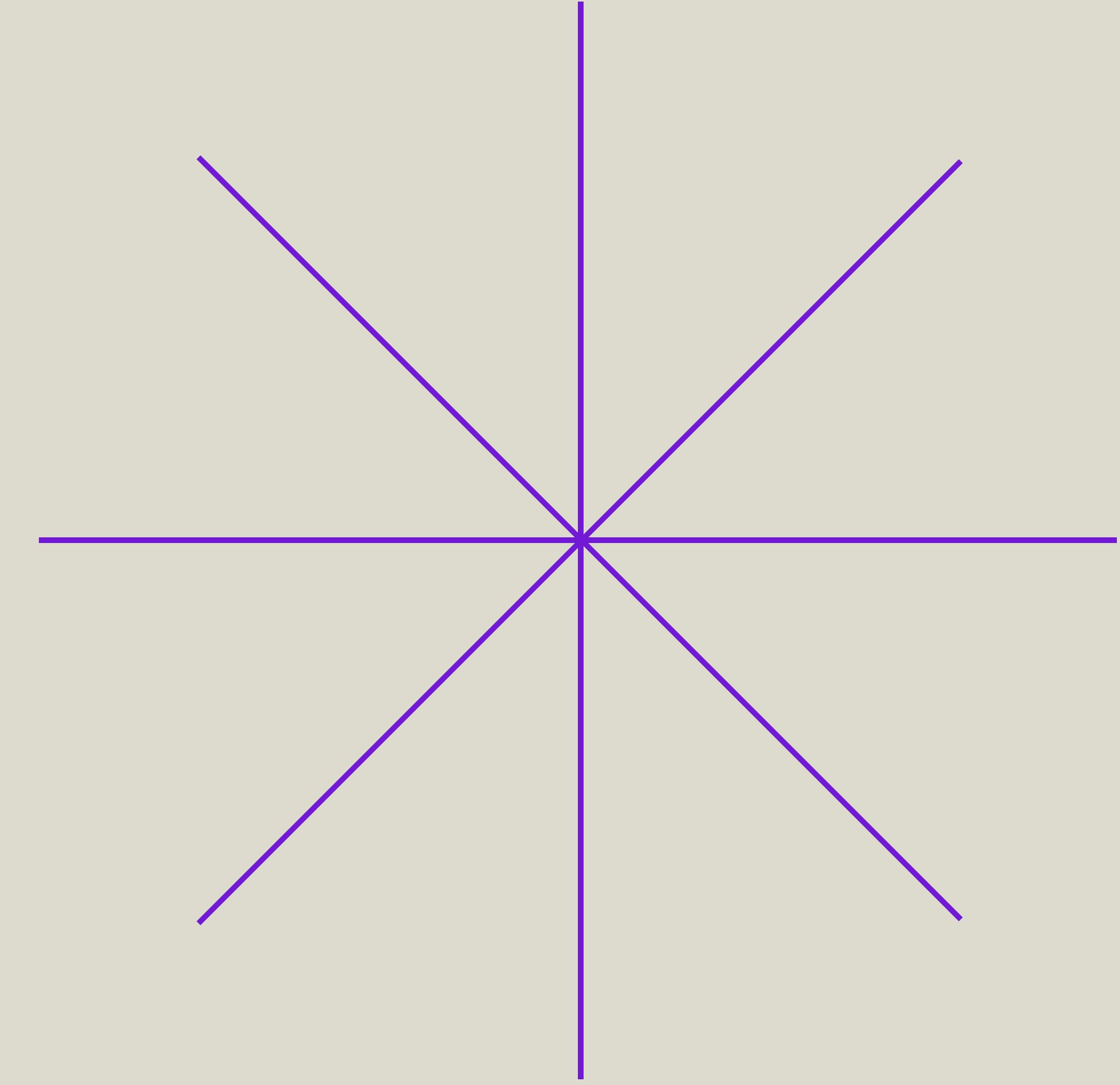
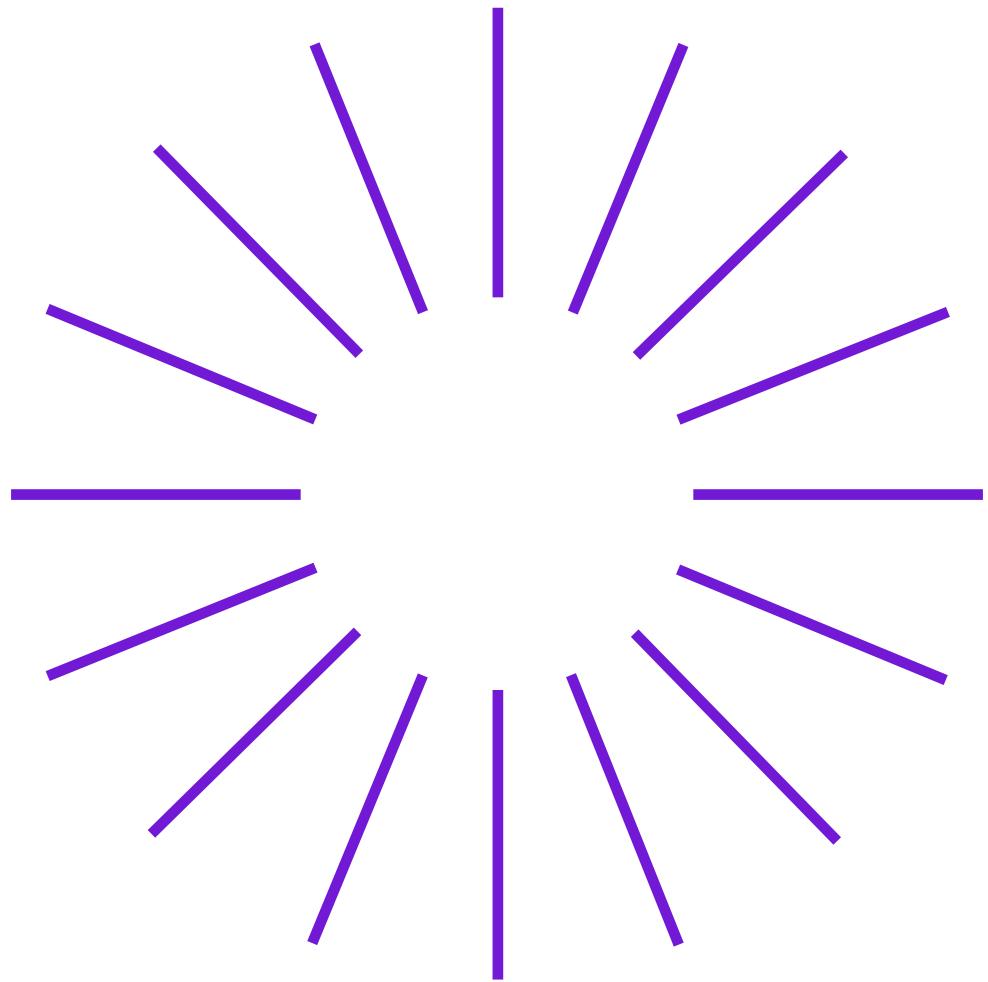


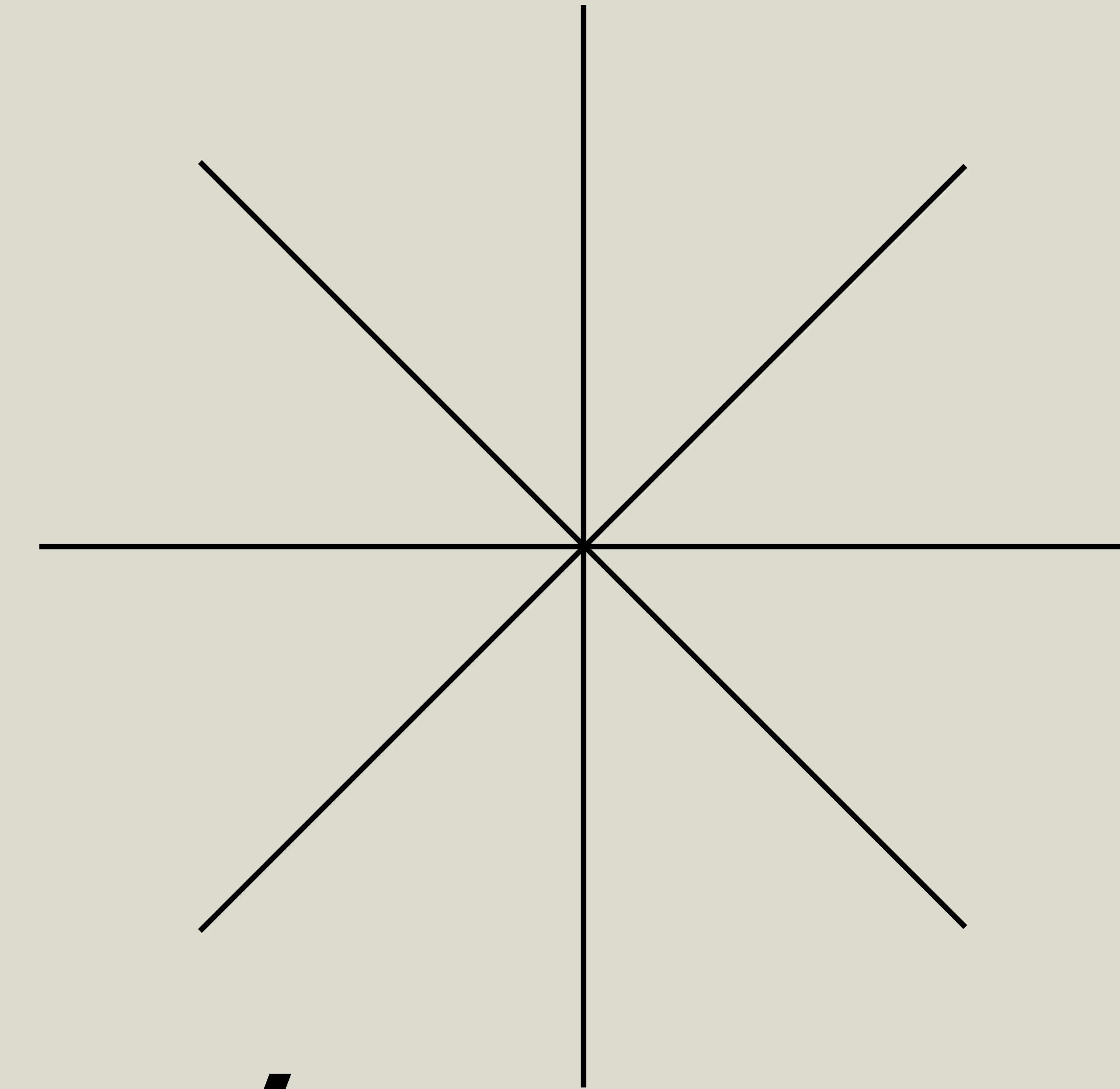
06. Space Invaders - Parte 3



ÍNDICE

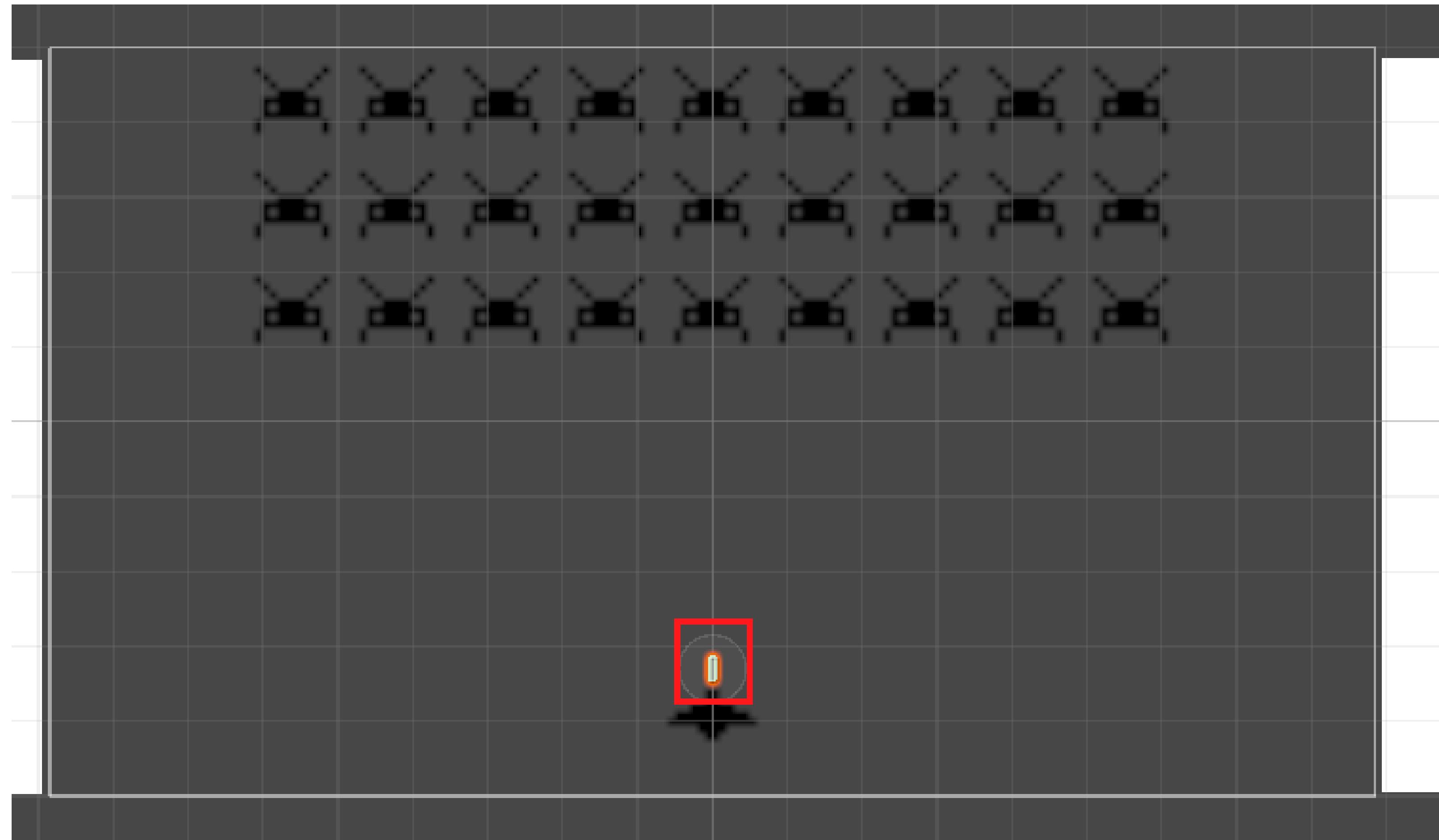
01. Criação do laser/ataque
02. Morte do Inimigo
03. Vitória
04. Morte do Jogador
05. UI do Jogo
06. Recomeçar



