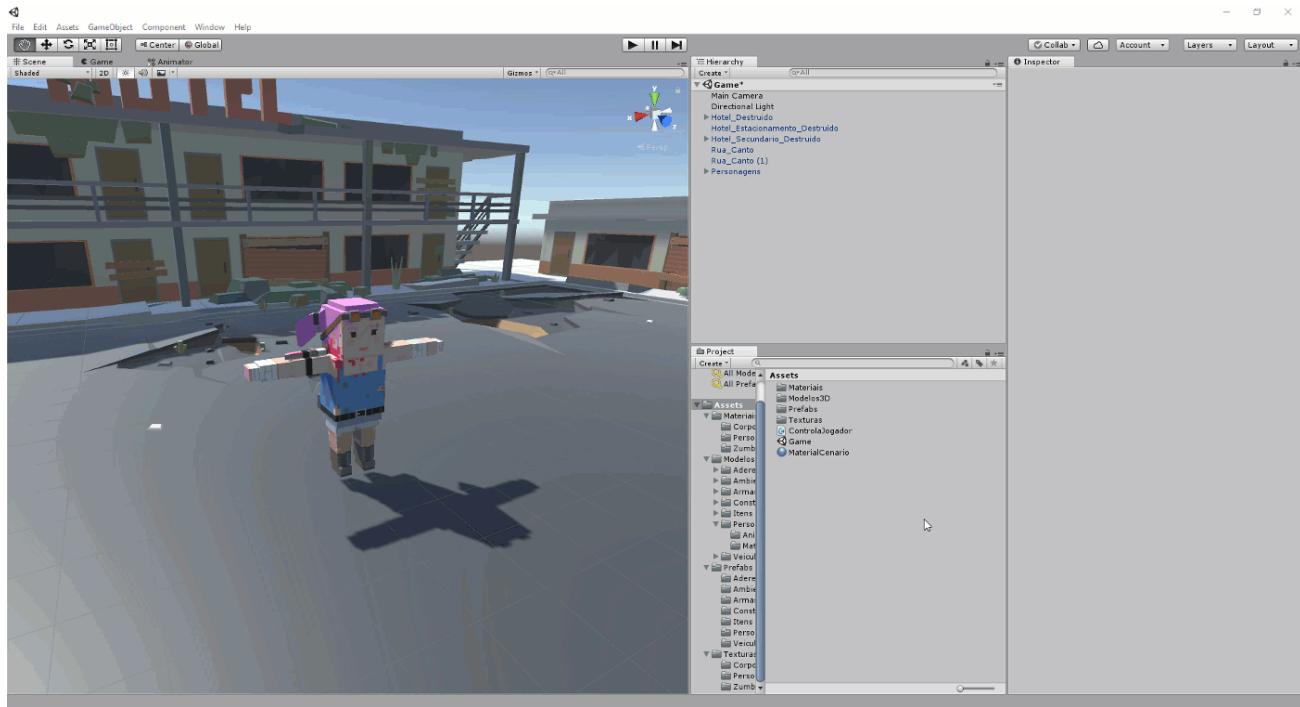


09

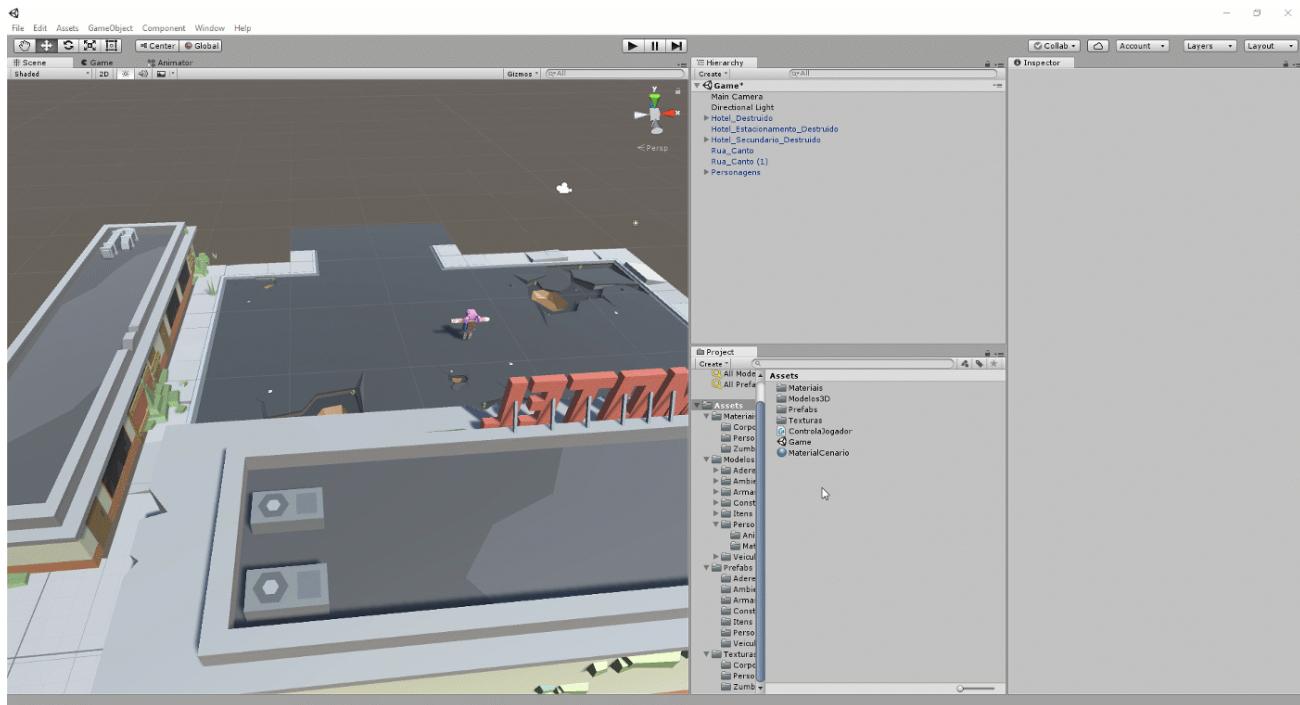
Câmera seguir Jogador

Já temos o personagem andando mas é chato a câmera não acompanha-lo e ele sair da visão do nosso jogo, isto seria até um erro no nosso jogo.

Posicione a câmera de uma forma confortável no jogo, movimentando e rotacionando 60º no eixo X.



