



O PULO

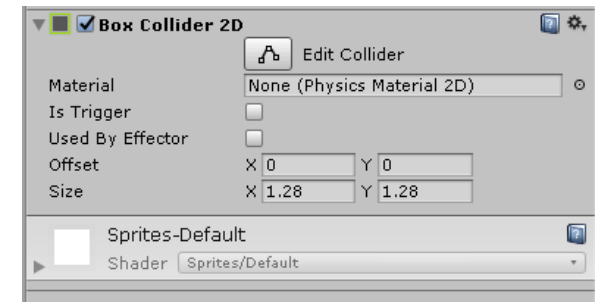
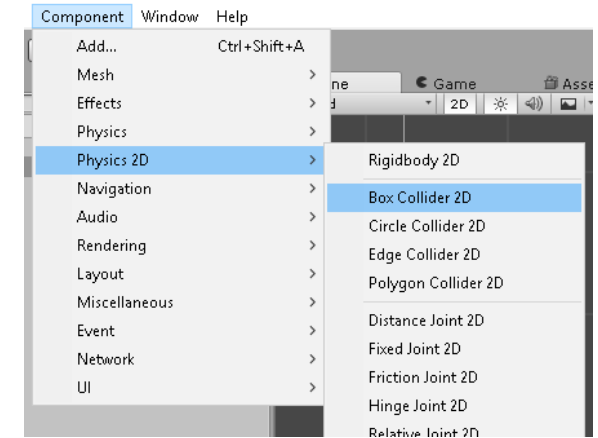
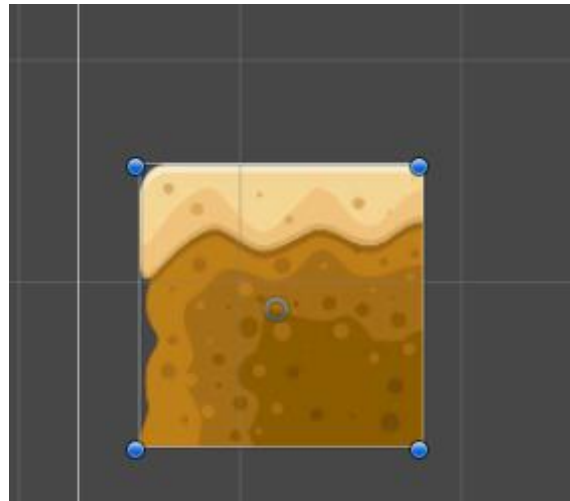
Prof. Me. Hélio Esperidião

Relembrando

- ...
- Caixa de Colisão?
- Corpo Rígido?

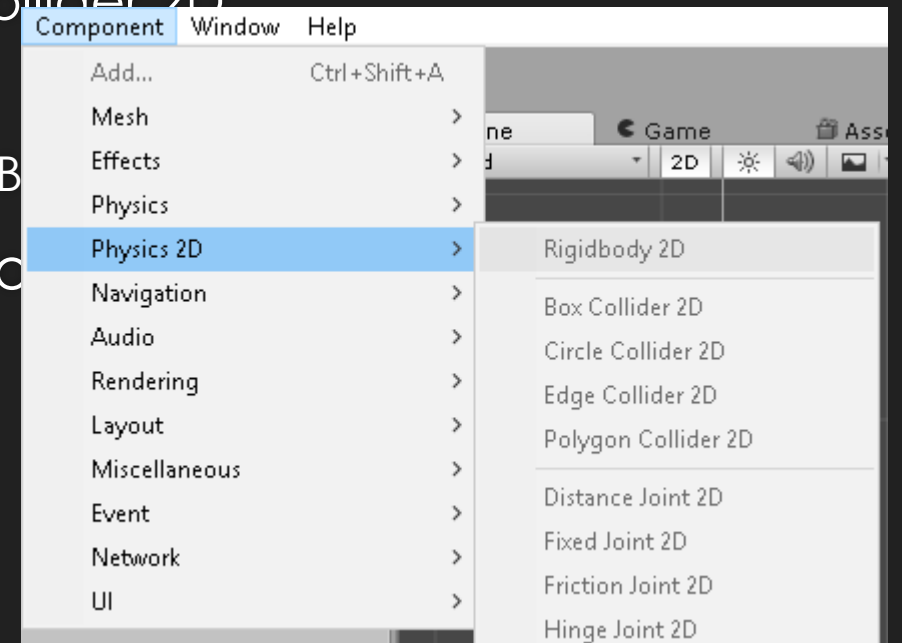
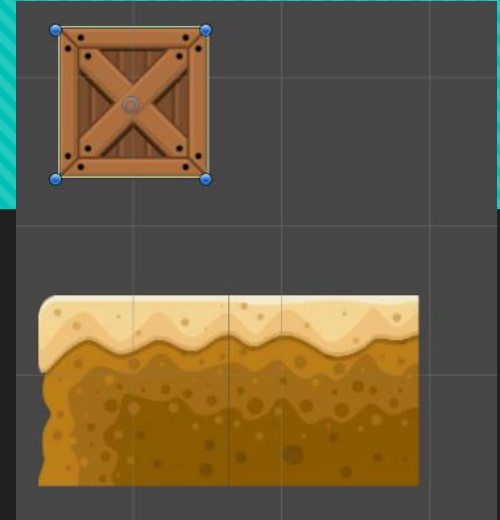
Colisão de elementos

- Acrescente o chão do cenário.
- Clique sobre o componente.
- Vá ao menu Component >> Physics2D>>Box Collider 2D
- Faça para todos os elementos onde pode ocorrer colisão.