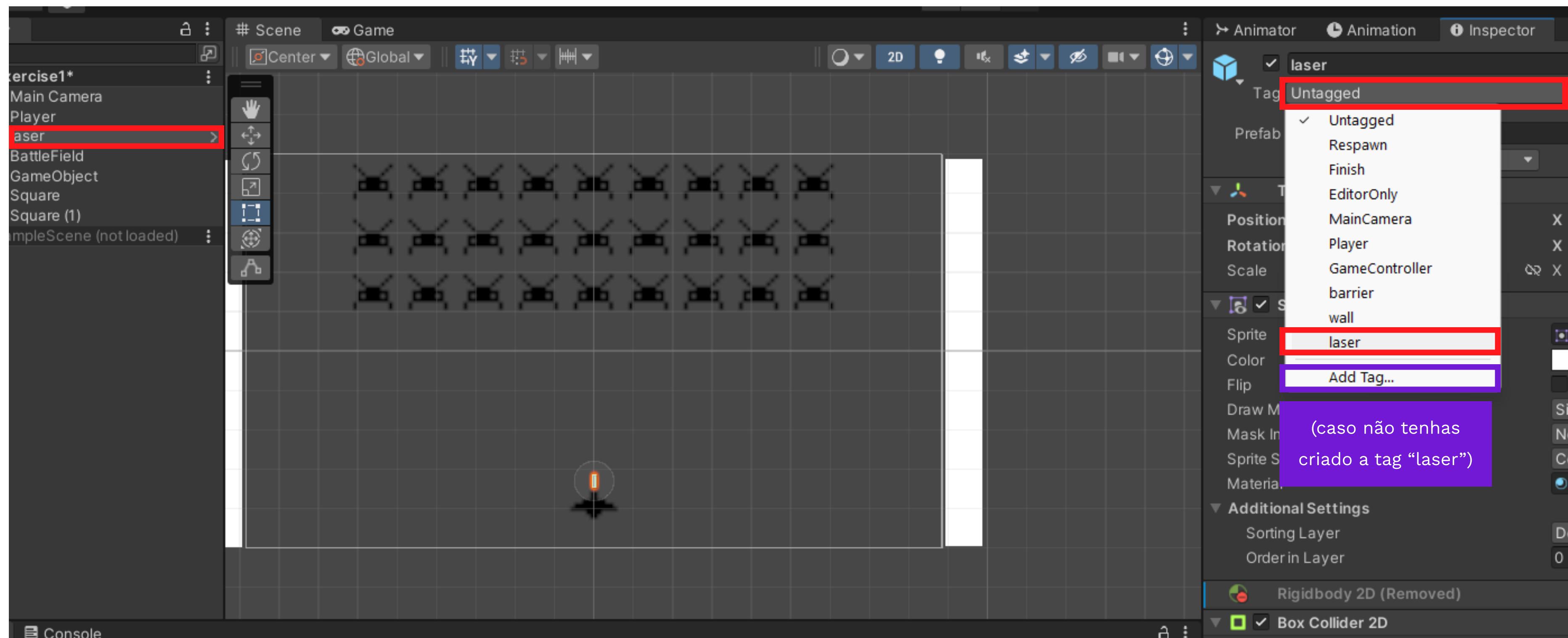


01 Criação do laser/ataque

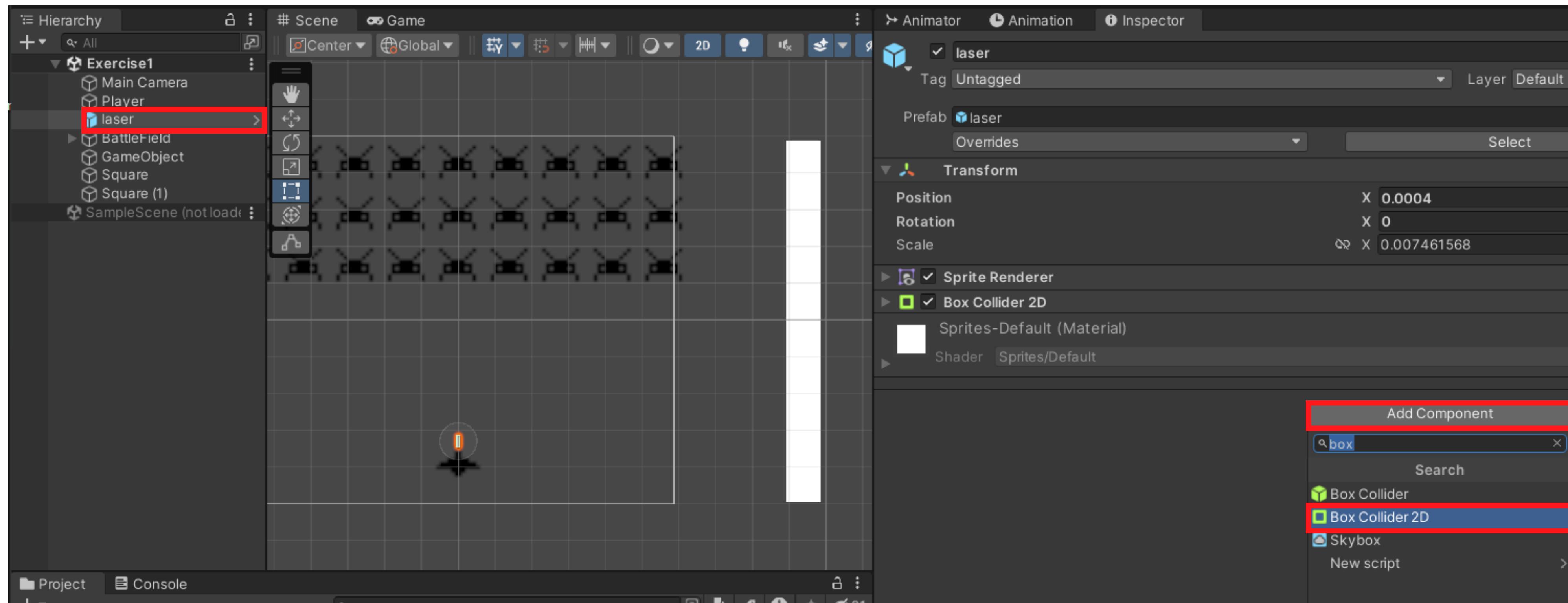
CRIA UM UM SPRITE QUADRADO PARA O LASER



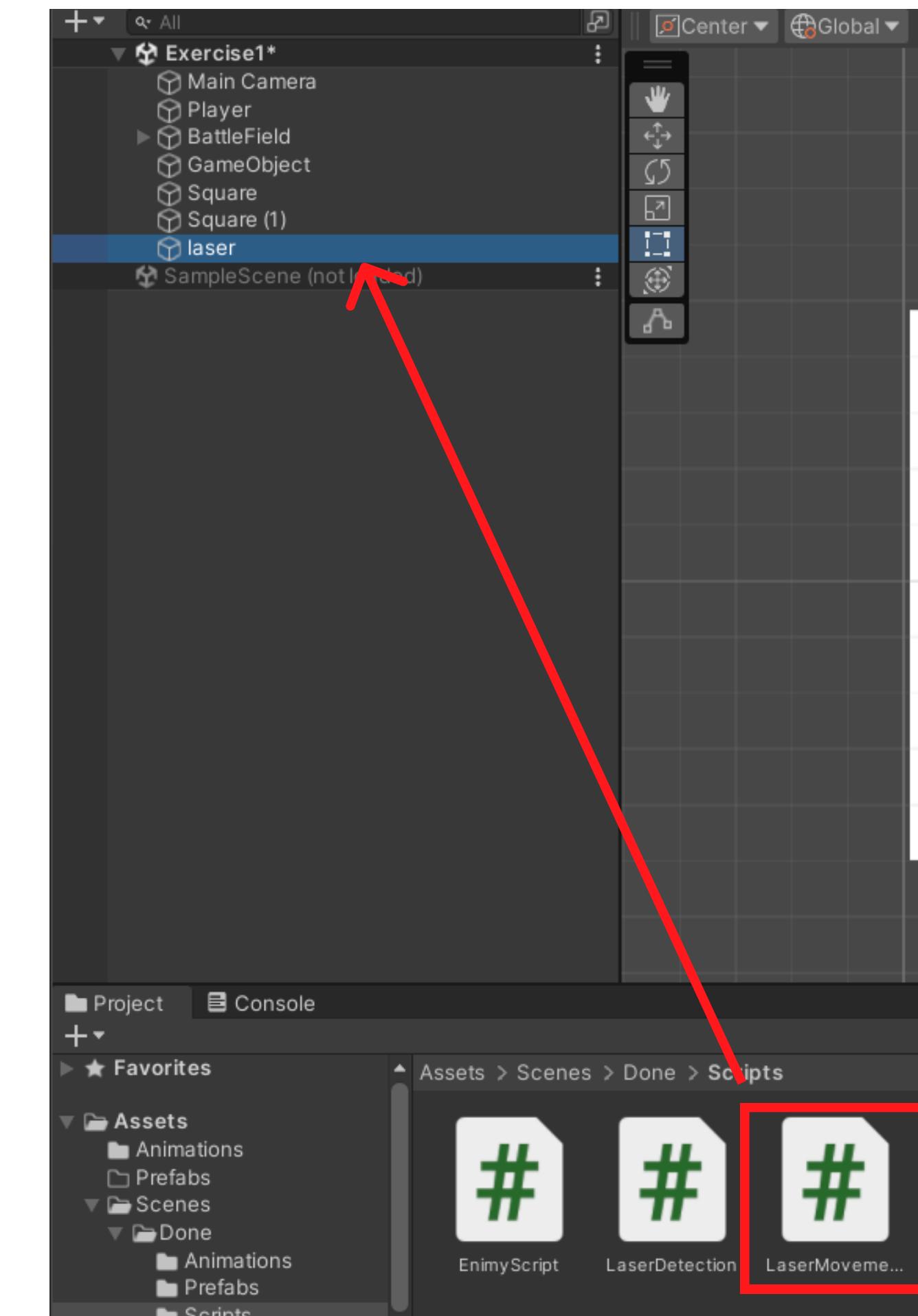
ADICIONA A TAG “LASER” AO OBJETO DO LASER



ADICIONA O COMPONENTE “BOX COLLIDER 2D”



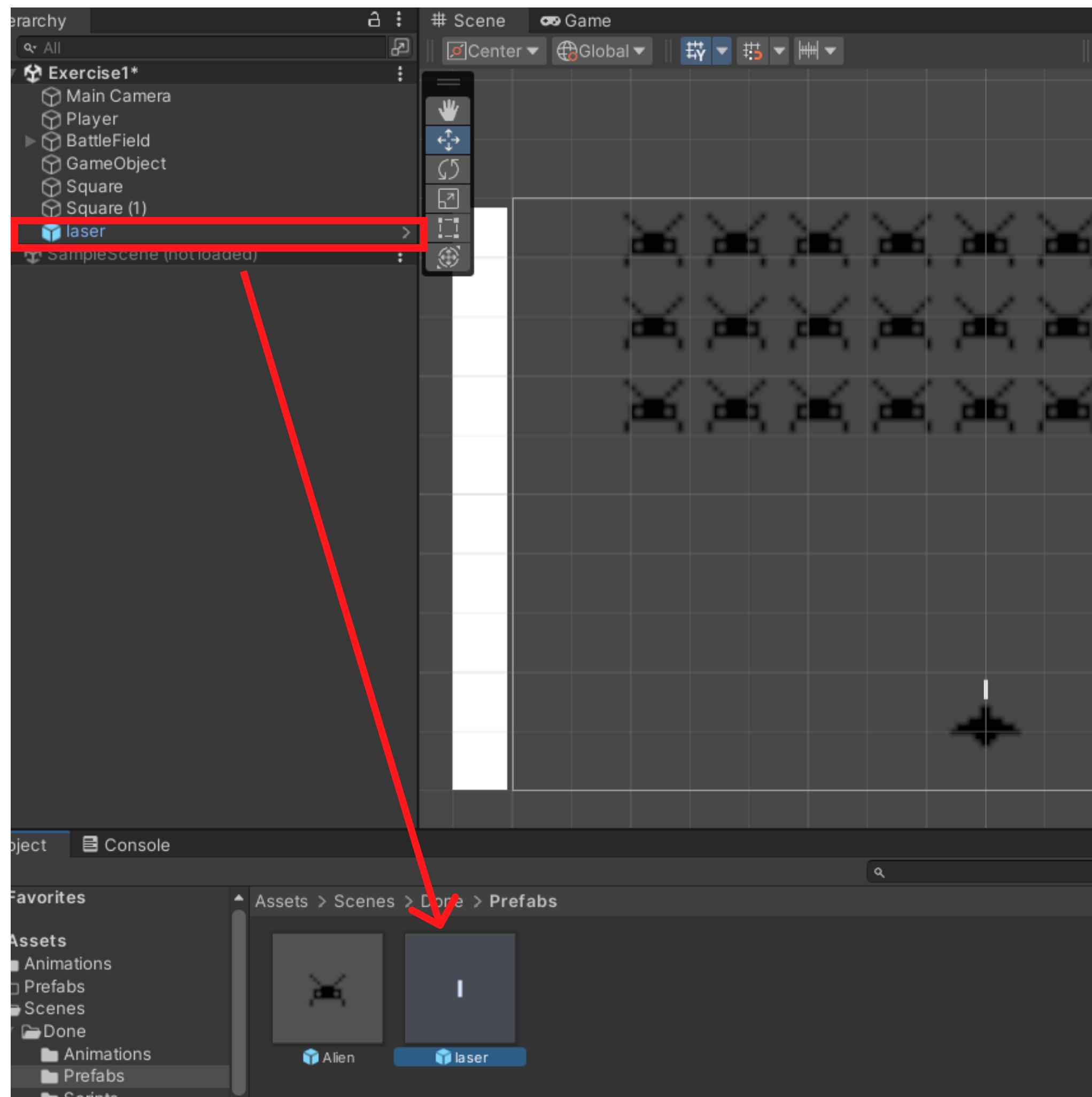
CRIA UM SCRIPT “LASERMovement” E ASSOCIA-O AO TEU LASER

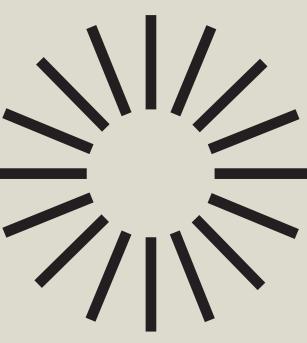


MUDA A POSIÇÃO DO LASER NO MÉTODO UPDATE (O Y DEVE AUMENTAR LENTAMENTE)

```
void Update()
{
    gameObject.transform.position =
        new Vector2(gameObject.transform.position.x, gameObject.transform.position.y + 0.01f);
}
```

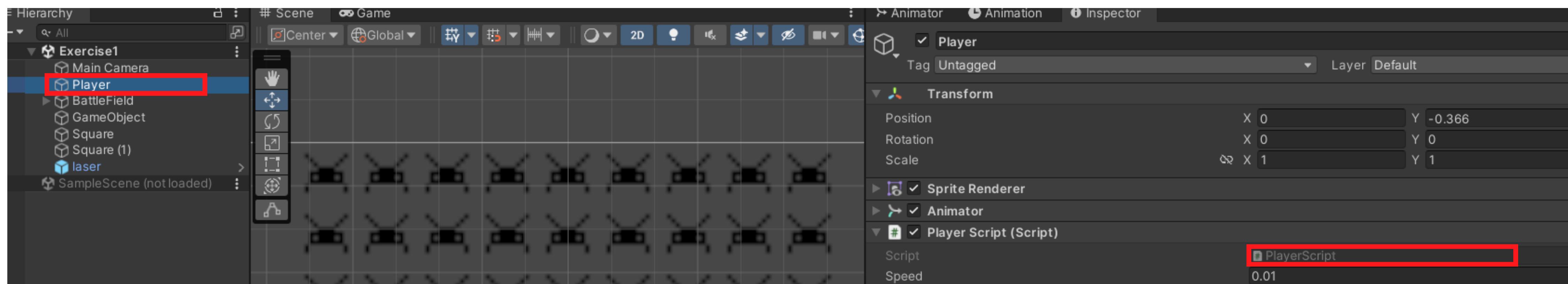
CRIA UM PREFAB BASEADO NO TEU LASER





**AGORA PODES ELIMINAR O LASER
DA TUA CENA**

ABRE O TEU SCRIPT ‘PLAYERSCRIPT’ QUE ESTÁ ASSOCIADO AO PLAYER



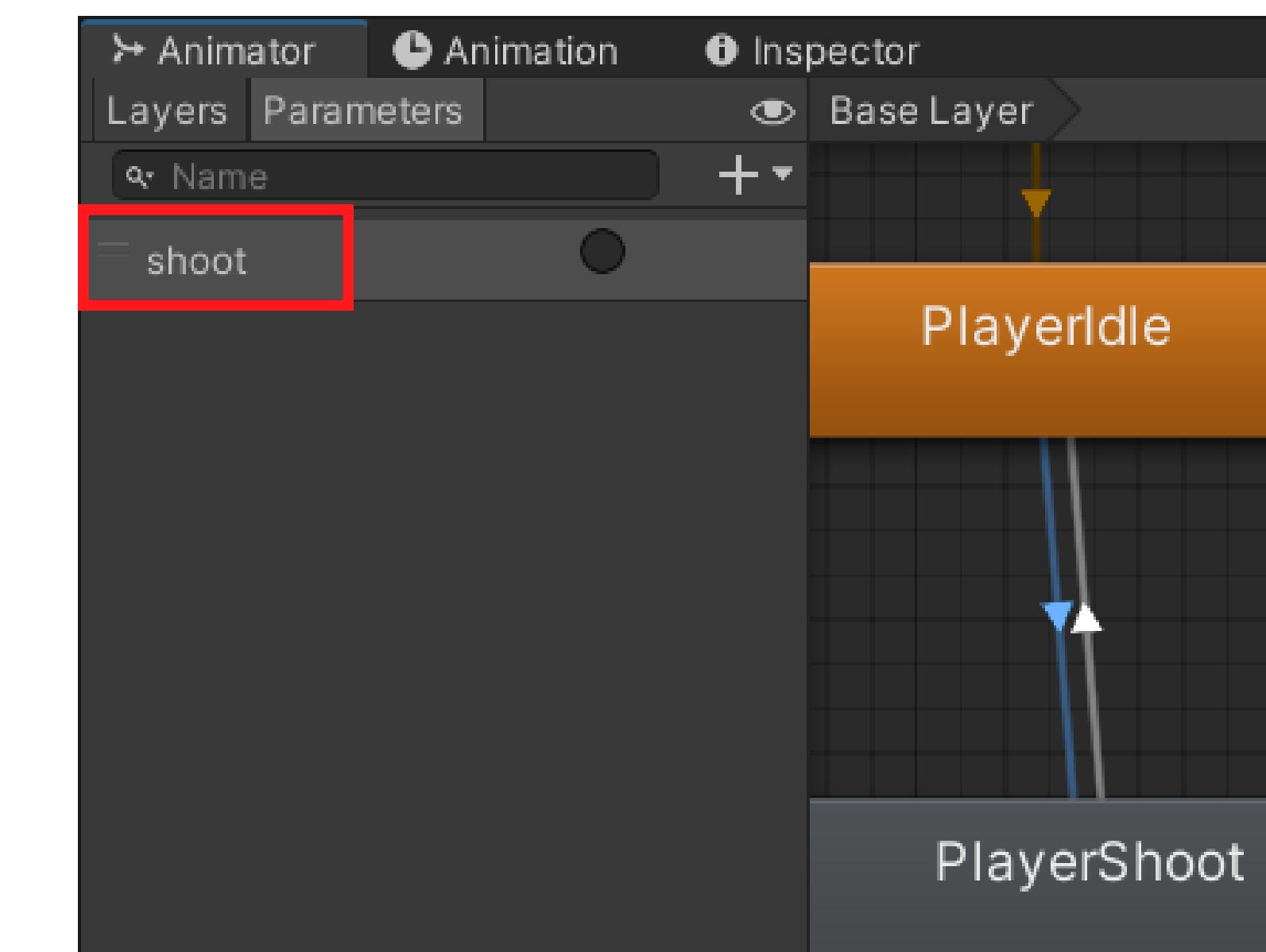
DENTRO DO MÉTODO UPDATE CRIA UMA CONDIÇÃO

```
void Update()
{
    //buscar o input "setas horizontais" do jogador
    float horizontalInput = Input.GetAxis("Horizontal");
    //muda a posicao do objeto de acordo com o input
    playerTransform.position = new Vector2(playerTran
    if (Input.GetKeyDown(KeyCode.Space))
    {
        |
    }
}
```

Esta condição
refere-se ao clique
da tecla espaço.

ACIONA O TRIGGER CORRESPONDENTE AO PARAMETRO NO CONTROLLER DO PLAYER

```
if (Input.GetKeyDown(KeyCode.Space))
{
    //aciona a animação de ataque do player
    playerAnim.SetTrigger("shoot");
}
```



CRIA UMA VARIÁVEL PARA O GAMEOBJECT DO LASER

```
.Script do Unity (1 referência de ativo) | 0 referências
public class PlayerScript : MonoBehaviour
{
    //inicializar as variaveis
    Transform playerTransform;
    [SerializeField] float speed = 0.01f;
    Animator playerAnim;

    [SerializeField] GameObject laserBeam;

    // Start is called before the first frame update
    Mensagem do Unity | 0 referências
    void Start()
```

INSTANCIA UM NOVO LASER

```
if (Input.GetKeyDown(KeyCode.Space))
{
    //aciona a animação de ataque do player
    playerAnim.SetTrigger("shoot");
    //vai ser instanciado um novo laser/bala
    Instantiate(laserBeam, playerTransform.position, Quaternion.identity);
}
```

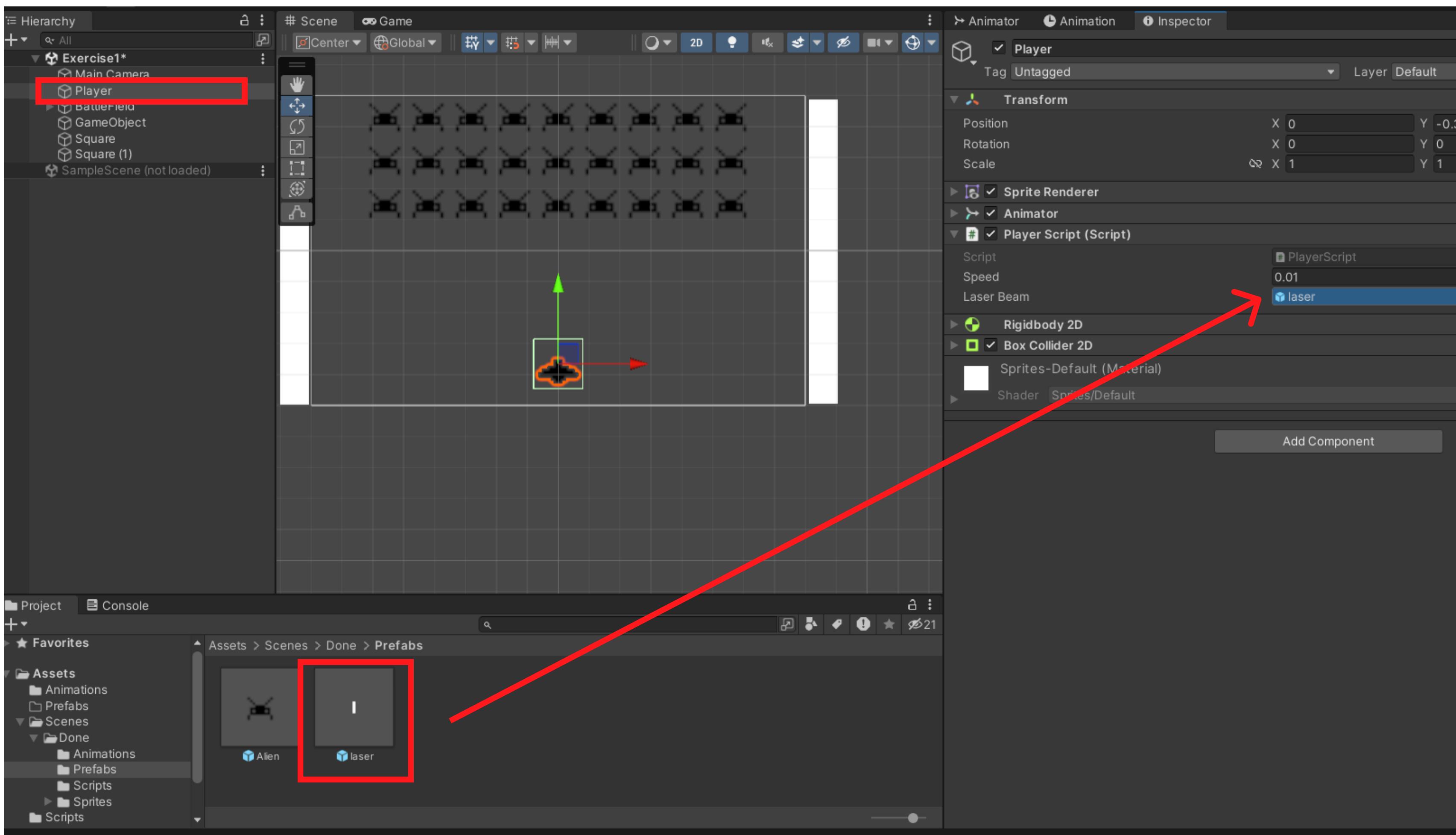
O método “Instantiate” é responsável por instanciar objetos e precisa de pelo menos 3 parâmetros (o objeto a instanciar; o “pai” do objeto, ou seja, onde na cena o instanciar; a rotação no mundo, apenas “Quaternion.identity”)