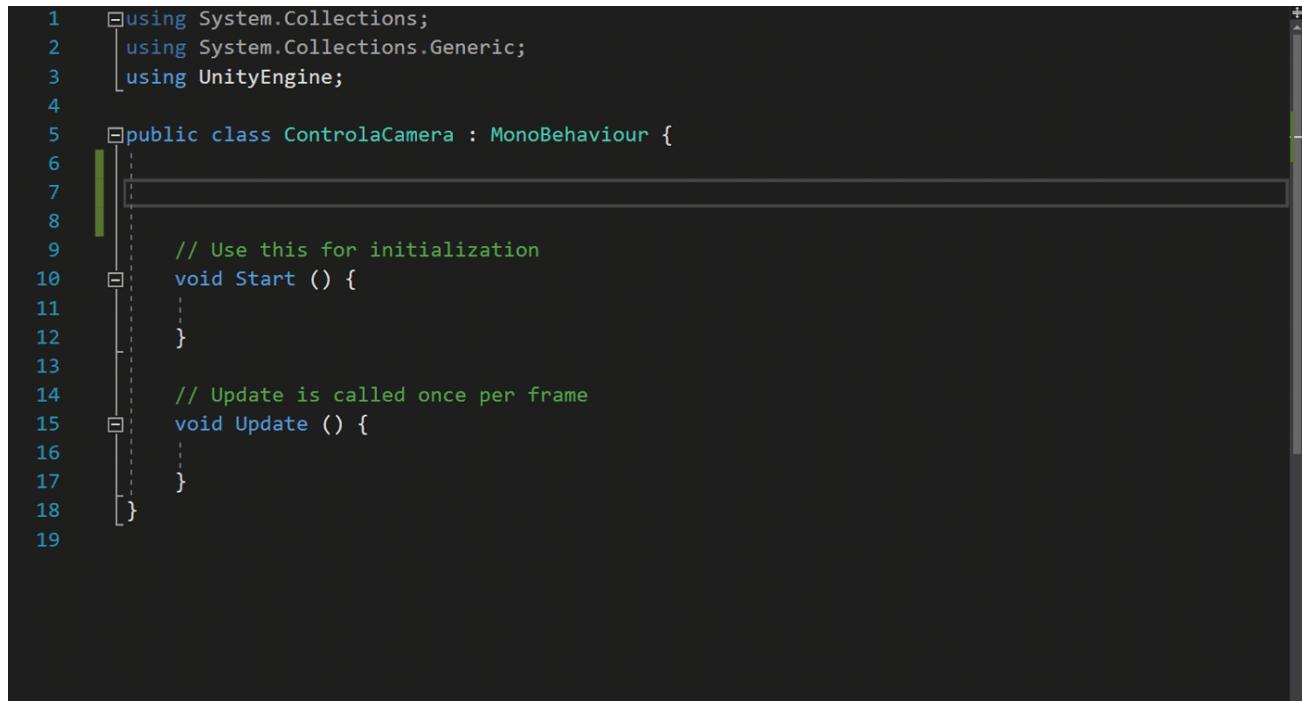
Agora que temos a câmera numa posição interessante para o nosso jogo iniciaremos a parte da movimentação criando um código. Não faz sentido utilizarmos o *Script* que controla o jogador para isso, então crie um novo código com o nome **ControlaCamera** assim evitamos que nossos códigos se misturem e fiquem confusos, lembre-se de jogar o código para sua Câmera que já vem por padrão na *Hierarquia*.



Vamos agora fazer a câmera seguir o personagem para isso temos que ter uma variável que receberia o nosso personagem. Para fazer isto vamos criar uma variável do tipo `GameObject` que pode valer qualquer objeto do jogo incluindo nosso personagem.

Digite no topo do seu script `public GameObject Jogador` assim temos a nossa variável com um nome que remete ao que ela representa, depois vamos no *Inspector* no Unity e vamos arrastar o nosso jogador para essa variável pública assim temos nosso jogador anexado a ela.

Lembre de salvar o *Script* para ver a modificação.

