

Variando a velocidade de movimento

Temos então a movimentação do nosso personagem! Só que ele está muito rápido, vamos corrigir isto?

Para fazer isto vamos criar uma variável pública do tipo `float`, fora de qualquer método do Unity. As variáveis públicas são muito interessantes no Unity, pois podemos editá-las no *Inspetor (Inspector)*. Crie a variável desta forma: `public float Velocidade = 10`. Assim, temos um valor inicial de velocidade de movimento, que é 10.

Lembre-se de salvar o *Script* para ver esta modificação com o atalho **Ctrl + S (CMD + S no macOS)**.

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ControlaJogador : MonoBehaviour {
6
7
8
9      // Update is called once per frame
10     void Update (){
11         float eixoX = Input.GetAxis("Horizontal");
12         float eixoZ = Input.GetAxis("Vertical");
13
14         Vector3 direcao = new Vector3(eixoX, 0, eixoZ);
15
16         transform.Translate(direcao);
17     }
18 }
19

```

Para aplicar esta velocidade de movimento temos que multiplicar a nossa `direcao` pelo valor da variável, tendo assim no nosso código o seguinte:

```

void Update ()
{
    float eixoX = Input.GetAxis("Horizontal");
    float eixoZ = Input.GetAxis("Vertical");

    Vector3 direcao = new Vector3(eixoX, 0, eixoZ);

    transform.Translate(direcao * Velocidade);
}

```

Podemos notar que nosso deslocamento está muito mais rápido. Isso acontece porque o método `Update` do Unity roda a cada *Quadro (Frame)* do nosso jogo, então ele está deslocando o valor da `direcao` multiplicada pela nossa `Velocidade` a cada *Quadro*.

Para corrigir isto, vamos utilizar uma variável que o próprio Unity nos fornece: o `Time.deltaTime`. Ela faz com que o deslocamento não seja a cada *Quadro*, mas apontará o valor de deslocamento indicado por segundo. Temos então que atualizar nosso código para:

```
void Update ()  
{  
    float eixoX = Input.GetAxis("Horizontal");  
    float eixoZ = Input.GetAxis("Vertical");  
  
    Vector3 direcao = new Vector3(eixoX, 0, eixoZ);  
  
    transform.Translate(direcao * Velocidade * Time.deltaTime);  
}
```

Lembre-se de salvar o *Script* para ver esta modificação com o atalho **Ctrl + S (CMD + S no macOS)**.

Você pode fazer download do projeto Unity desta aula [clicando aqui \(https://github.com/alura-cursos/unity-shooter/raw/master/Final_Aula1.zip\)](https://github.com/alura-cursos/unity-shooter/raw/master/Final_Aula1.zip):