DENTRO DO MÉTODO UPDATE CRIA UMA CONDIÇÃO

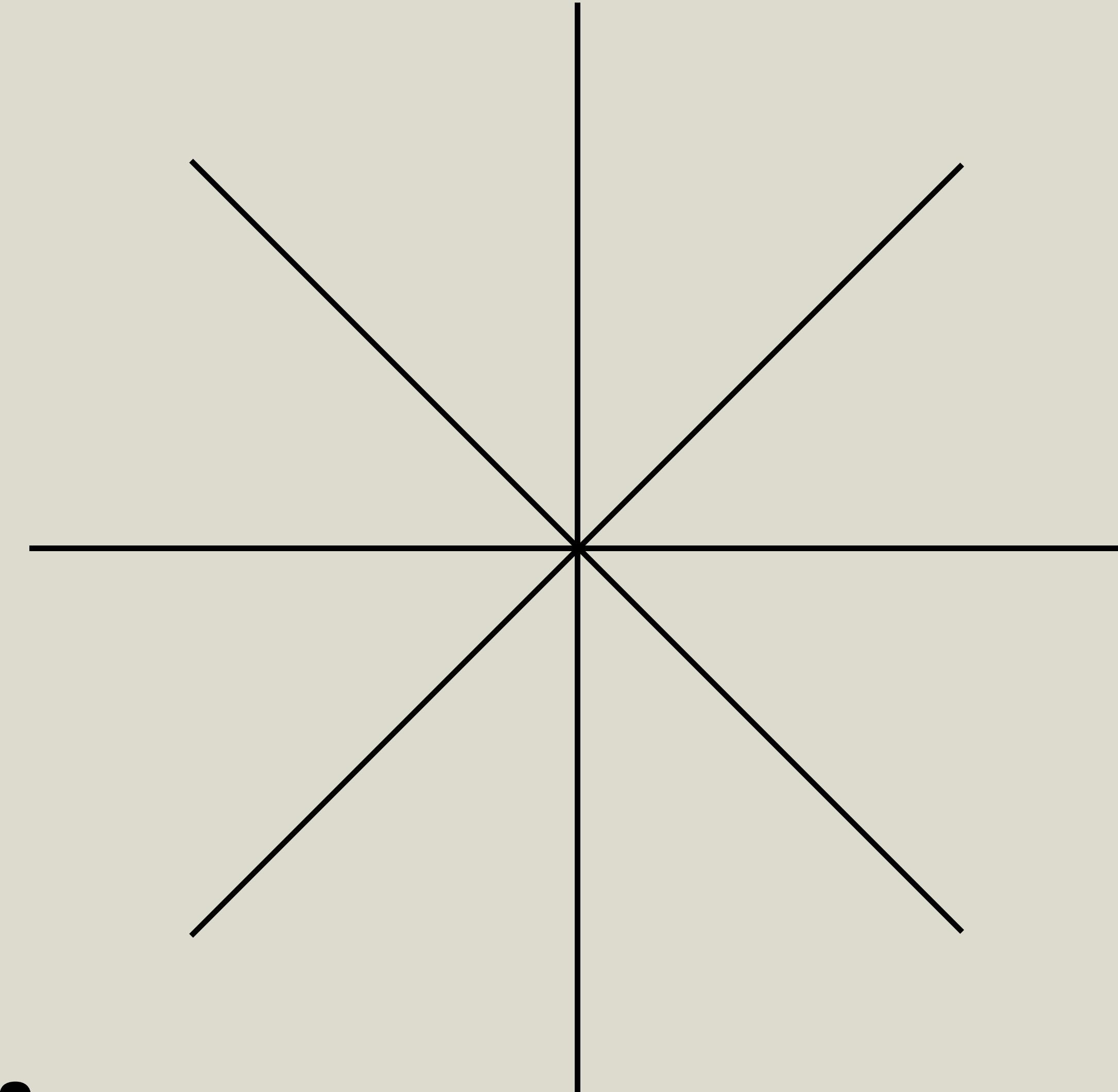
```
void Update()
{
    //buscar o input "setas horizontais" do jogador
    float horizontalInput = Input.GetAxis("Horizontal");
    //muda a posicao do objeto de acordo com o input
    playerTransform.position = new Vector2(playerTran
    if (Input.GetKeyDown(KeyCode.Space))
    {
        //código para saltar
    }
}
```

Esta condição
refere-se ao clique
da tecla espaço.