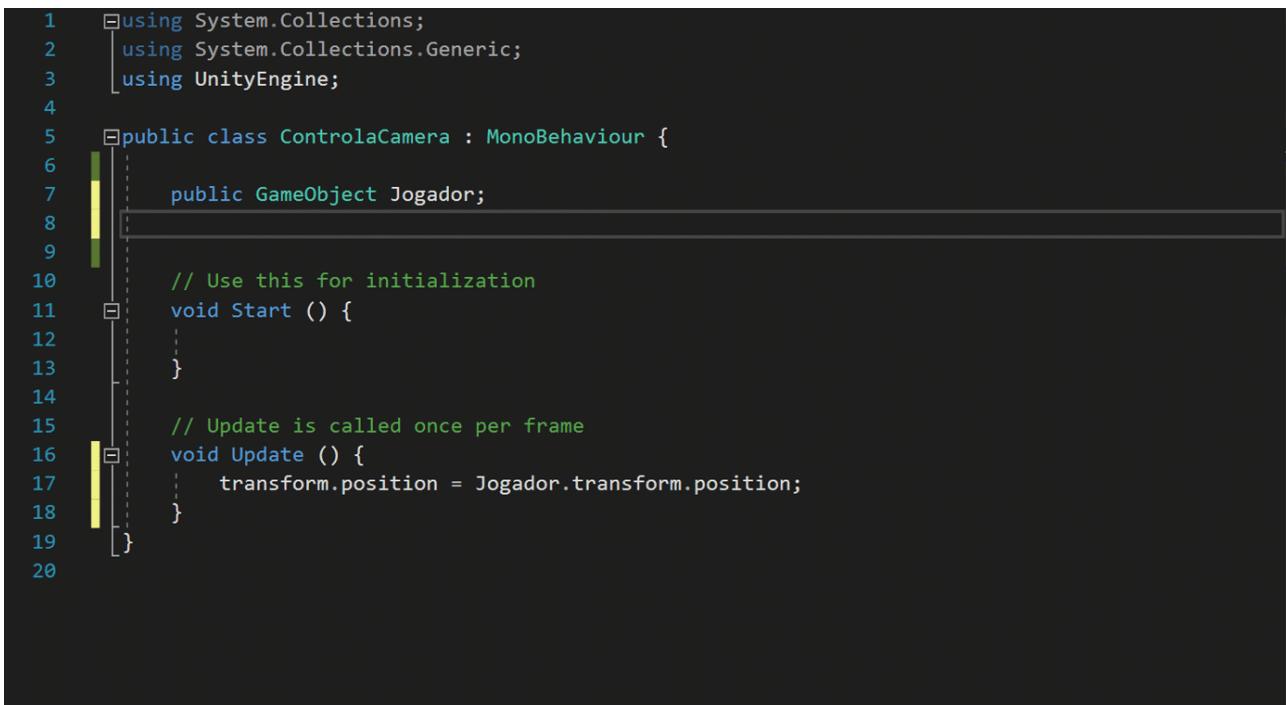


```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ControlaCamera : MonoBehaviour {
6
7
8
9  // Use this for initialization
10 void Start () {
11
12 }
13
14 // Update is called once per frame
15 void Update () {
16
17 }
18
19 }
```

Agora que há uma forma de ter o jogador no código temos que no `Update` escrever para a câmera seguir o jogador então nosso código fica assim:

```
void Update ()
{
    transform.position = Jogador.transform.position;
}
```

Ao testar o jogo nota-se que a câmera está seguindo de fato o jogador mas ela não manteve a distância que propusemos ao movimentá-la. Para manter essa distância temos que calcular o valor entre o Jogador e Câmera e usar este valor como a distância que queremos manter a Câmera para isso vamos criar uma nova variável no topo do *Script* do tipo `Vector3` já que a distância que será mantida tem um valor em X, Y e Z. Dê à variável o nome de `distCompensar`.



```
1  using System.Collections;
2  using System.Collections.Generic;
3  using UnityEngine;
4
5  public class ControlaCamera : MonoBehaviour {
6
7      public GameObject Jogador;
8
9
10     // Use this for initialization
11     void Start () {
12
13     }
14
15     // Update is called once per frame
16     void Update () {
17         transform.position = Jogador.transform.position;
18     }
19 }
20
```

Como a variável foi criada no topo do *Script* ela pode ser utilizada em qualquer lugar do código.

Para calcular a distância entre a Câmera e o Jogador faremos o cálculo no método `Start`, que roda uma vez só quando o jogo começa, já que não faz sentido calcular este valor durante o jogo todo, essa distância não irá mudar. Então escreva no método `Start` a seguinte linha:

```
void Start ()
{
    distCompensar = transform.position - Jogador.transform.position;
}
```

Pronto! Agora essa distância calculada podemos utilizar na movimentação que temos no `Update` mandando a Câmera seguir o Jogador mas levando em consideração que essa distância tem que ser mantida então somamos essa distância ao valor de movimento.

```
void Update ()
{
    transform.position = Jogador.transform.position + distCompensar;
}
```

Lembre-se de salvar o *Script*!