

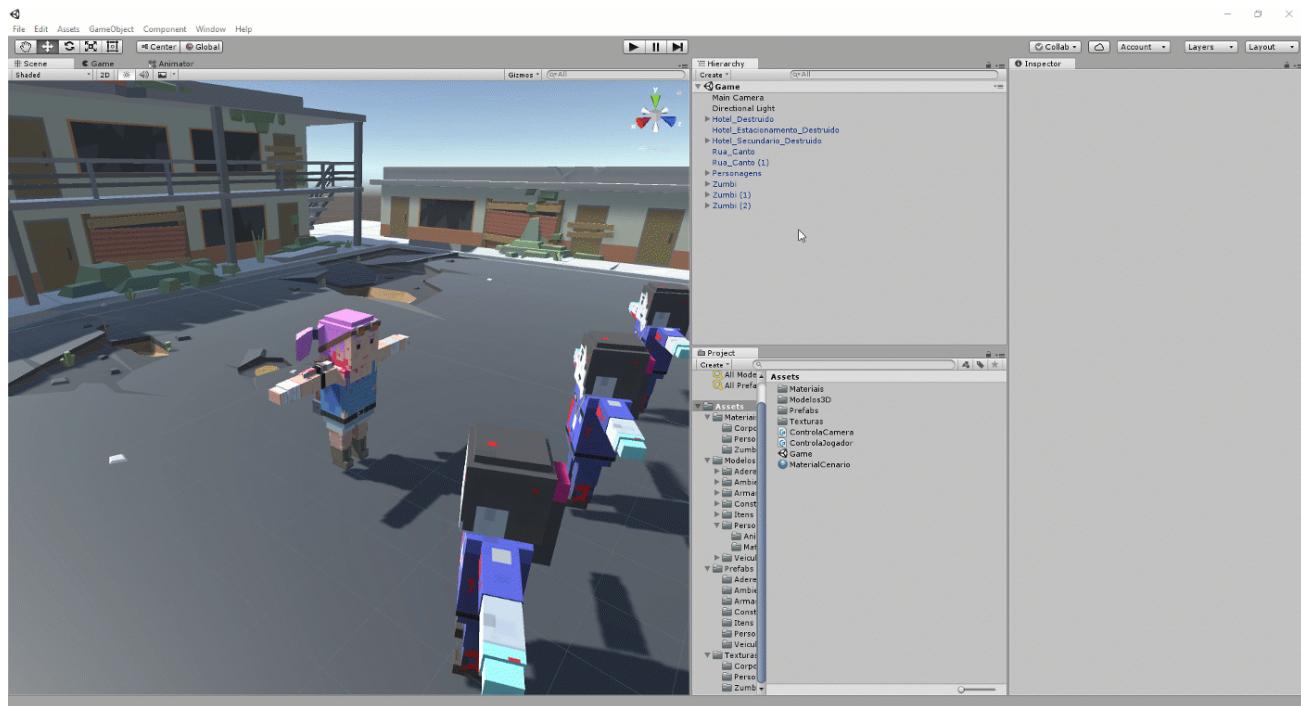
15

Movimentando pela Física

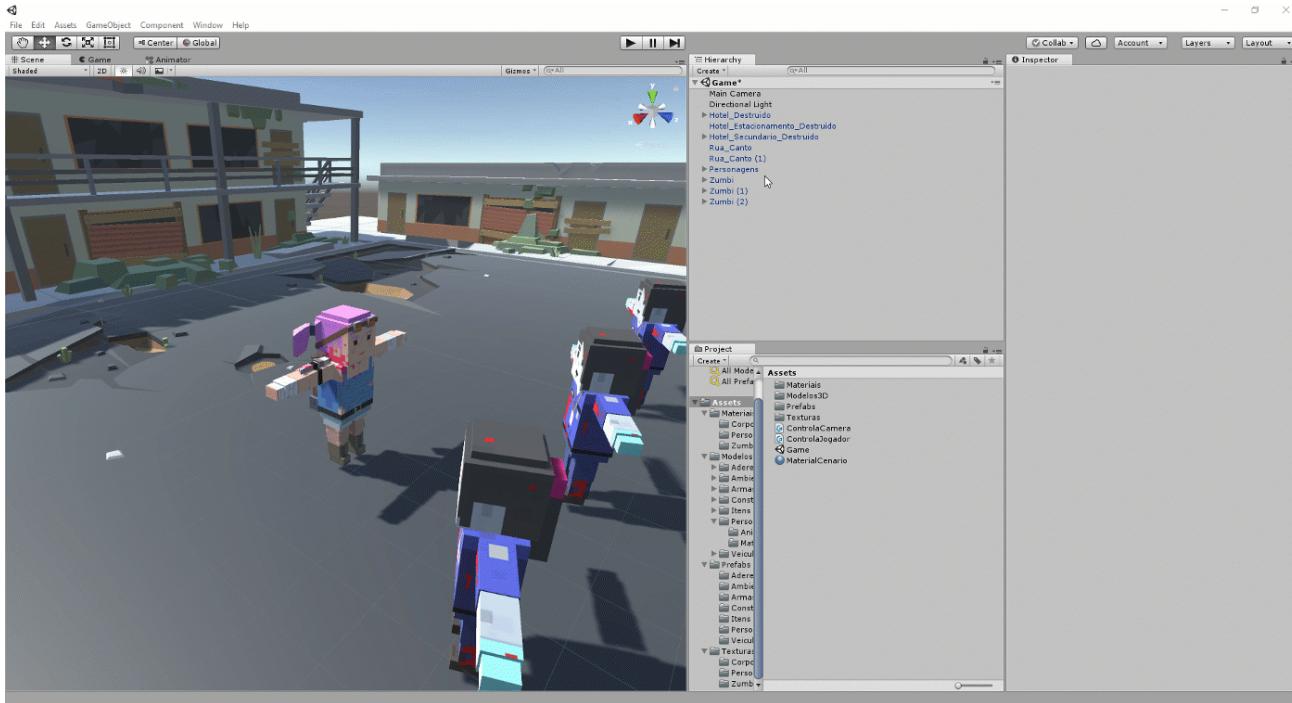
Ao testar o nosso jogo agora verificamos que o personagem ainda não está colidindo com o cenário, por que?