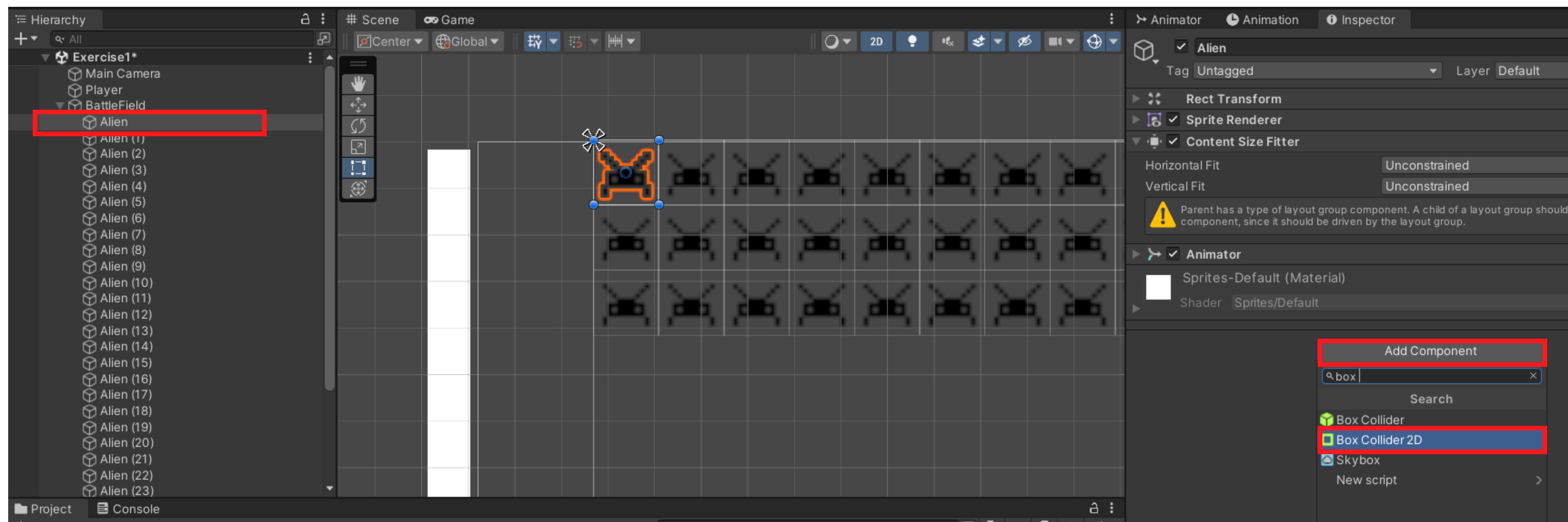
ATRIBUI O PREFAB DO LASER À VARIÁVEL



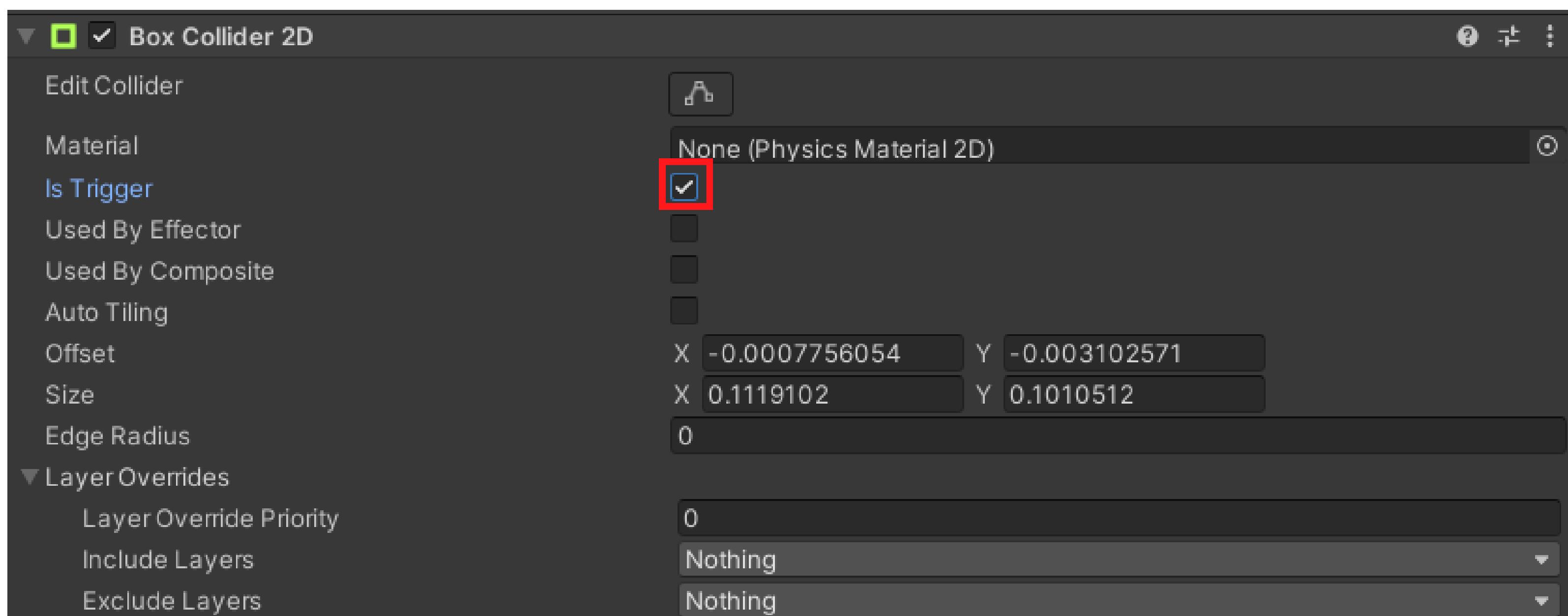
02 Morte do inimigo



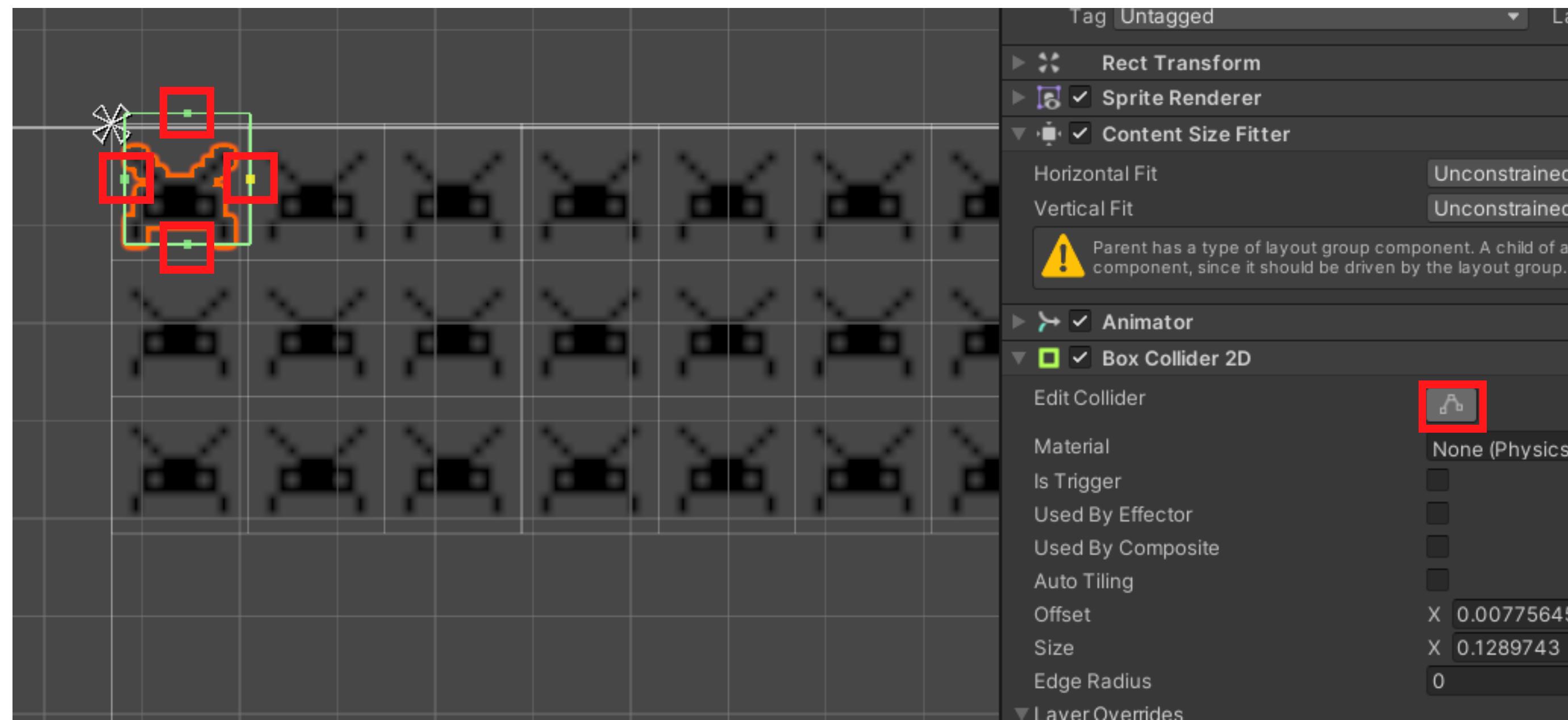
ADICIONA O COMPONENTE “BOX COLLIDER 2D” AO TEU PRIMEIRO INIMIGO



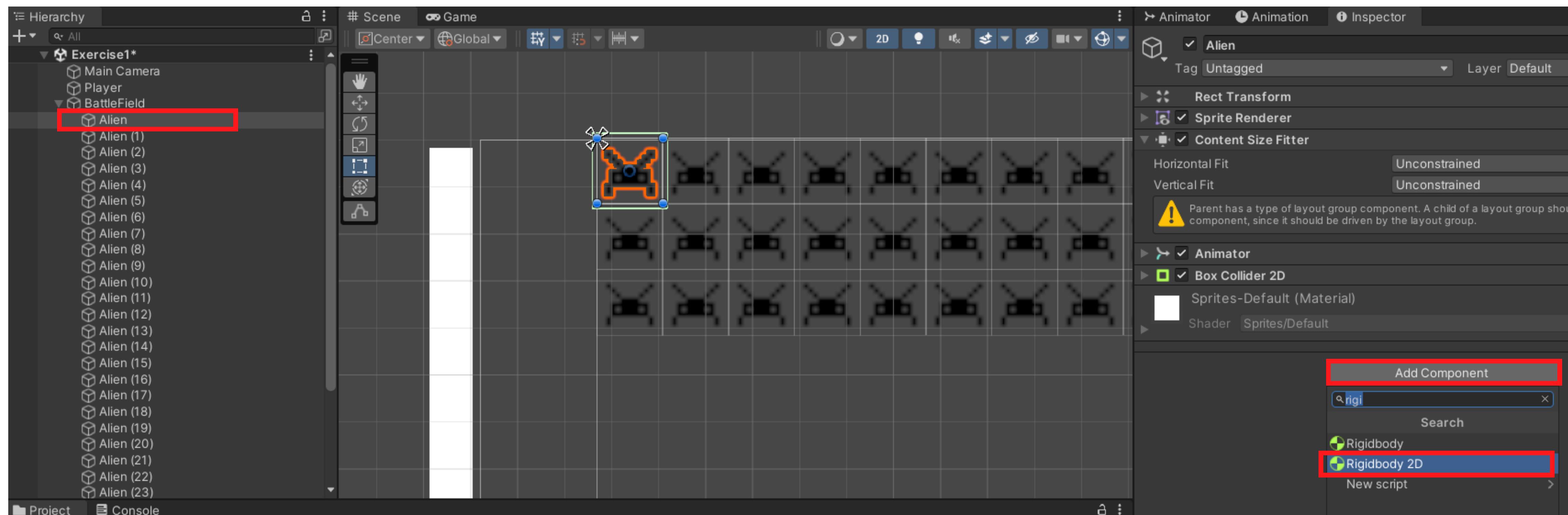
ACIONA A OPÇÃO “ISTRIGGER” DO BOX COLLIDER 2D DO TEU PRIMEIRO INIMIGO



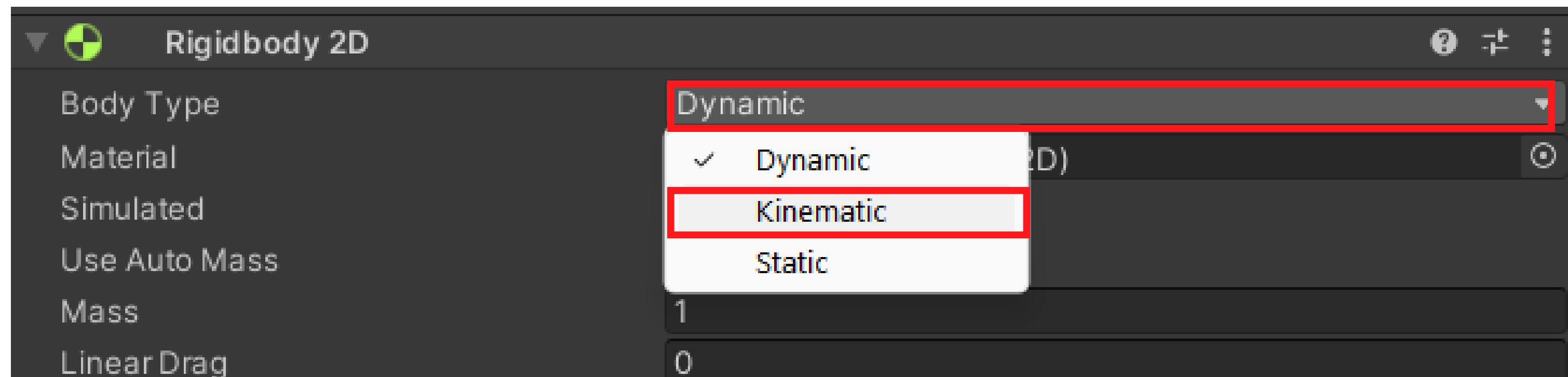
EDITA A SUPERFÍCIE DE COLISÃO



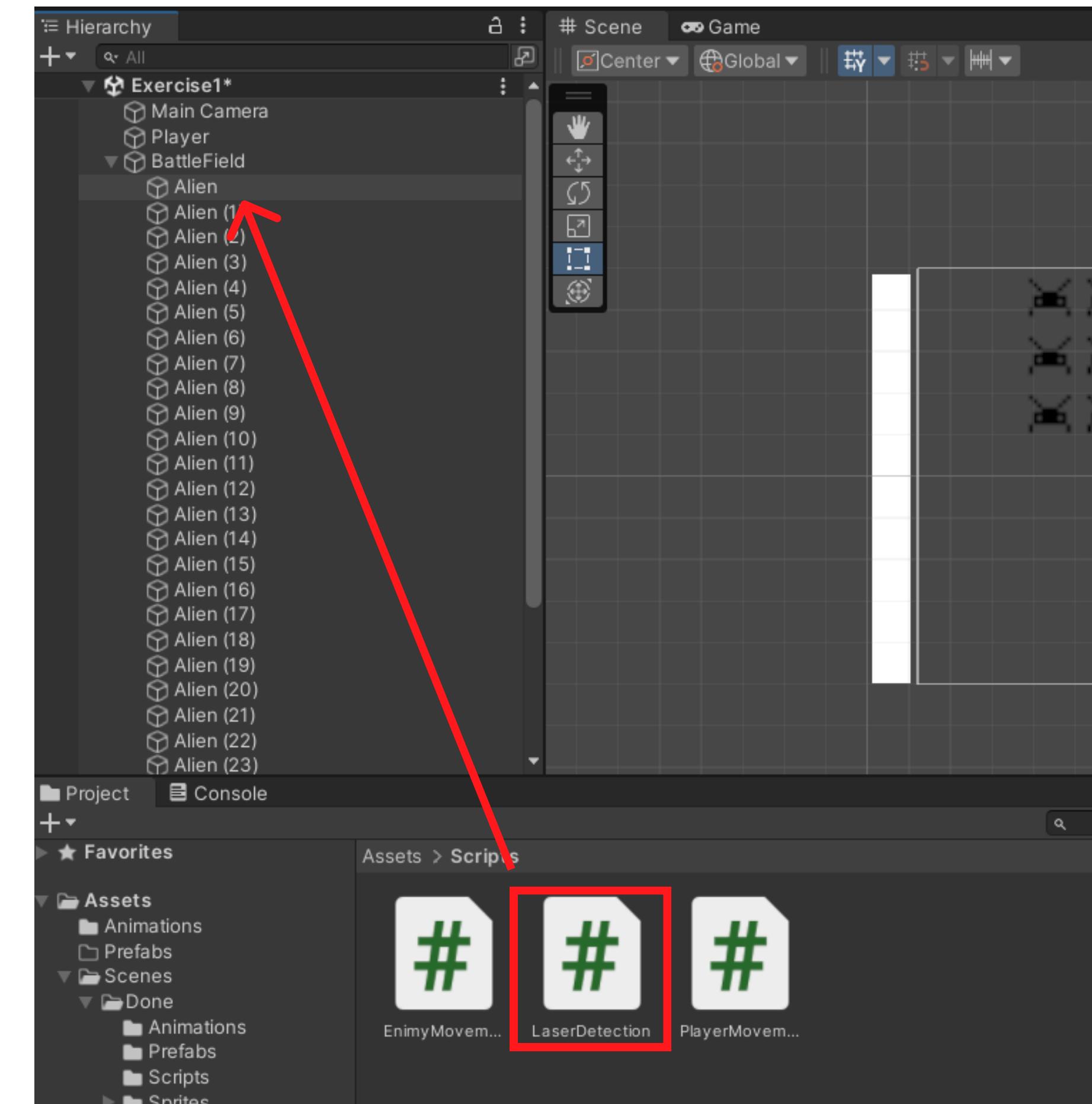
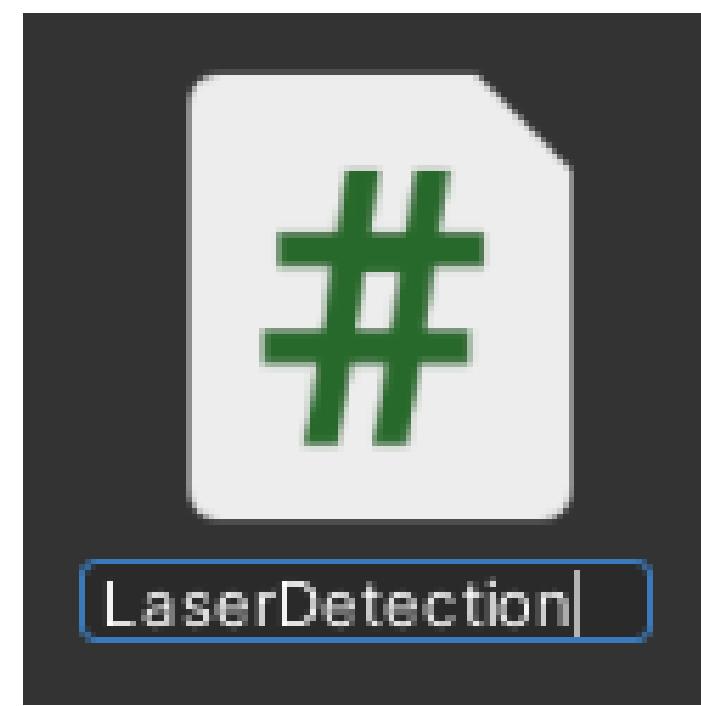
ADICIONA O COMPONENTE “RIGID BODY 2D” AO TEU PRIMEIRO INIMIGO



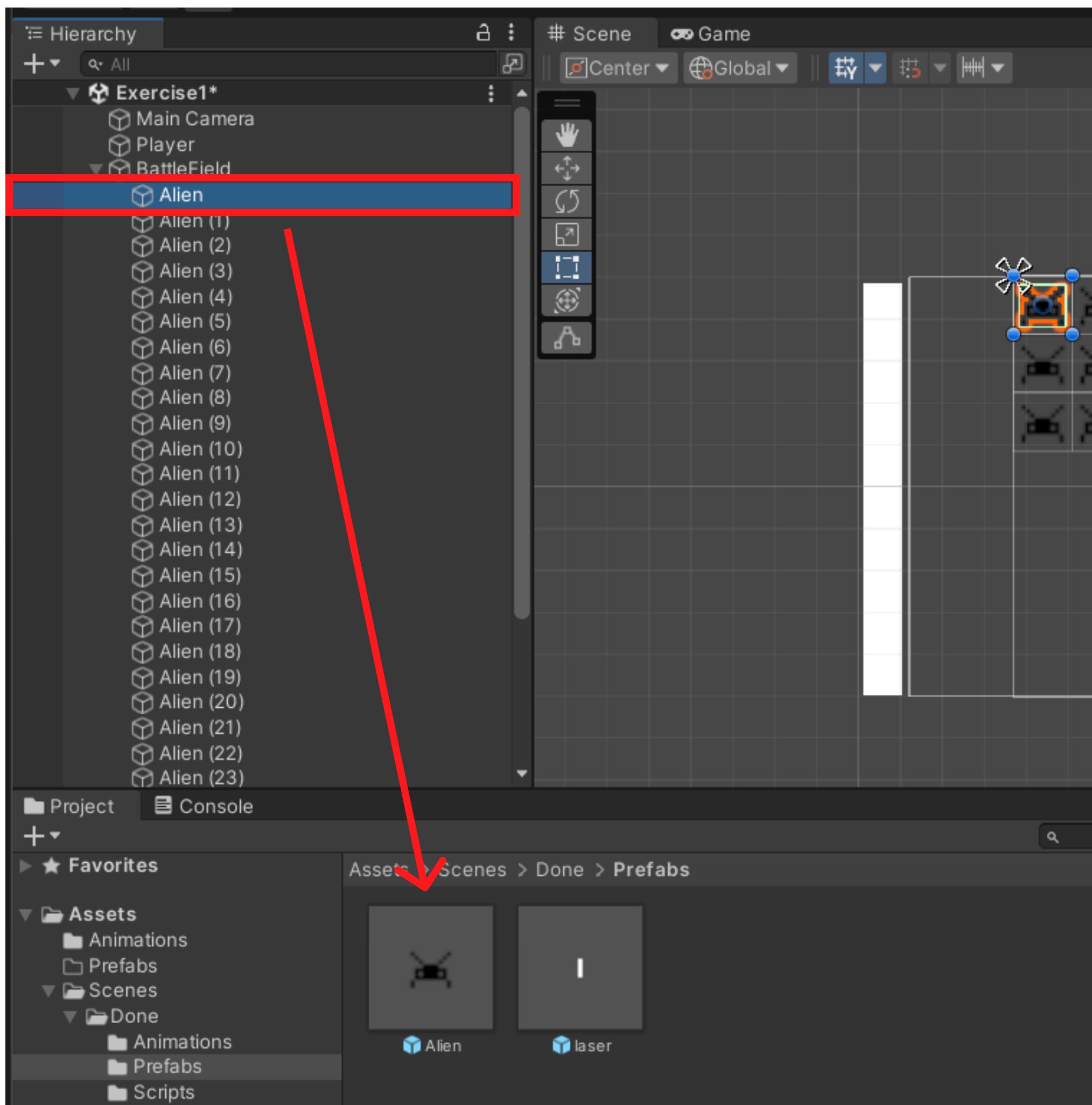
MUDIFICA O “BODY TYPE” PARA KINEMATIC



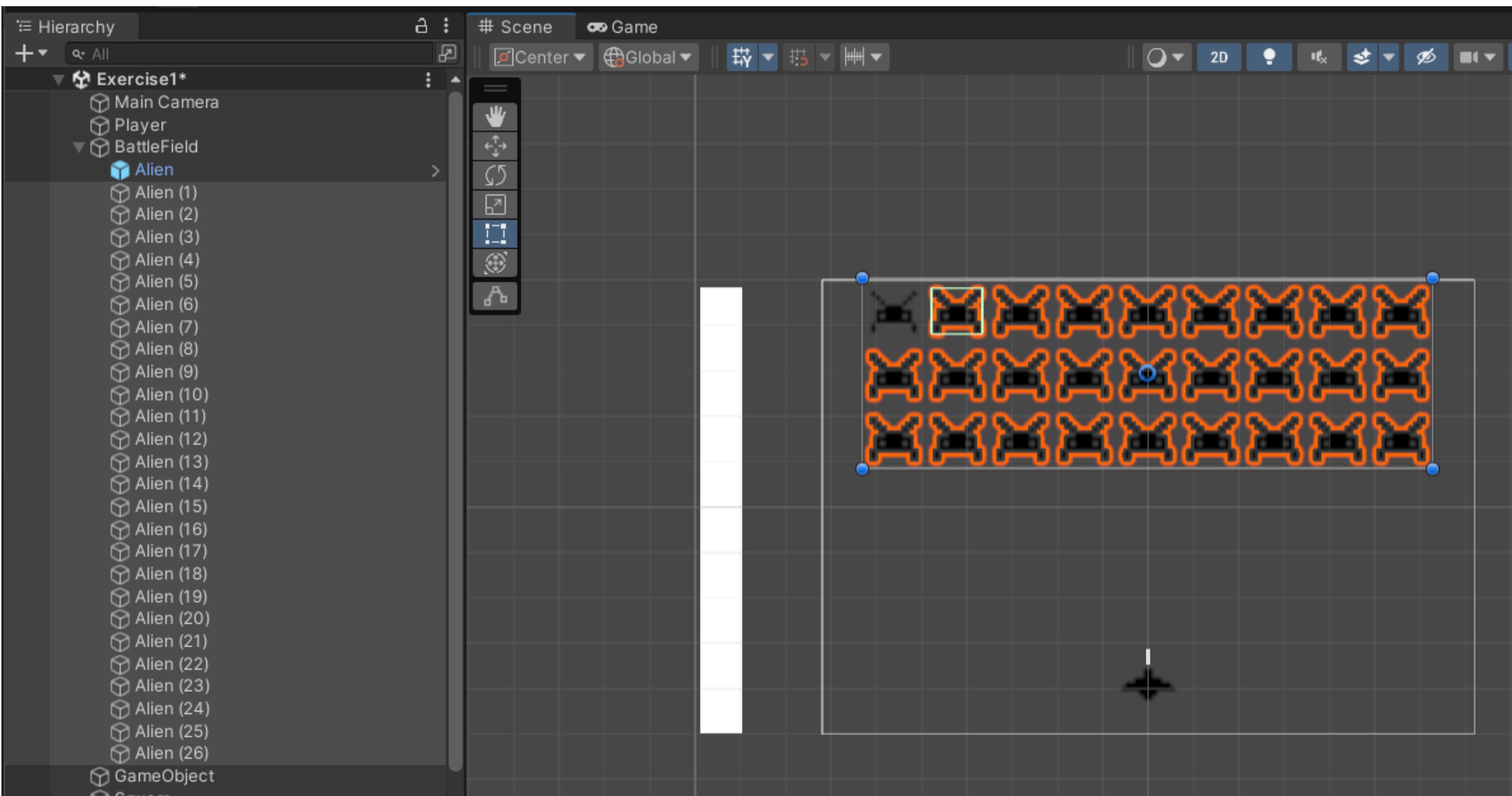
CRIA UM SCRIPT CHAMADO “LASERDETECTION” E ASSOCIA-O AO TEU PRIMEIRO INIMIGO



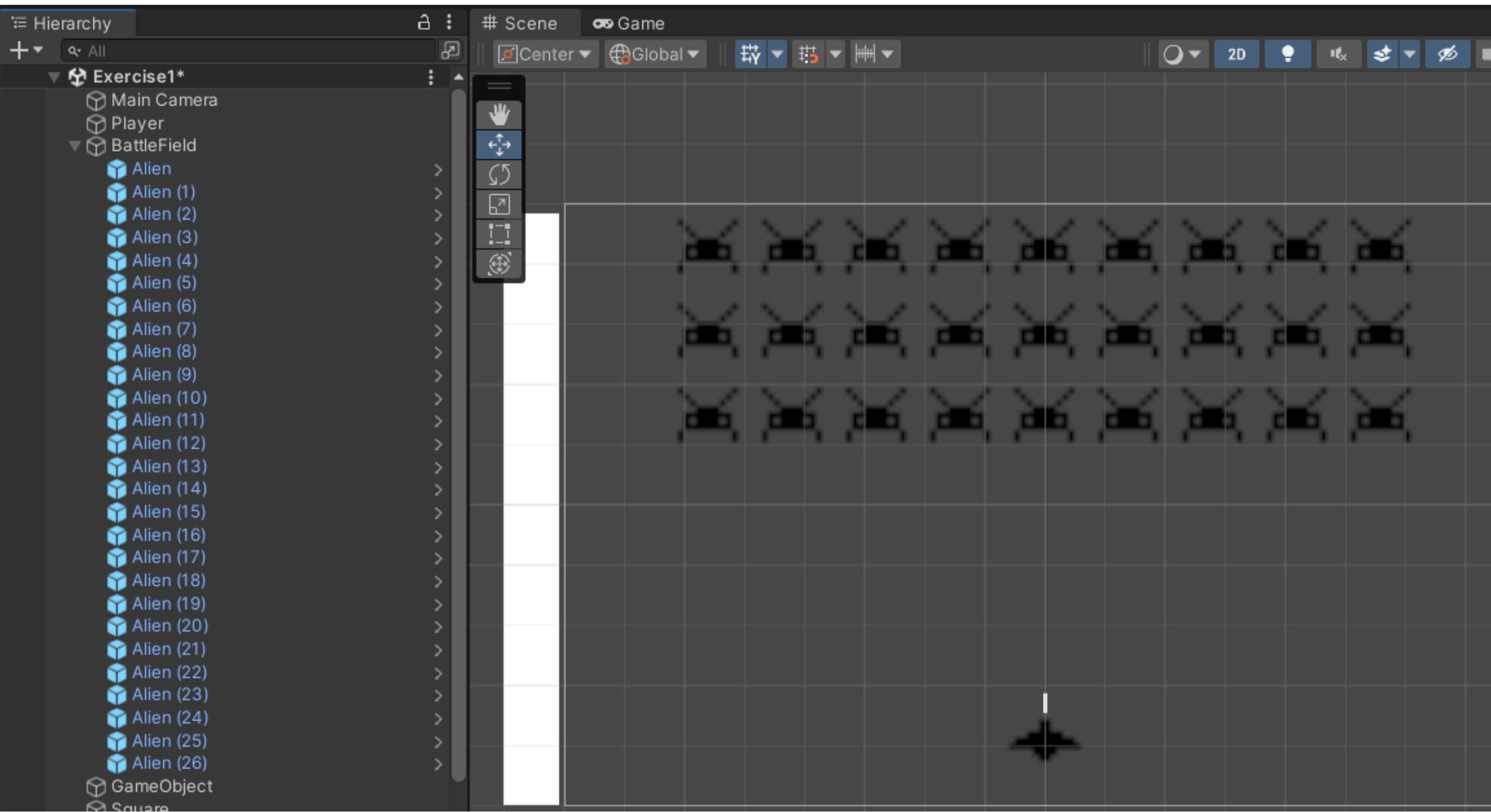
CRIA UM PREFAB BASEADO NO TEU PRIMEIRO INIMIGO



ELIMINA TODOS OS OUTROS INIMIGOS



COPIA E COLA O TEU PREFAB



EDITA O SCRIPT “LASERMOVEMENT” E ADICIONA O MÉTODO “ONTRIGGERENTER2D”

```
Script do Unity (1 referência de ativo) | 0 referências
public class LaserDetection : MonoBehaviour
{
    // Start is called before the first frame update
    Mensagem do Unity | 0 referências
    void Start()
    {

    }

    // Update is called once per frame
    Mensagem do Unity | 0 referências
    void Update()
    {

    }

    Mensagem do Unity | 0 referências
    private void OnTriggerEnter2D(Collider2D collision)
    {
        }

    }

}
```

Este método é responsável por detetar uma colisão, através do componente “box collider 2D”.

