

Toque indireto

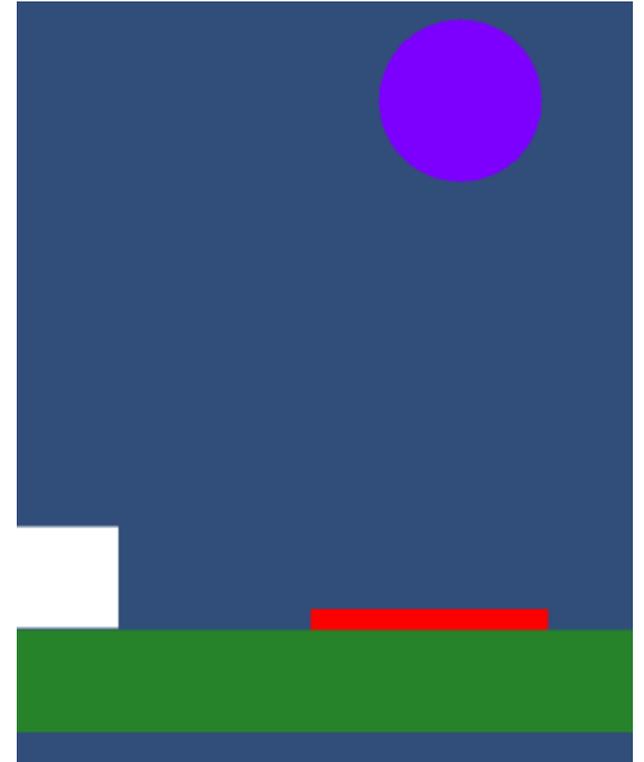
Pro. Me. Hélio Esperidião

Introdução.

- O objetivo dessa aula é permitir que o personagem toque em um objeto A e promova uma ação no Objeto B.
- Suponha que o personagem toque em uma armadilha no chão e objetos comecem a cair.

Configuração inicial.

- Círculo:
 - Physic2D>>Circle Collider 2D
 - Physic2D>>RigidBody 2D
 - Gravity Scale: 0
 - Tag: morte
 - Is Trigger: false
- Armadilha vermelha:
 - Physic2D>>Box Collider 2D
 - Tag: armadilha
 - Is Trigger: true



Atributos

```
float Vx;  
float Vy;  
float VelocidadeAndar;  
float VelocidadePular;  
float ContadorPulos;  
float TotalPulos;  
float DirecaoHorizontal;  
Rigidbody2D CorpoRigido;  
SpriteRenderer Renderer;
```

```
void Start ()
```

```
void Start () {  
    TotalPulos = 2;  
    VelocidadeAndar = 5;  
    VelocidadePular = 5;  
    CorpoRigido = GetComponent<Rigidbody2D> ();  
    Renderer = GetComponent<SpriteRenderer>();  
    CorpoRigido.freezeRotation = true;  
  
}
```

```
void Update ()
```

```
void Update () {  
    MovimentoHorizontalFlip ();  
    PuloDuplo ();  
}
```

void OnTriggerEnter2D()

```
void OnTriggerEnter2D(Collider2D objetoTriggerTocado){
    string tagTocadaTrigger = objetoTriggerTocado.gameObject.tag;
    if (tagTocadaTrigger == "armadilha01") {
        Rigidbody2D corpoRigidoBola;

        //Encontra o elemento com a tag morte e recupera o corpo rígido dele.
        corpoRigidoBola = GameObject.FindGameobjectWithTag ("morte").GetComponent<Rigidbody2D> ();
        //obtendo o corpo rígido da bola é possível modificar sua escala de gravidade
        //para um valor diferente de zero.
        corpoRigidoBola.gravityScale = 1.5f;
    }
}
```

void OnCollisionEnter2D()

```
void OnCollisionEnter2D(Collision2D objetoTocado)
{
    ContadorPulos = 0;
    string tagObjetoTocado = objetoTocado.gameObject.tag;
    if (tagObjetoTocado == "morte") {
        print ("você morreu :D");
    }
}
```

void PuloDuplo()

