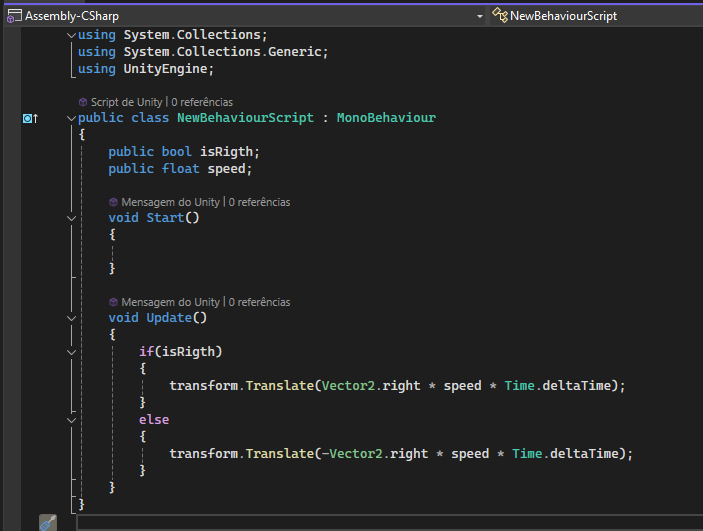
A edição do script pode ser feita até pelo bloco de notas mas para facilitar este momento use o programa que mais lhe agrade para edição de códigos. Caso não tenha seu programa predileto use o Visual Studio por ser considerado o oficial para este tipo de edições em C#.



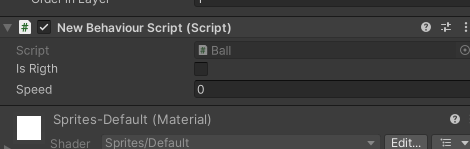
Vamos lembrar de usar o delta.Time pois ele regula a questão de uso de frames no jogo

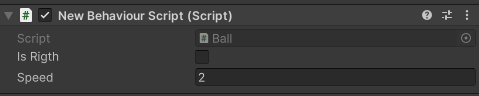
transform.Translate(Vector2.right \* speed \* Time.deltaTime);

Após concluir nosso código que fará o movimento lateral da bolinha devemos sempre salvar todas as modificações antes de retornar para Unity.

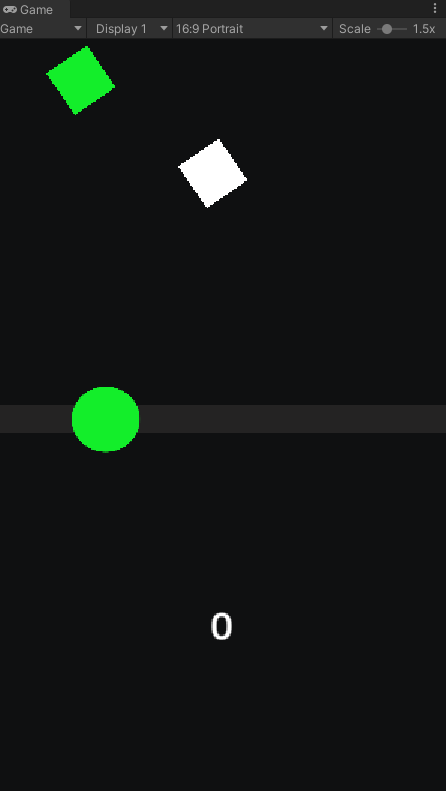


Devemos inserir alguma velocidade ao nosso objeto

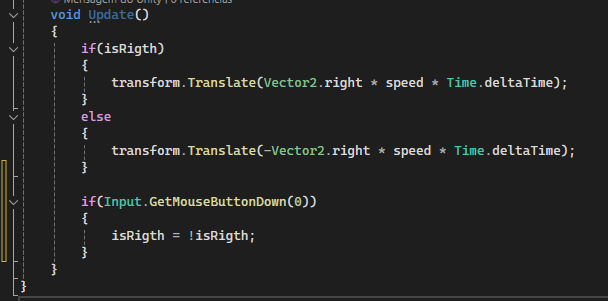




Clique em play para testar se existe o movimento



Agora que temos o movimento devemos gerar a lógica que este movimento seja acionado pelo toque em tela. Retorne ao script ball e faça as modificações a seguir.



Teste seu projeto novamente usando o play ( não se esqueça sempre de a cada modificação de código, sempre salvar para atualizar o script junto a Unity)

A partir desta aula temos um espaço de código do projeto onde tudo ficará disponível em nosso repositório no GitHub.



https://github.com/rildexter/pjd/tree/main/djm

# Referencias

1. Unity Technologies. (2020). **Unity User Manual.** Unity Technologies.
2. [C#, 2020] **Visual C# Developer Center**, Microsoft Docs.

Microsoft Corporation. (2020).

1. Geig, M. (2018). **Unity 2018 Game Development in 24 Hours**, Sams Teach Yourself. Sams Publishing.
2. Hocking, J. (2015). **Unity in Action: Multiplatform Game Development in C#.** Manning Publications.