DESTROI O LASER APÓS A COLISÃO

```
private void OnTriggerEnter2D(Collider2D collision)
{
    if (collision.tag == "laser") {
        //o objeto que colide com o inimigo é destruído
        Destroy(collision.gameObject);
    }
}
```

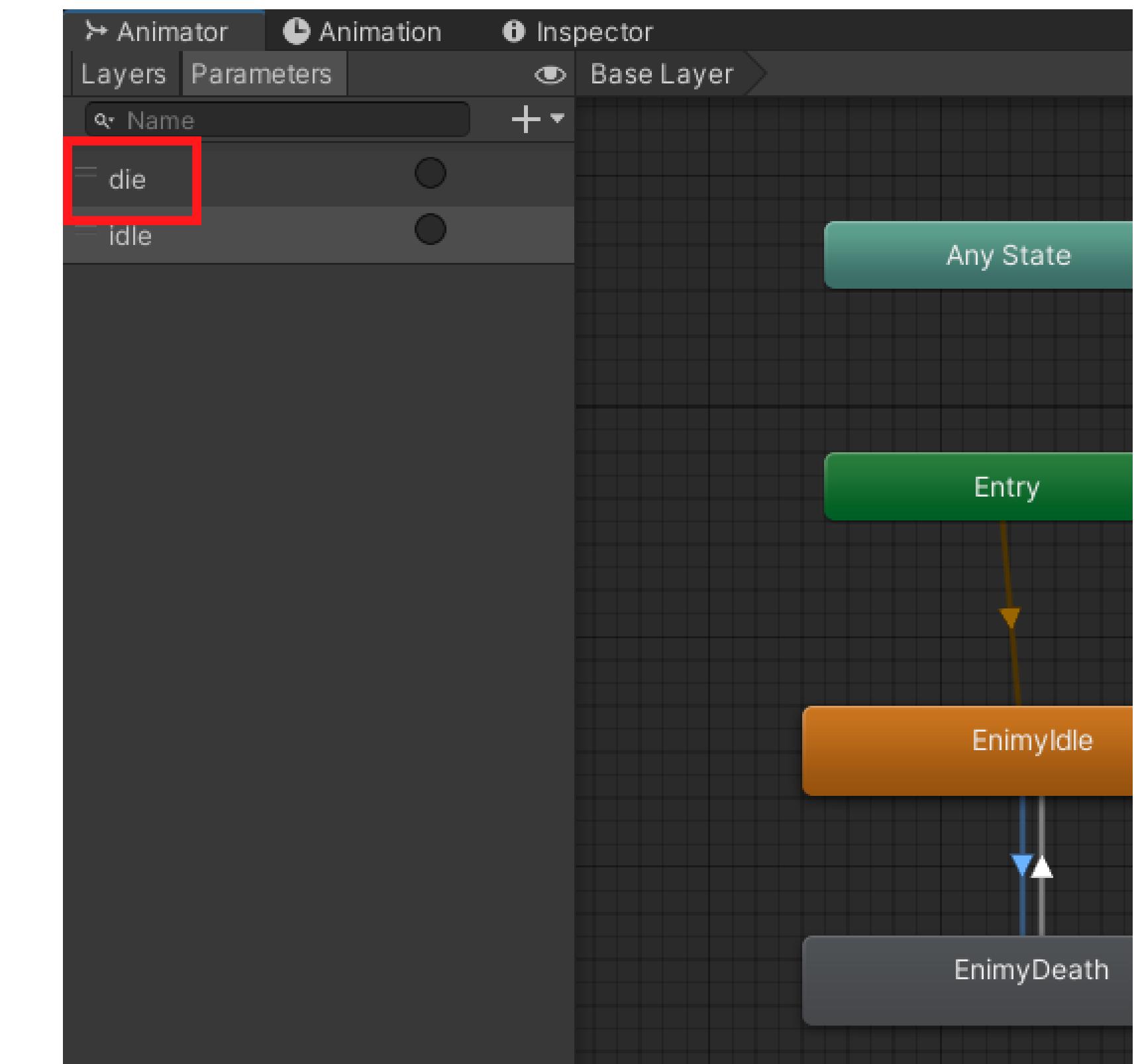
O método “Destroy” é responsável por destruir qualquer objeto.

INICIALIZA UMA VARIÁVEL REFERENTE AO ANIMATOR (O CONTROLLER QUE CRIASTE PARA AS ANIMAÇÕES DO TEU INIMIGO)

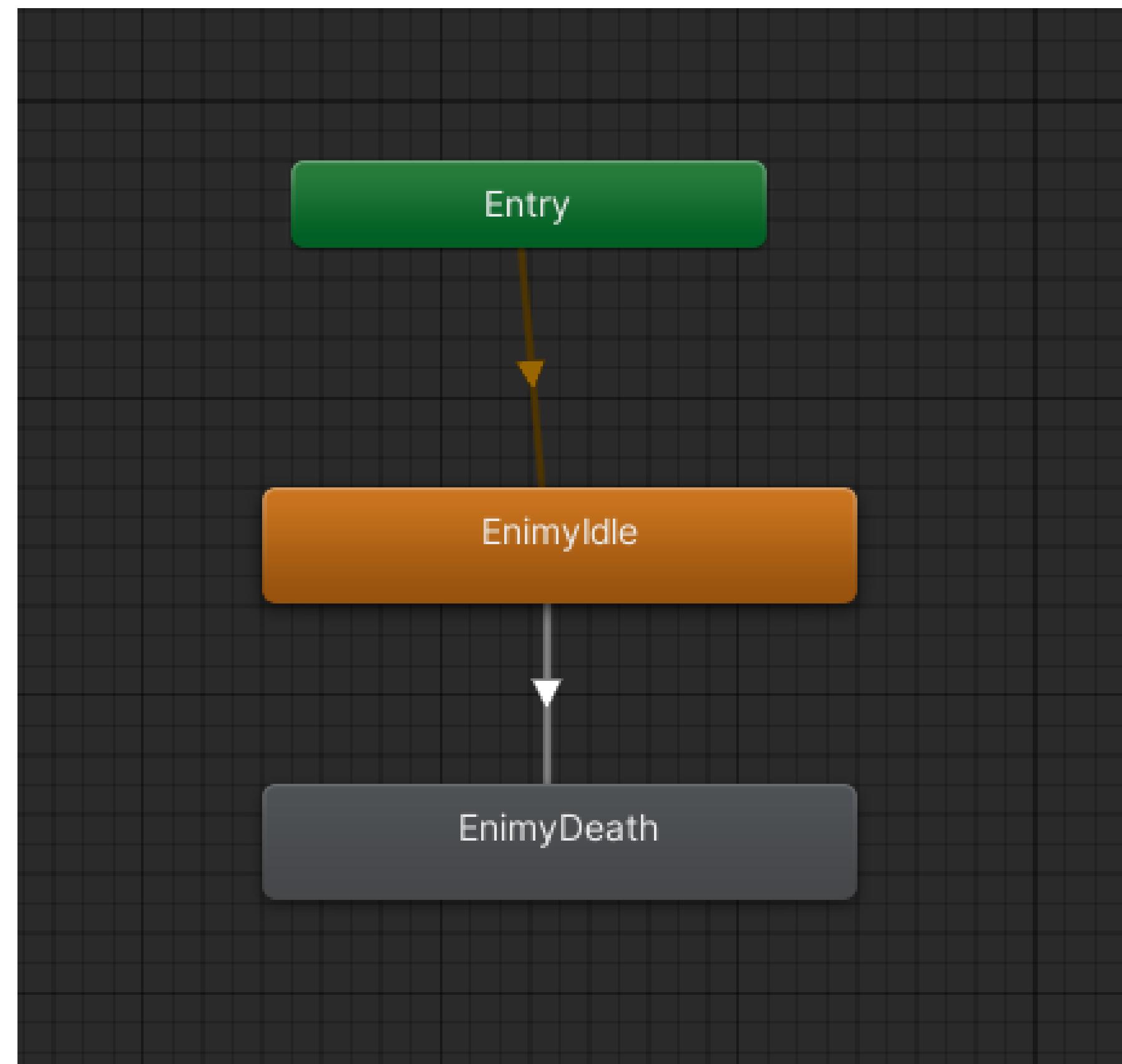
```
Script do Unity (Referência de ativo) | 0 referências
public class LaserDetection : MonoBehaviour
{
    Animator EnemyAnimator;
    // Start is called before the first frame update
    Mensagem do Unity | 0 referências
    void Start()
    {
        EnemyAnimator = this.gameObject.GetComponent<Animator>();
    }
}
```

ACIONA A ANIMAÇÃO DE “MORTE”, DE ACORDO COM O TRIGGER NO TEU ANIMATOR

```
private void OnTriggerEnter2D(Collider2D collision)
{
    if (collision.tag == "laser") {
        //o objeto que colide com o inimigo é destruído
        Destroy(collision.gameObject);
        //aciona a animação de morte do inimigo
        EnemyAnimator.SetTrigger("die");
    }
}
```



CERTIFICA-TE QUE A ANIMAÇÃO DO INIMIGO NÃO REGRESSA A ‘IDLE’ AUTOMÁTICAMENTE



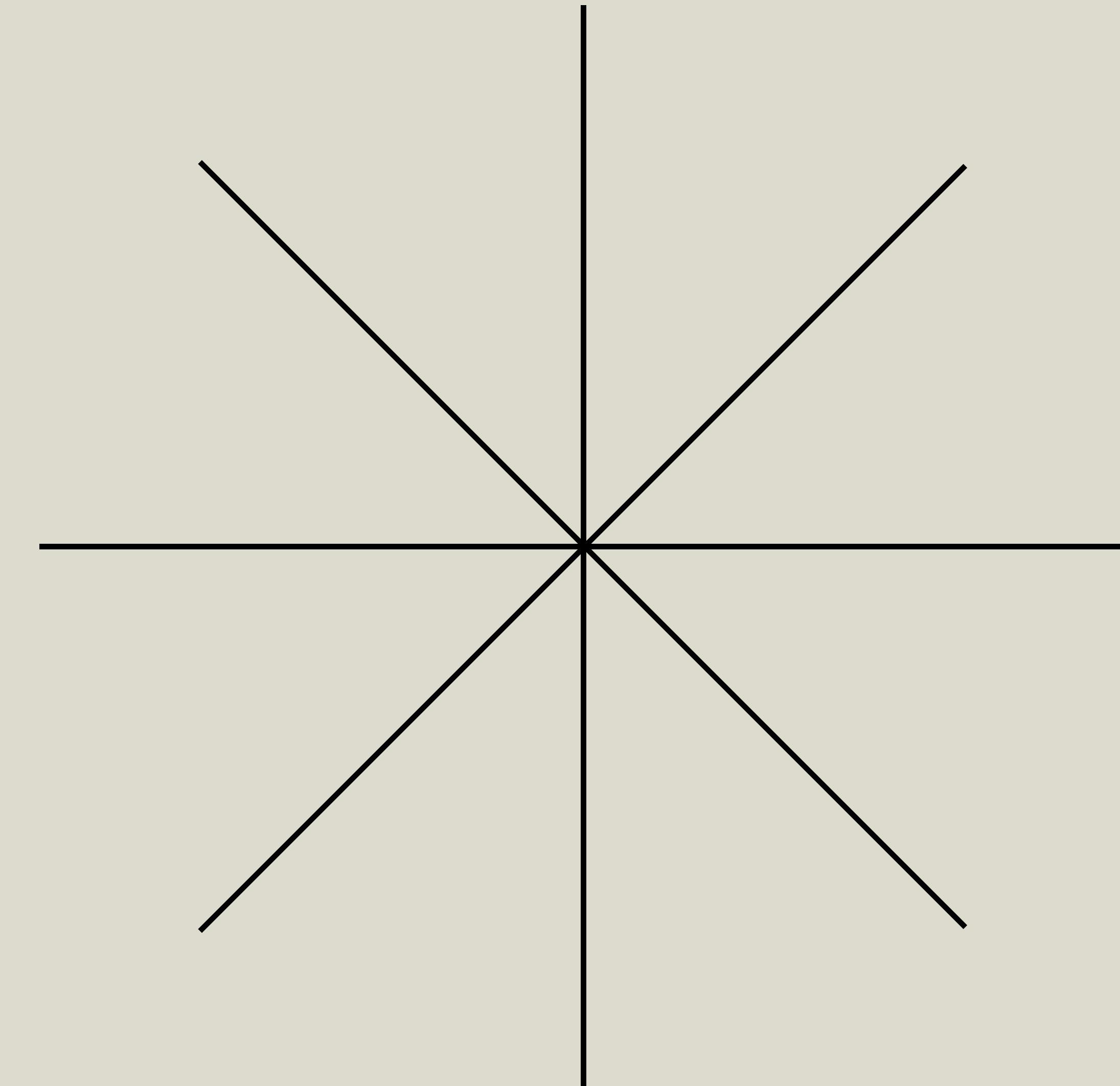
INICIALIZA UMA VARIÁVEL PARA O COMPONENTE COLLIDER 2D DO TEU INIMIGO

```
public class LaserDetection : MonoBehaviour
{
    Animator EnemyAnimator;
    Collider2D EnemyCollider;
    // Start is called before the first frame update
    Mensagem do Unity | 0 referências
    void Start()
    {
        EnemyAnimator = this.gameObject.GetComponent<Animator>();
        EnemyCollider = this.gameObject.GetComponent<Collider2D>();
    }
}
```

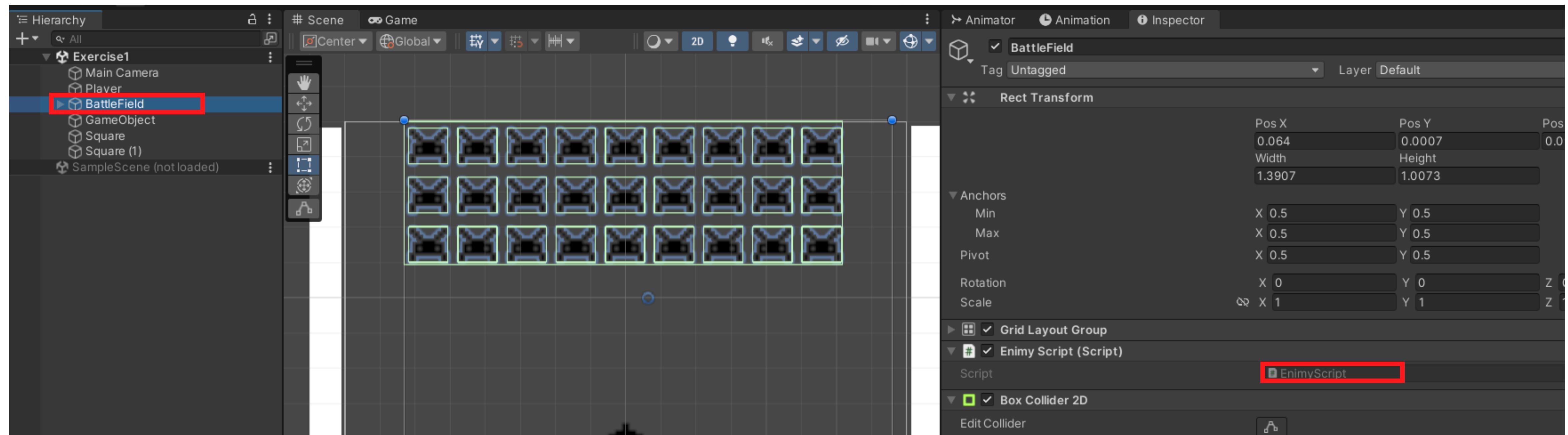
DESATIVA O COMPONENTE COLLIDER 2D

```
↳ Mensagem do Unity para referencias
private void OnTriggerEnter2D(Collider2D collision)
{
    if (collision.tag == "laser") {
        //o objeto que colide com o inimigo é destroido
        Destroy(collision.gameObject);
        //aciona a animação de morte do inimigo
        EnemyAnimator.SetTrigger("die");
        //desativa o componente collider2D
        EnemyCollider.enabled = false;
    }
}
```

03 Vitória



ABRE O TEU SCRIPT ‘ENEMYSCRIPT’ QUE ESTÁ ASSOCIADO AO OBJETO ‘BATTLEFIELD’



CRIA UMA VARIÁVEL ‘ENEMYCOUNT’ DO TIPO INT COM UM VALOR IDÊNTICO À QUANTIDADE DE INIMIGOS NA CENA

```
public class EnemyScript : MonoBehaviour
{
    float moveSpeed = 0.1f;
    float moveDirection = 1;

    public int enemyCount = 27;

    // Start is called before the first frame
    void Start()
    {
```

CRIA UM MÉTODO PARA O FIM DO JOGO:

```
// Update is called once per frame

void Update()
{
    transform.Translate(Vector2.right
}

0 referências
public void EndOfGame(string state)
{
}


private void OnTriggerEnter2D(Collider2D collision)
{
```

CRIA UMA LÓGICA DENTRO DO TEU MÉTODO PARA CADA CASO DE FIM DE JOGO (VITÓRIA OU MORTE)

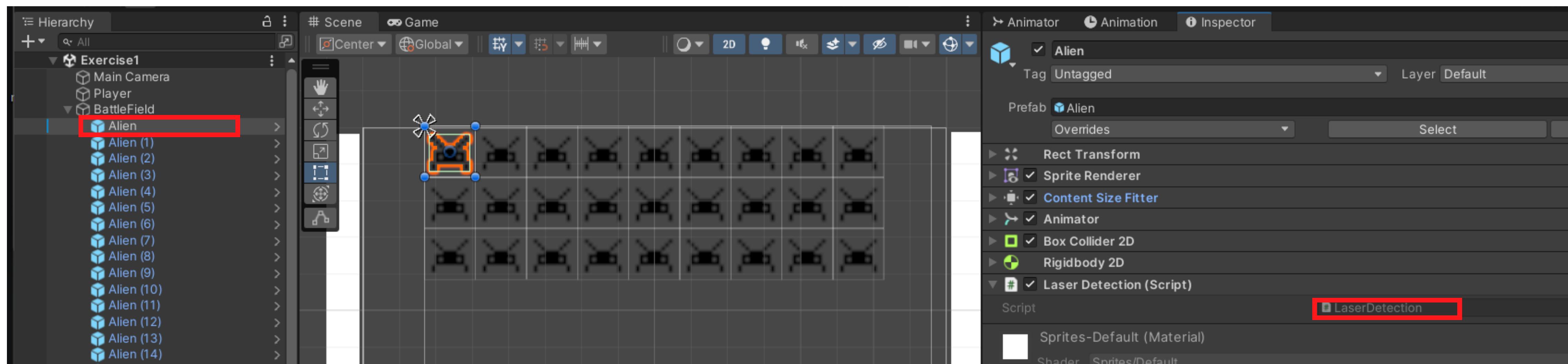
```
1 referência
public void EndOfGame(string state)
{
    if(state == "win")
    {
        print("You Won!");
    }
    else
    {
        print("You Lost!");
    }
}
```

ADICIONA UMA CONDIÇÃO PARA DETETAR A MORTE DE TODOS OS INIMIGOS

```
// Update is called once per frame
↳ Mensagem do Unity | 0 referências
void Update()
{
    transform.Translate(Vector2.right * ...

    //condicao para ganhar o jogo
    if(enemyCount == 0)
    {
        EndOfGame("win");
    }
}
```

ABRE O TEU SCRIPT ‘LASERDETECTION’ QUE ESTÁ ASSOCIADO A CADA INIMIGO



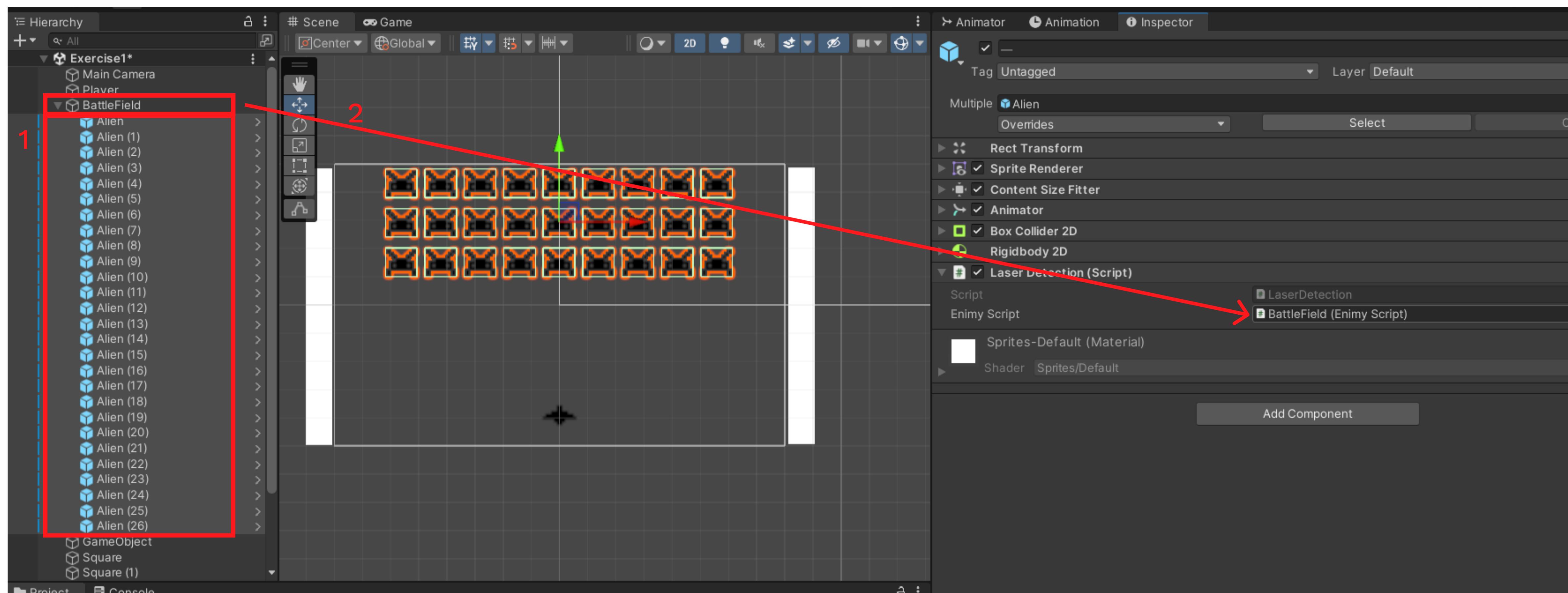
CRIA UMA VARIÁVEL ‘ENIMYSCRIPT’ DA MESMA CLASSE QUE O TEU SCRIPT ‘ENIMYSCRIPT’ NO TEU ‘BATTLEFIELD’

```
public class LaserDetection : MonoBehaviour
{
    Animator EnemyAnimator;
    Collider2D EnemyCollider;

    [SerializeField] EnimyScript enimyScript;

    // Start is called before the first frame update
}
```