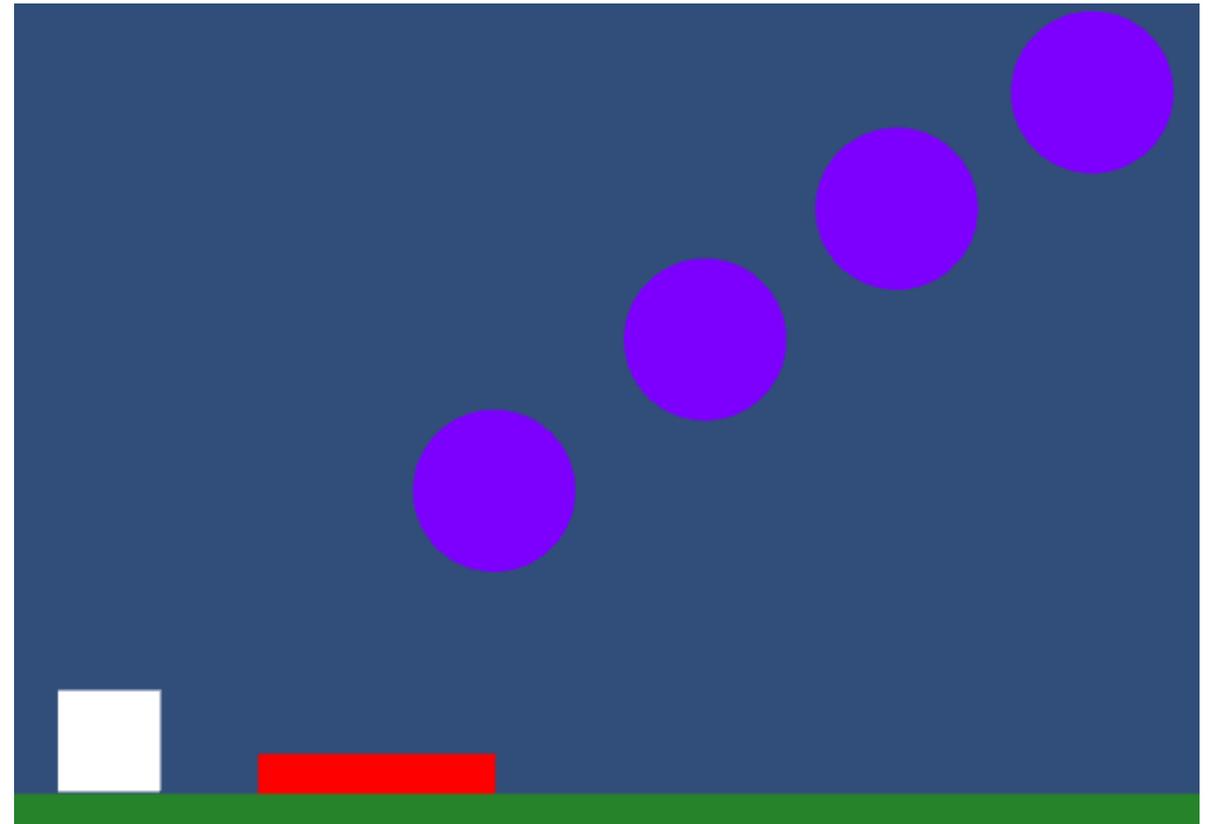
```
void PuloDuplo(){
    //verdadeiro(true) se o botão de Jump foi pressionado caso contrário igual a false
    bool apertou = Input.GetButtonDown ("Jump");
    // esse bloco só é executado se for apertado o botão de Jump
    // e se o personagem ContadorPulos < TotalPulos
    if (apertou == true && ContadorPulos < TotalPulos ) {
        ContadorPulos = ContadorPulos+1;
        Vx = CorpoRigido.velocity.x;
        Vy = VelocidadePular;
        Vector2 pulo = new Vector2 (Vx, Vy);
        CorpoRigido.velocity = pulo;
    }
}
```

void MovimentoHorizontalFlip()

```
void MovimentoHorizontalFlip(){
    DirecaoHorizontal = Input.GetAxis ("Horizontal");
    // Gera uma nova velocidade em x
    Vx = VelocidadeAndar * DirecaoHorizontal;
    //Recupera a velocidade em y que o personagem já possui
    Vy = CorpoRigido.velocity.y;
    //Cria um vetor de velocidade com os valores da velocidade em x e da velocidade em y (vx e vy)
    Vector2 andar = new Vector2 (Vx, Vy);
    // O vetor de velocidade é adicionado a velocidade do corpo rígido do personagem
    CorpoRigido.velocity = andar;
    if (DirecaoHorizontal < 0) {
        Renderer.flipX = true;
    }else if(DirecaoHorizontal > 0){
        Renderer.flipX = false;
    }
}
```

E se Existirem várias bolas?

- Círculos:
 - `Physic2D>>Circle Collider 2D`
 - `Physic2D>>RigidBody 2D`
 - Gravity Scale: 0
 - Tag: morte
 - Is Trigger:false
- Armadilha vermelha:
 - `Physic2D>>Box Collider 2D`
 - Tag: armadilha
 - Is Trigger: true



Retorna um array, "lista ou vetor" com todos
Os objetos marcados com a tag "morte"
Como configuramos quatro bolas com a tag "morte"
o vetorObjetos terá 4 posições, da posição 0 a posição 3.

```
void OnTriggerEnter2D(Collider2D objetoTriggerTocado){  
    string tagTocadaTrigger = objetoTriggerTocado.gameObject.tag;  
    if (tagTocadaTrigger == "armadilha01") {  
        GameObject[] vetorObjetos;  
        //encontra todos os elementos com a tag morte e armazena em vetorObjetos  
        vetorObjetos= GameObject.FindGameObjectsWithTag ("morte");  
        int qtdObjetos = vetorObjetos.Length;  
        for (int i = 0; i < qtdObjetos; i++) {  
            Rigidbody2D corpoRigidoBola;  
            corpoRigidoBola = vetorObjetos[i].GetComponent<Rigidbody2D> ();  
            corpoRigidoBola.gravityScale = 0.2f;  
        }  
    }  
}
```

Conta o tamanho do array

Cria uma estrutura de repetição para passar por todas
As posições do vetorObjetos

Muda a escala de gravidade para 0.2f

Recupera o rigidBody do
objeto localizado na posição
0,1,2 e 3 do array .