Rigidbody 2D

- Acrescenta ao caixote
- Define-o como caixa de colisão 2D
 - Vá ao menu Component >> Physics2D >> Box Collider 2D
- Acrescente Física ao de corpo Rígido
 - Vá ao menu menu Component >> Physics2D >> Rigidbody 2D
- Como a caixa é um corpo rígido ao rodar o jogo ela vai cair se se colidir com o chão que por sinal é uma caixa de colisão.



Nome do Arquivo

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

public class Personagem : MonoBehaviour {

    //A propriedade personagemRigidbody2D faz
    //referência ao Rigidbody2D que foi adicionado ao personagem
    public Rigidbody2D personagemRigidbody2D;
    public float direcaoHorizontal=0;
    // Use this for initialization
    void Start () {
        //Inicializa o Rigidbody2D que foi acrescentado ao personagem.
        this.personagemRigidbody2D = gameObject.GetComponent<Rigidbody2D>();
        this.personagemRigidbody2D.freezeRotation = true;
    }

    // Update is called once per frame
    void Update () {
        this.direcaoHorizontal = Input.GetAxis("Horizontal");
        float intensidade = 5.0f;
        float x = this.direcaoHorizontal * intensidade;
        float y = personagemRigidbody2D.velocity.y;
        Vector2 movimento = new Vector2(x,y);
        this.personagemRigidbody2D.velocity = movimento;
    }
}
```

Variável do tipo Rigidbody2D

Variável do tipo float

Executa somente quando
O jogo é iniciado

Impede movimento rotacional

Executa a todo momento

Recebe até 1 para direita e
até -1 para esquerda.
Quando parado Recebe 0.

Determina a
intensidade do movimento

Variável do tipo Vector

gameObject é o elemento
de jogo(Asset) que recebeu o script

Recupera o Rigidbody do elemento
de jogo que recebeu o script

A variável personagemRigidbody2D recebe o
Rigidbody do elemento de jogo que recebeu
o script

y recebe a velocidade em y
do Rigidbody

Cria um novo vetor 2d
com os valores de x e y

O corpo Rígido recebe um novo vetor de
velocidade
Chamado movimento, movimento
é uma variável do tipo
vector2d

O pulo

Tudo entre as chaves { }
Só executa se o jogador
Apertou o botão de
pulo

y recebe a
intensidade do pulo

Crie um vetor 2d com os
valores
de x e y.

```
void Update () {  
    if (Input.GetButton("Jump")) {  
        float intensidadePulo = 5.0f;  
        float x = personagemRigidbody2D.velocity.x;  
        float y = intensidadePulo;  
        Vector2 movimento = new Vector2(x, y);  
        this.personagemRigidbody2D.velocity=movimento;  
    }  
}
```

Determina a intensidade do pulo

x recebe a velocidade em x
do Rigidbody, isso faz com que o
ato de pular modifique a trajetória
em x. Elemento que possui corpo
rígido

Acrescenta ao corpo rígido uma nova velocidade
Que é igual ao vetor 2d movimento.

Melhorando o código

MovimentoPulo é chamado a todo momento pois está dentro do método Update

MovimentoPulo verificará se o usuário apertou o botão de pular e acrescentará um novo vetor de velocidade ao Corpo Rígido do elemento que recebeu o script

A todo momento é chamado o método MovimentoPulo

MovimentoPulo é um método que executa o código necessário para o jogador pular.

```
void Update () {  
    MovimentoPulo(5);  
}  
void MovimentoPulo(float intensidadePulo)  
{  
    if (Input.GetButton("Jump"))  
    {  
        float x = personagemRigidbody2D.velocity.x;  
        float y = intensidadePulo;  
        Vector2 movimento = new Vector2(x, y);  
        this.personagemRigidbody2D.velocity = (movimento);  
    }  
}
```

O valor 5 será passado para variável intensidadePulo

intensidadePulo Vale 5

Código Completo

- Pulo e movimento horizontal

```
public class Personagem : MonoBehaviour {
    public Rigidbody2D personagemRigidbody2D;
    public float direcaoHorizontal=0;
    void Start () {
        this.personagemRigidbody2D = gameObject.GetComponent<Rigidbody2D>();
        this.personagemRigidbody2D.freezeRotation = true;
    }
    void Update () {
        MovimentoPulo(5);
        MovimentoHorizontal(5);
    }
    void MovimentoPulo(float intensidadePulo){
        if (Input.GetButton("Jump")){
            float x = personagemRigidbody2D.velocity.x;
            float y = intensidadePulo;
            Vector2 movimento = new Vector2(x, y);
            this.personagemRigidbody2D.velocity = (movimento);
        }
    }
    void MovimentoHorizontal(float intensidade){
        this.direcaoHorizontal = Input.GetAxis("Horizontal");
        float x = this.direcaoHorizontal * intensidade;
        float y = personagemRigidbody2D.velocity.y;
        Vector2 movimento = new Vector2(x, y);
        this.personagemRigidbody2D.velocity = movimento;
    }
}
```