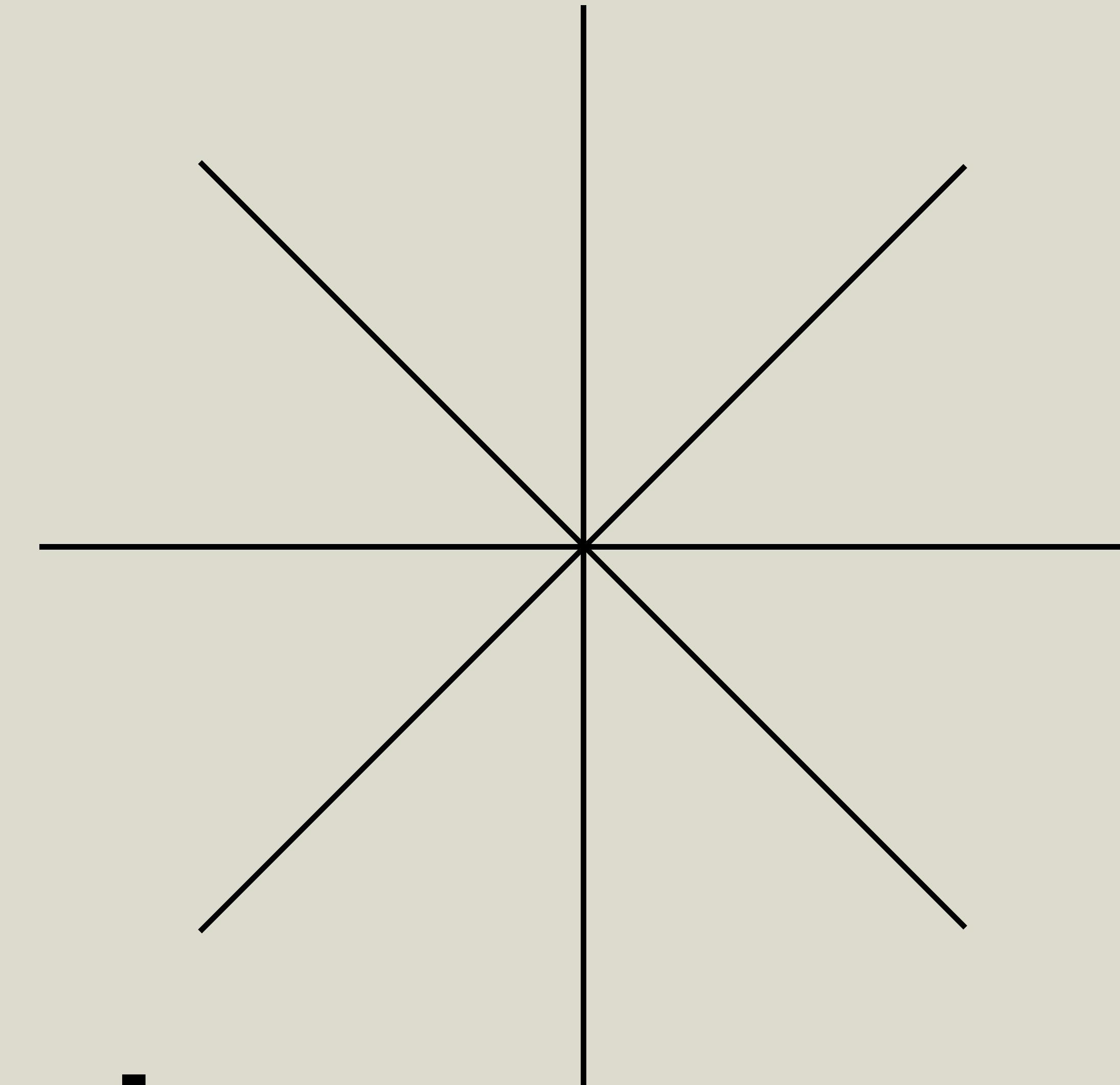
ATRIBUI À VARIÁVEL O SCRIPT DO OBJETO ‘BATTLEFIELD’ PARA CADA INIMIGO



DIMINUI A VARIÁVEL ‘ENEMYCOUNT’ A CADA MORTE

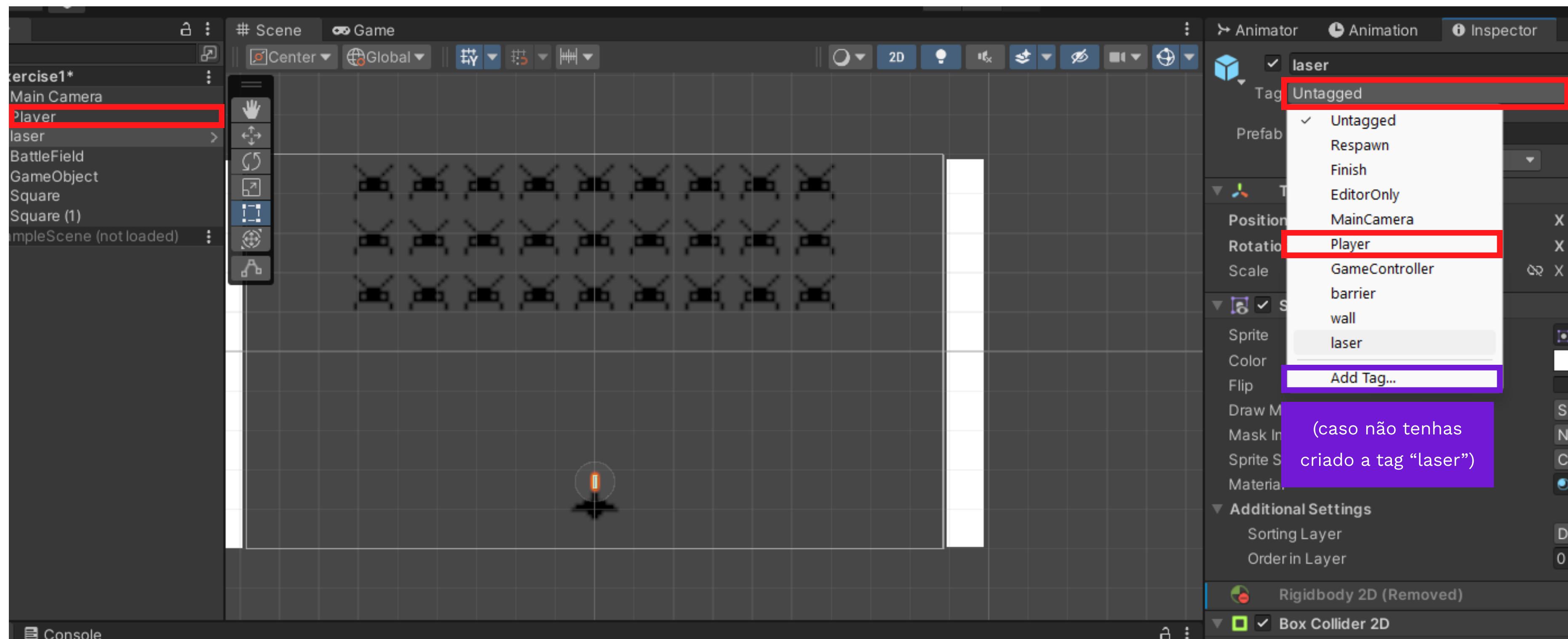
```
private void OnTriggerEnter2D(Collider2D collision)
{
    if (collision.tag == "laser") {
        //o objeto que colide com o inimigo é destruído
        Destroy(collision.gameObject);
        //aciona a animação de morte do inimigo
        EnemyAnimator.SetTrigger("die");
        //desativa o componente collider2D
        EnemyCollider.enabled = false;

        //diminui o número 'enemycount' no outro script
        enemyScript.enemyCount--;
    }
}
```

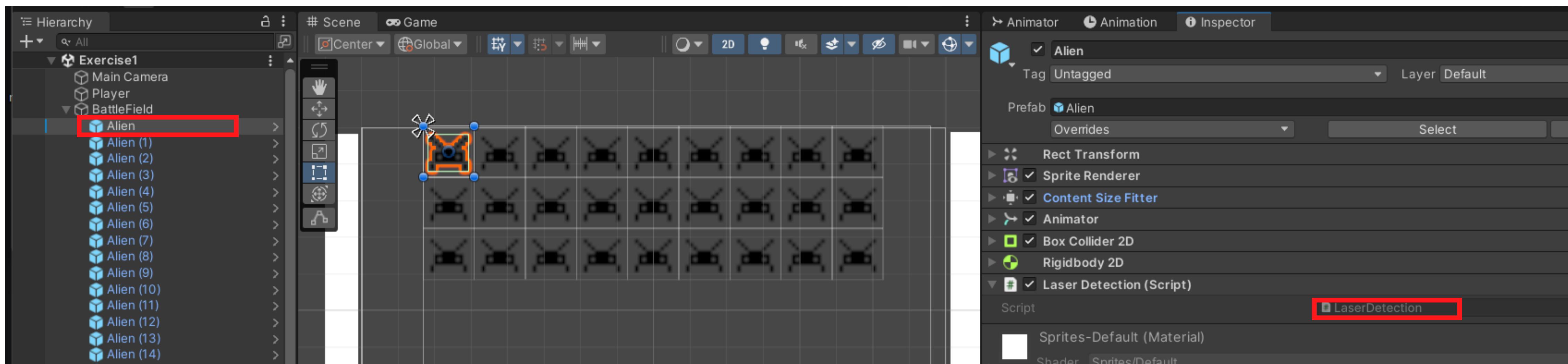


04 Morte do Jogador

ADICIONA A TAG “PLAYER” AO OBJETO DO PLAYER



ABRE O TEU SCRIPT ‘LASERDETECTION’ QUE ESTÁ ASSOCIADO A CADA INIMIGO



DETECA A COLISÃO ENTRE O INIMIGO E O PLAYER

```
private void OnTriggerEnter2D(Collider2D collision)
{
    if (collision.tag == "laser")...
    else if(collision.tag == "Player")
    {
    }
}
```

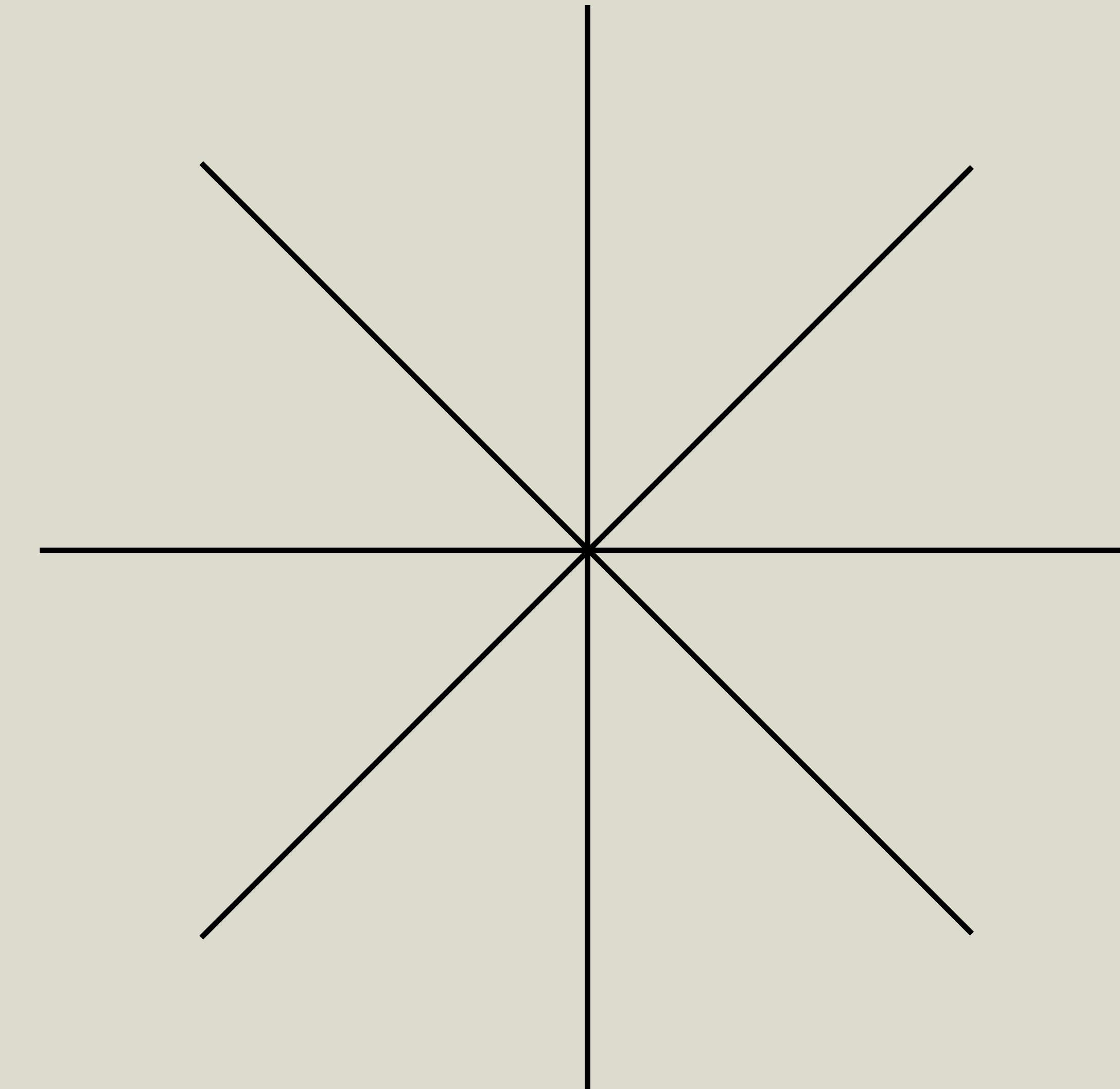
DESTROI O OBJETO PLAYER

```
else if(collision.tag == "Player")
{
    Destroy(collision.gameObject);
}
```

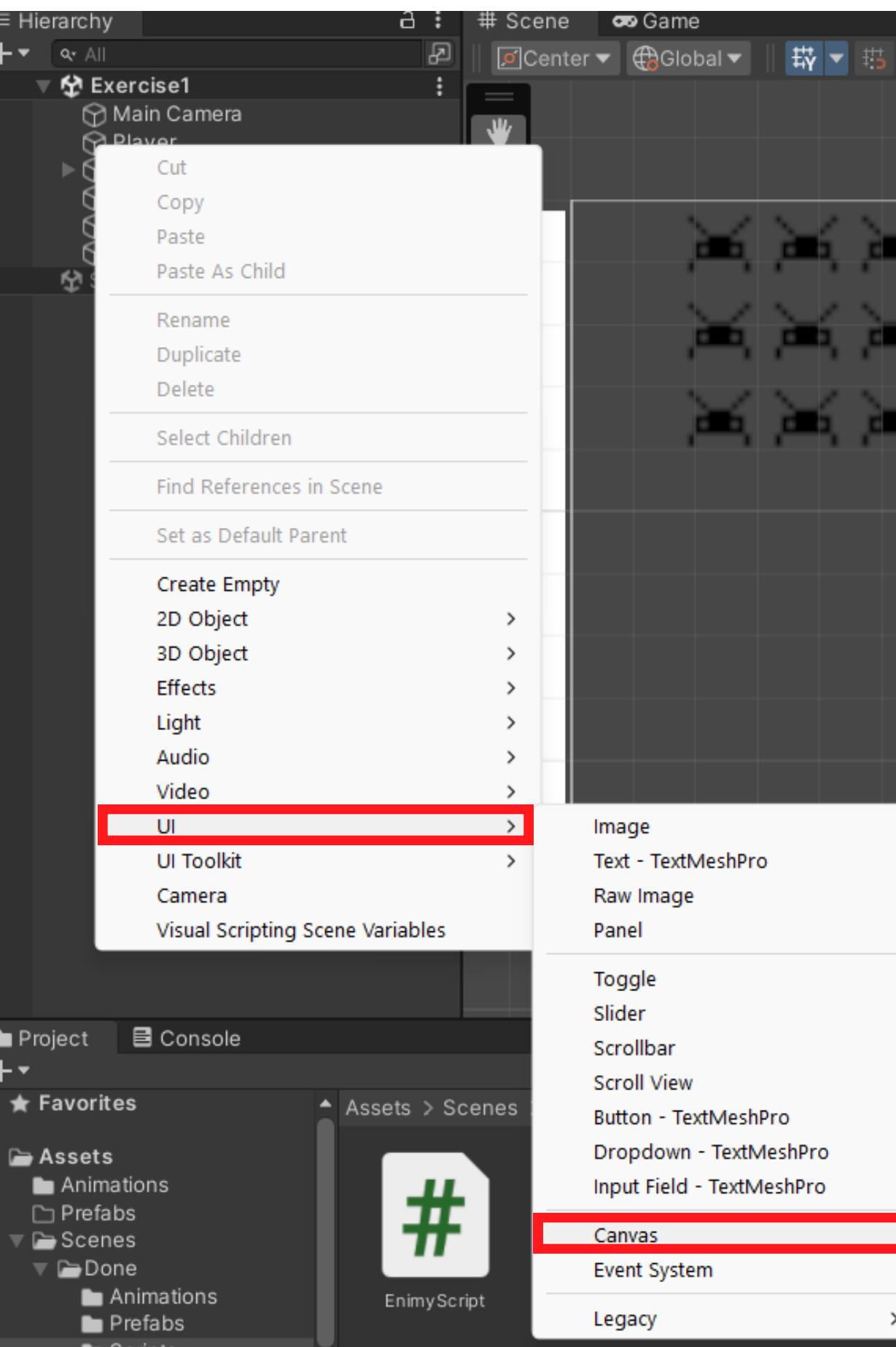
CHAMA A FUNÇÃO DE FIM DO JOGO

```
else if(collision.tag == "Player")
{
    //destroi o objeto player
    Destroy(collision.gameObject);
    //chama a função fim de jogo
    enemyScript.EndOfGame("loose");
}
```