O Unity nos fala que para uma colisão de um objeto com outro acontecer da forma que queremos um dos objetos tem que ter um *Componente de Física*, já que o objeto que se movimenta e colide com todos os outros é o nosso jogador vamos colocar este *Componente* nele.

Selecione o **Jogador** e navegue no *Inspector* clique em **Add Component > Physics > Rigidbody** este é o *Componente* de física de corpos rígidos do Unity.



Agora que temos o **Rigidbody** no nosso personagem podemos verificar que ele já está colidindo com o cenário porém ele cai pelo cenário e rotaciona pela física, para corrigirmos este comportamento vamos no componente de **Rigidbody** abrir as opções de **Constraints** e vamos marcar para congelar a movimentação em Y com **Freeze Position Y** e todas as rotações com **Freeze Rotation**.



Quando nosso personagem anda podemos observar que ele bate nos objetos e temos algumas tremidas, isso é porque o personagem está tentando entrar para dentro do colisor e a física está jogando ele para fora.

Para corrigir isto temos que alterar nossa movimentação para que a física movimente o personagem, assim ela sabe quando ele tem que parar no colisores. Vamos então efetuar algumas mudanças no código do nosso personagem.

Vamos trocar o `transform.Translate (direcao * Velocidade * Time.deltaTime);` por outro método do Unity que utiliza o **Rigidbody**, acessando assim este componente e utilizando o método `MovePosition` para fazer a movimentação, só que aqui temos que ter a posição atual e somar o deslocamento então novamente pelo **Rigidbody** podemos ter a posição atual através do `GetComponent<Rigidbody>().position`.

Então nosso código anterior se transforma em:

```
 GetComponent<Rigidbody>().MovePosition(GetComponent<Rigidbody>().position + (direcao * Velocida
```

Só que o Unity nos informa que toda vez que queremos nos movimentar pela física temos que fazer isso pelo método `FixedUpdate`.

Então vamos criar este método e mover nossa linha de código para ele.

```

1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ControleJogador : MonoBehaviour {
6
7      public float Velocidade = 10;
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         GetComponent<Rigidbody>().MovePosition
17             (GetComponent<Rigidbody>().position + (direcao * Velocidade * Time.deltaTime));
18
19         if (direcao != Vector3.zero)
20         {
21             GetComponent<Animator>().SetBool("Movendo", true);
22         }
23         else
24         {
25             GetComponent<Animator>().SetBool("Movendo", false);
26         }
27     }
28
29     private void FixedUpdate()
30     {
31         GetComponent<Rigidbody>().MovePosition
32             (GetComponent<Rigidbody>().position + (direcao * Velocidade * Time.deltaTime));
33     }
34 }
```

Agora notamos que temos um erro na variável `direcao` porque ela está declarada no `Update` então não podemos utilizá-la em outro método então vamos mover sua declaração para o começo do *Script* para assim podermos utilizá-la em todos os lugares deste código.

```

13
14     Vector3 direcao = new Vector3(eixoX, 0, eixoZ);
15
16
17
18     if (direcao != Vector3.zero)
19     {
20         GetComponent<Animator>().SetBool("Movendo", true);
21     }
22     else
23     {
24         GetComponent<Animator>().SetBool("Movendo", false);
25     }
26
27
28     private void FixedUpdate()
29     {
30         GetComponent<Rigidbody>().MovePosition
31             (GetComponent<Rigidbody>().position + (direcao * Velocidade * Time.deltaTime));
32     }
33 }
34 
```

Você pode fazer download do projeto Unity desta aula [clicando aqui](https://github.com/alura-cursos/unity-shooter/raw/master/Final_Aula2.zip) (https://github.com/alura-cursos/unity-shooter/raw/master/Final_Aula2.zip).