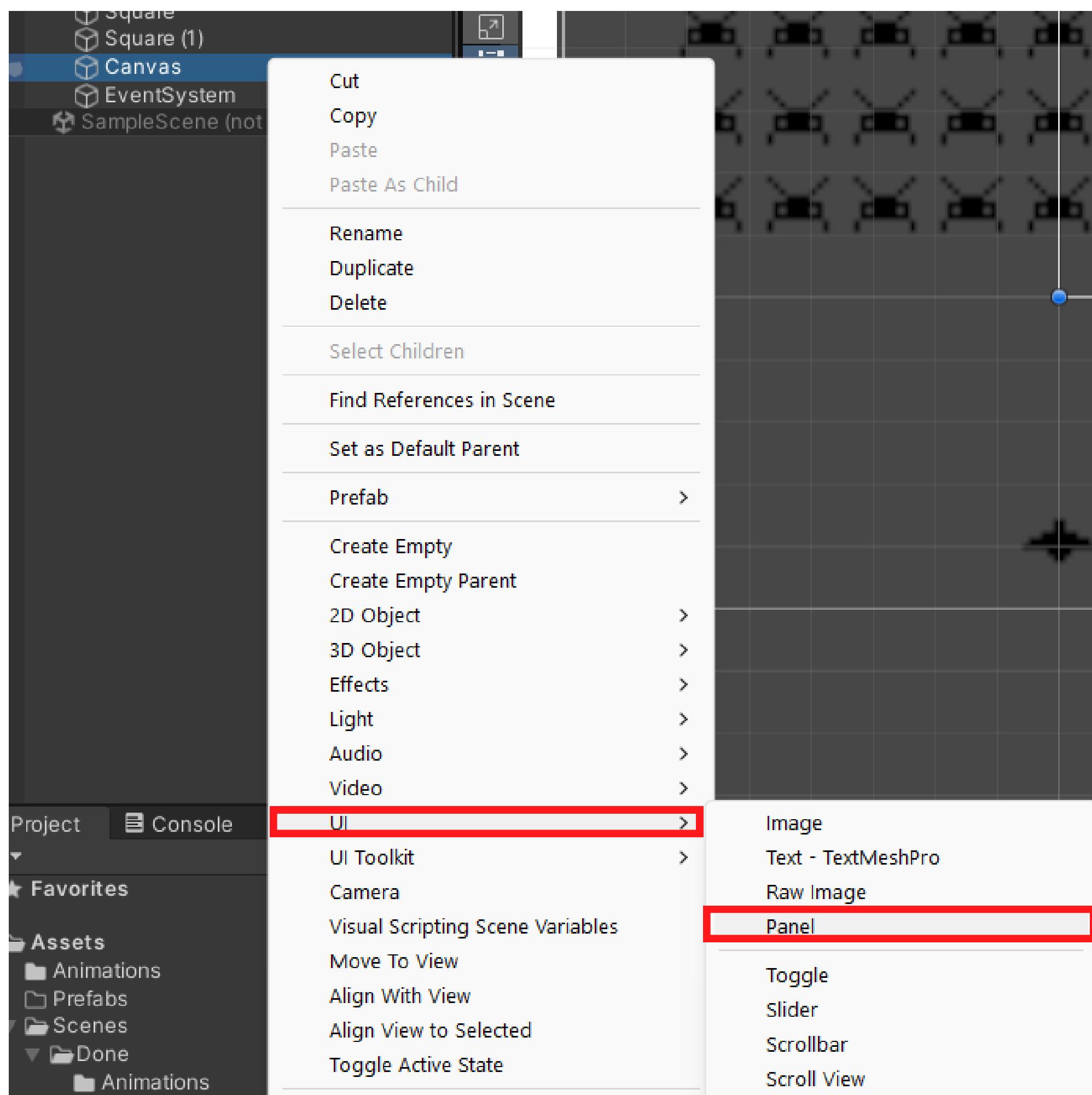
05 UI do Jogo



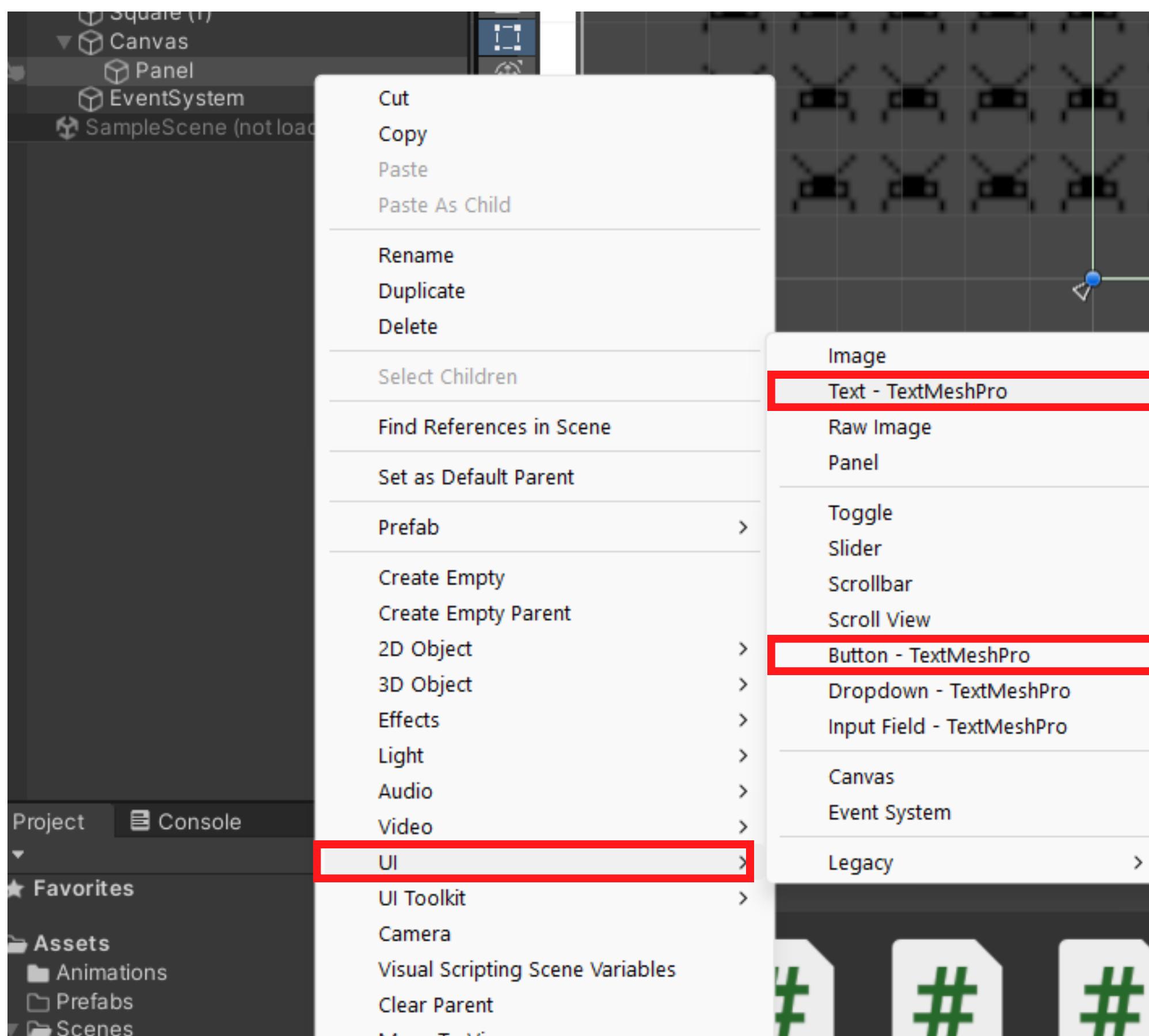
ADICIONA UM CANVAS À CENA



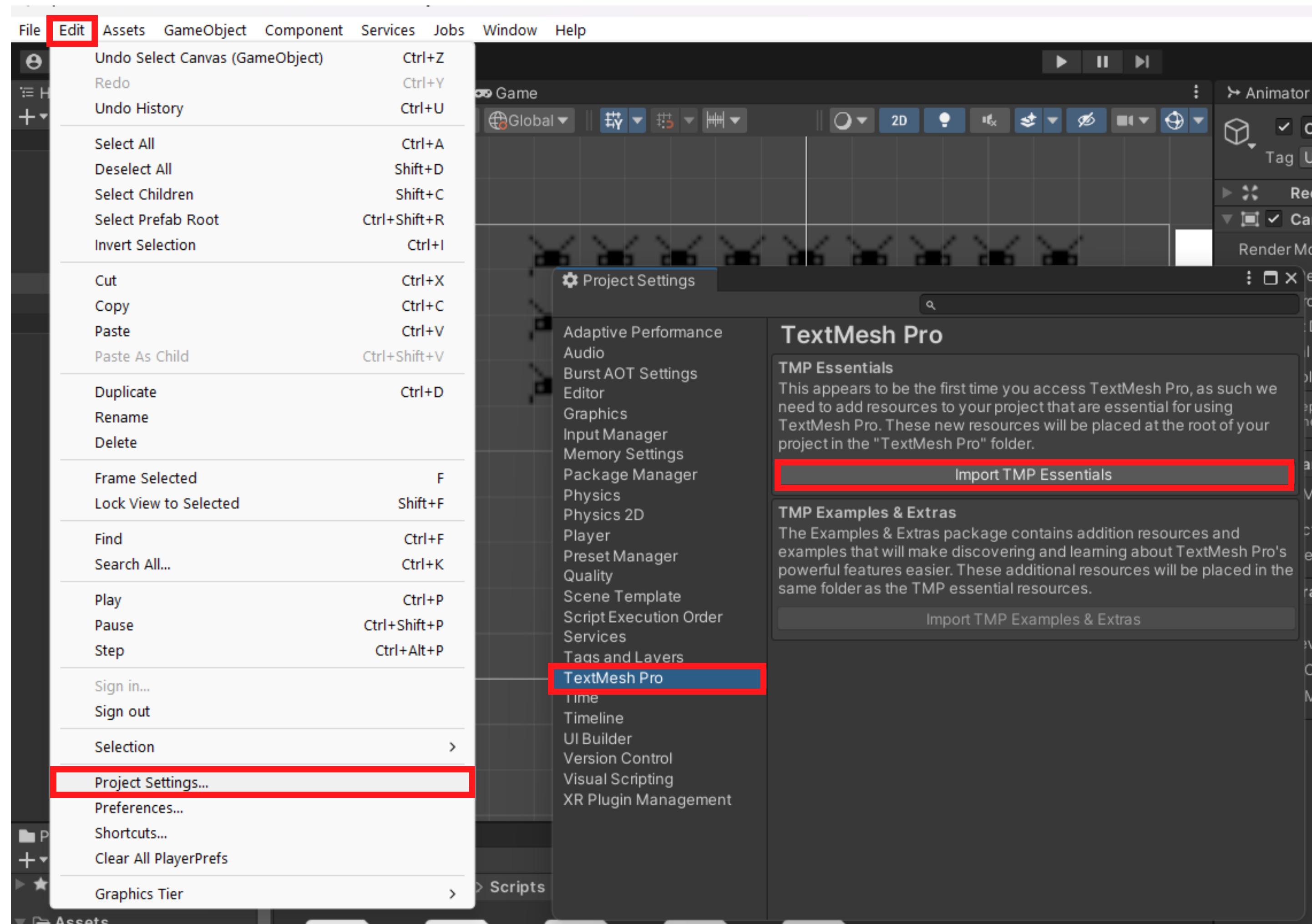
ADICIONA UM PAINEL PARA O UI NO CANVAS



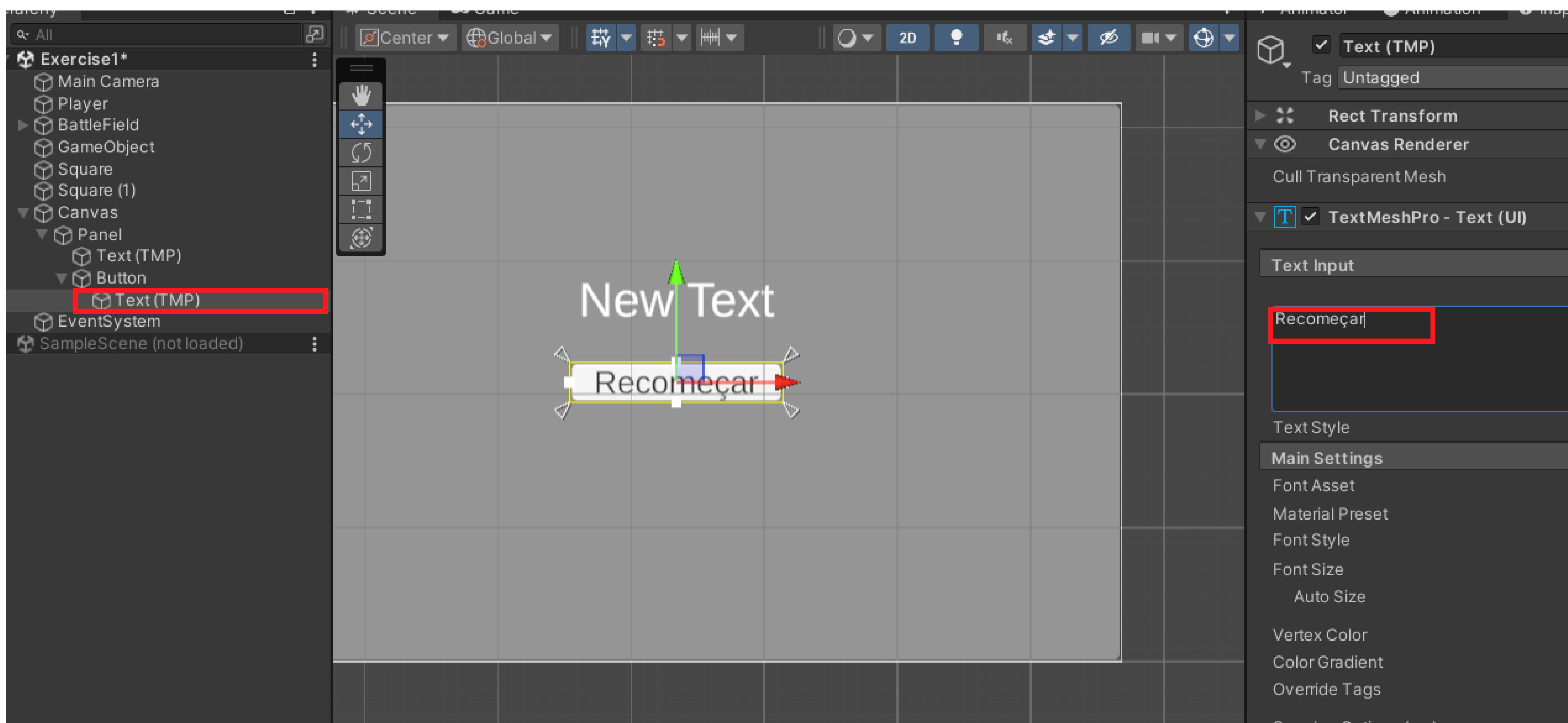
ADICIONA UMA CAIXA DE TEXTO E UM BOTAO NO PAINEL



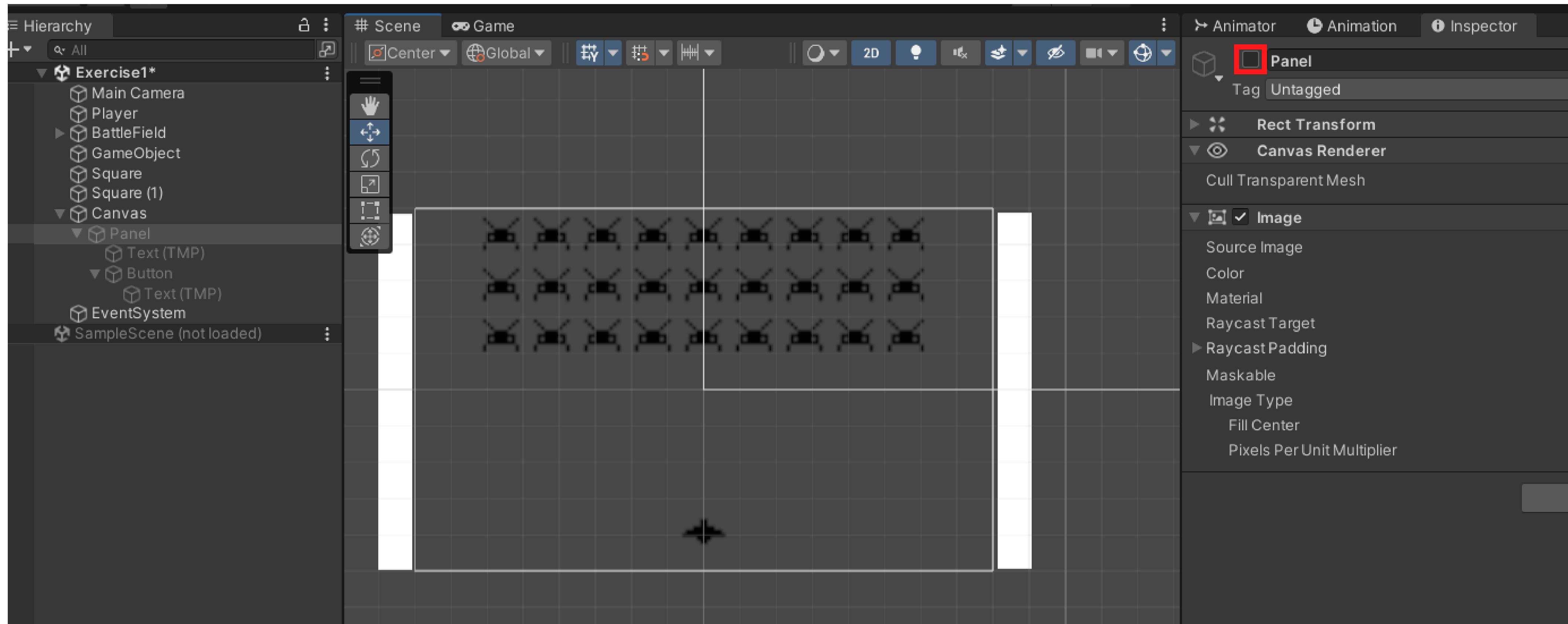
IMPORTA OS RECURSOS PARA USARES TEXTO (TEXT MESH PRO)



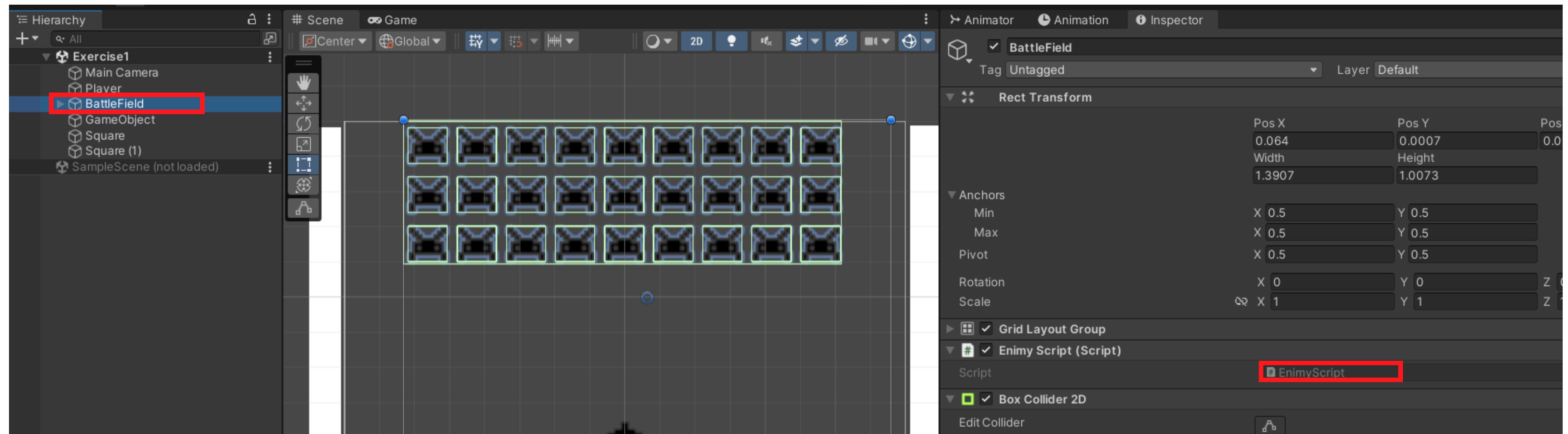
MUDIFICA O TEXTO DO BOTÃO PARA “RECOMEÇAR”



DESATIVA O PAINEL



ABRE O TEU SCRIPT ‘ENEMYSCRIPT’ QUE ESTÁ ASSOCIADO AO OBJETO ‘BATTLEFIELD’



EDITA A FUNÇÃO ‘ENDOFGAME’

```
2 referências
public void EndOfGame(string state)
{
    if(state == "win")
    {
        print("You Won!");
    }
    else
    {
        print("You Lost!");
    }
}
```

COLOCA O JOGO EM PAUSA

```
<referencias>
public void EndOfGame(string state)
{
    //pausa o 'tempo'
    Time.timeScale = 0;
    if(state == "win")
    {
```

IMPORTA A BIBLIOTECA “TMPRO” PARA O TEXTO NO PAINEL

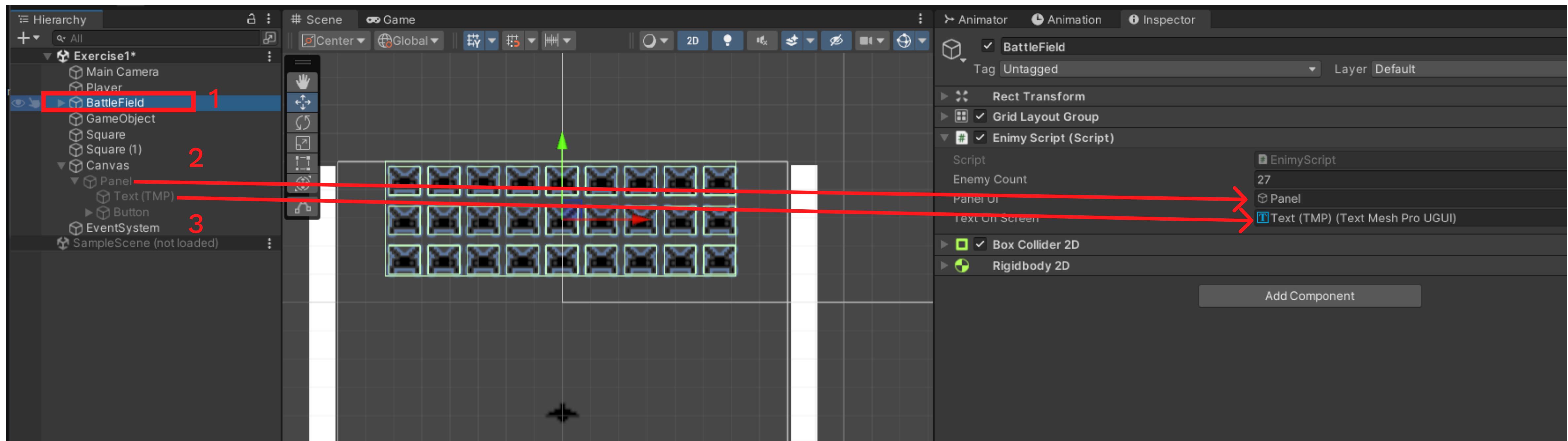
```
using UnityEngine;
using UnityEngine.Events;
using TMPro;
```

Script do Unity (1 referência de

INICIALIZA DUAS VARIÁVEIS, UMA PARA O PAINEL E UMA PARA O TEXTO

```
public int enemyCount = 27;  
  
[SerializeField] public GameObject panelUI;  
[SerializeField] public TextMeshProUGUI textOnScreen;  
  
// Start is called before the first frame update  
Mensagem do Unity | 0 referências
```

ATRUBUI OS OBJETOS DE UI E DO TEXTO RESPECTIVAMENTE, A PARTIR DO EDITOR



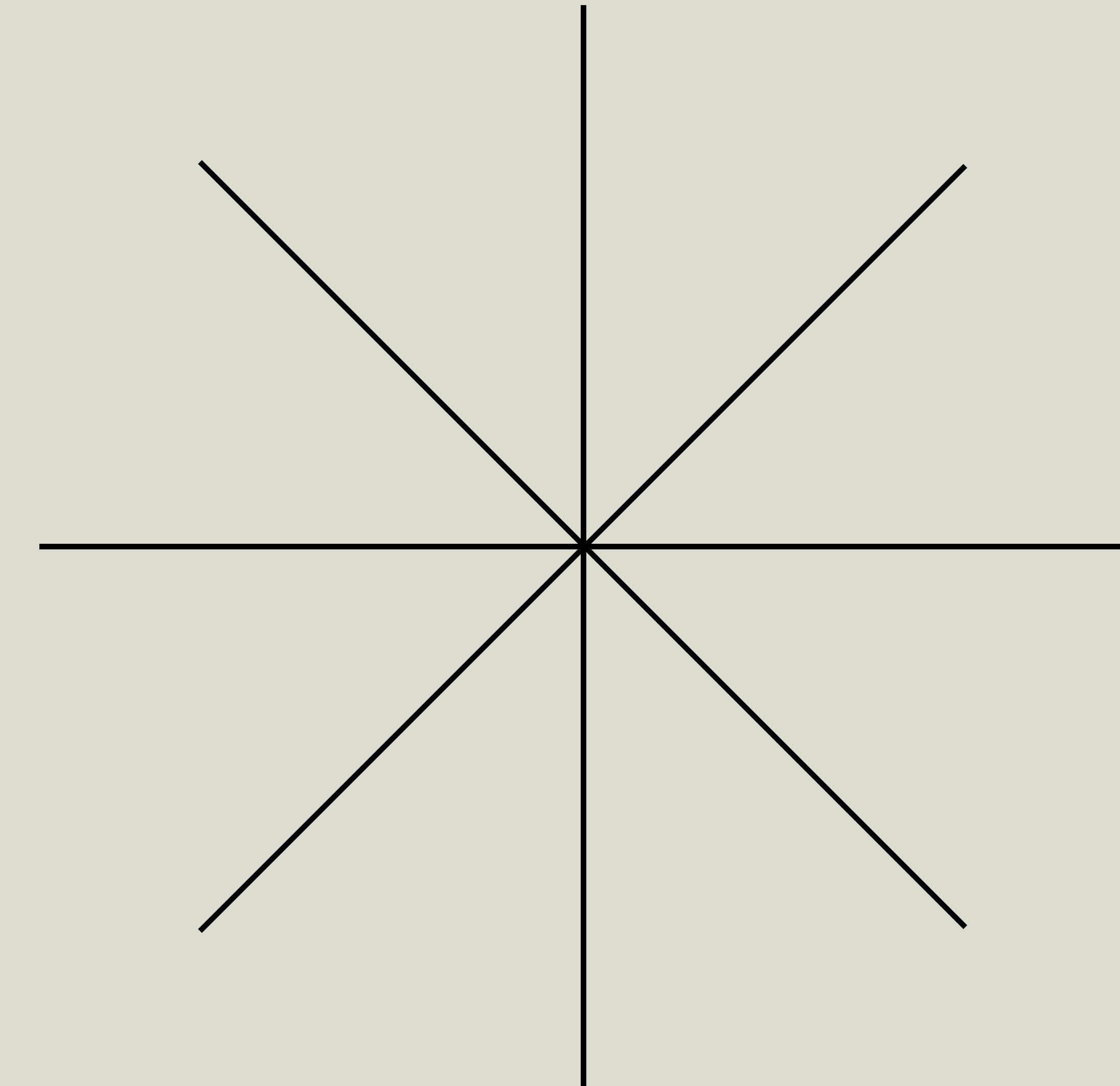
ATIVA O PAINEL A PARTIR DO CÓDIGO

```
public void EndOfGame(string state)
{
    //pausa o 'tempo'
    Time.timeScale = 0;
    //ativa o painel
    panelUI.SetActive(true);
    if(state == "win")
    {
        //ativa o painel
        panelUI.SetActive(true);
    }
}
```

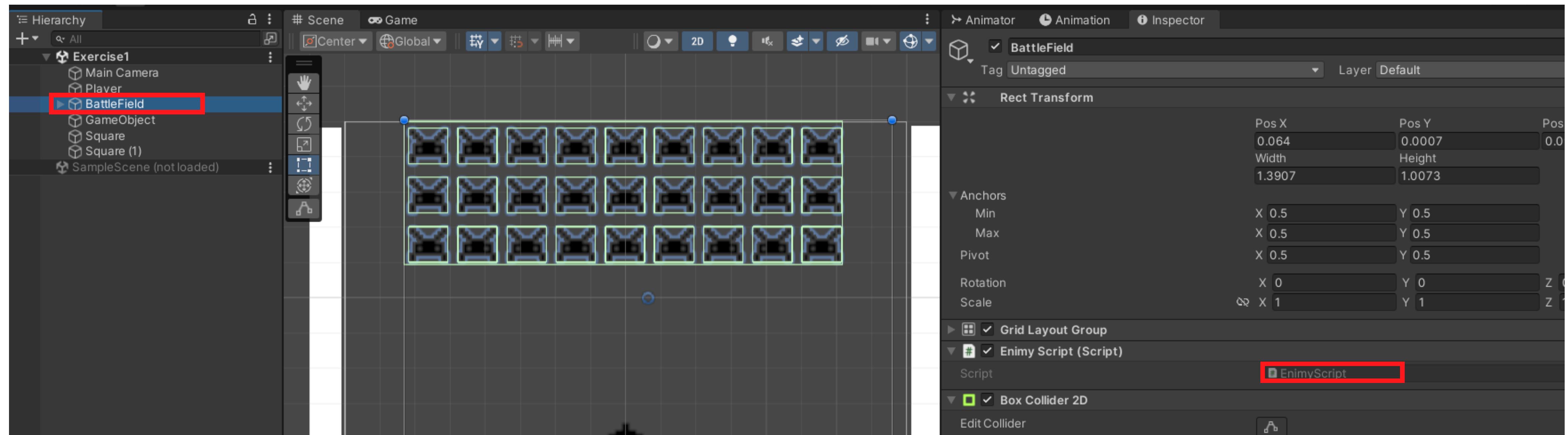
MUDIFICA O TEXTO PARA A MENSAGEM PARA CORRESPONDER A CADA SITUAÇÃO (MORTE OU VITÓRIA)

```
public void EndOfGame(string state)
{
    //pausa o 'tempo'
    Time.timeScale = 0;
    //ativa o painel
    panelUI.SetActive(true);
    if(state == "win")
    {
        textOnScreen.text = "You Won!";
        print("You Won!");
    }
    else
    {
        textOnScreen.text = "You Lost!";
        print("You Lost!");
    }
}
```

06 Recomeçar



ABRE O TEU SCRIPT ‘ENEMYSCRIPT’ QUE ESTÁ ASSOCIADO AO OBJETO ‘BATTLEFIELD’



IMPORTA A BIBLIOTECA PARA A GESTÃO DE CENAS

```
using UnityEngine.EventSystems;
using TMPro;
using UnityEngine.SceneManagement; { } na
    Script do Unity (1 referência de ativo) | 1 referência
public class EnemyScript : MonoBehaviour
```

CRIA UM MÉTODO PÚBLICO CHAMADO RESTART

```
5

0 referências
public void Restart()
{
}

Mensagem do Unity | 0 referências
private void OnTriggerEnter2D(Collide
```

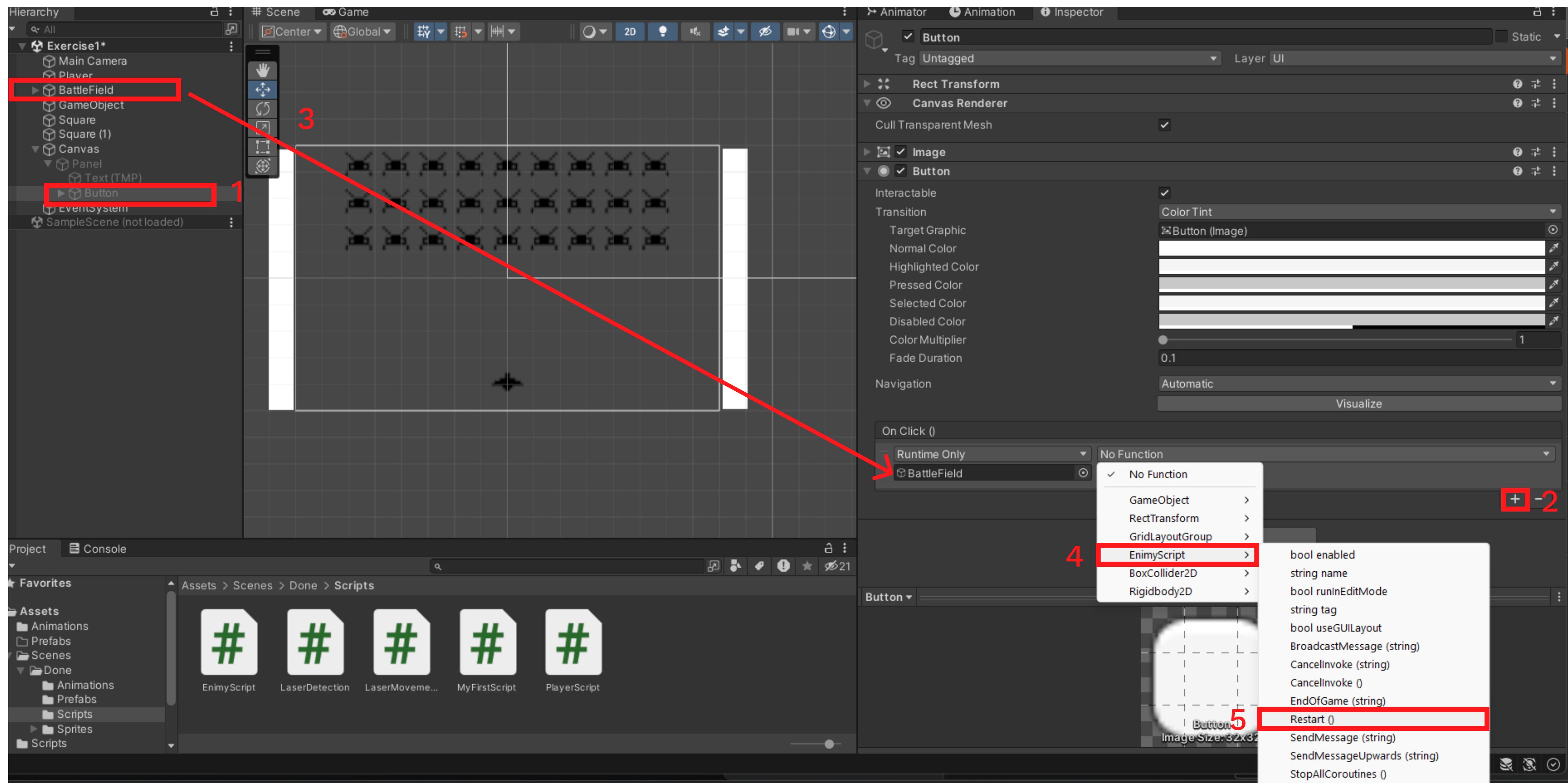
RETOMA O TEMPO

```
0 referências
public void Restart()
{
    Time.timeScale = 1;
}
```

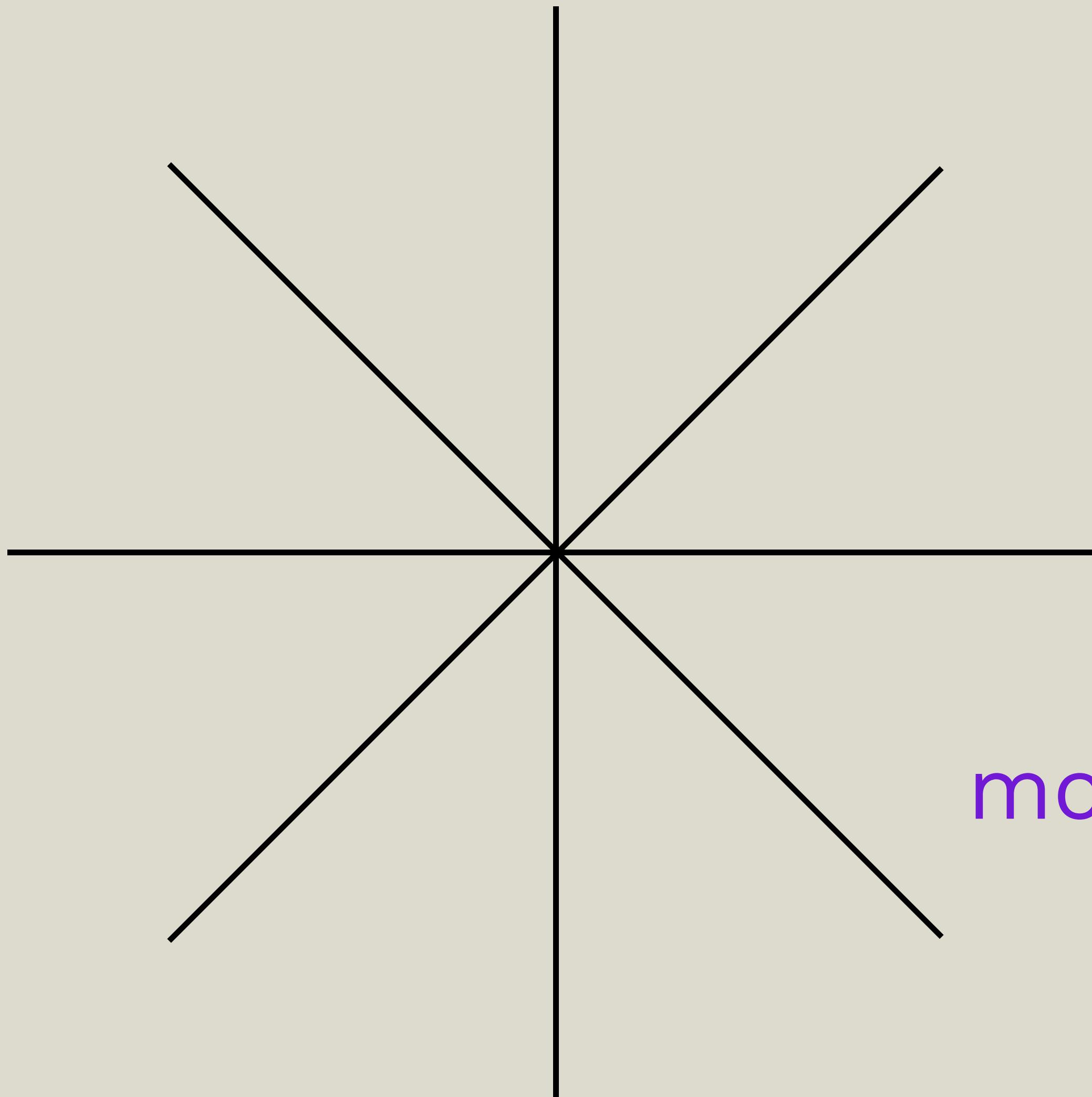
RECOMEÇA A CENA DENTRO DO MÉTODO

```
0 referencias
public void Restart()
{
    Time.timeScale = 1;
    SceneManager.LoadScene(0);
    //alternativa- colocar o nome exato da cena:
    SceneManager.LoadScene("SampleScene");
}
```

CHAMA O MÉTODO ‘RESTART’ A PARTIR DO BOTÃO



**DESAFIO:
IMPLEMENTA UM SISTEMA DE 3 VIDAS
QUE INCLUI UMA MECÂNICA DE ATAQUE
PARA OS INIMIGOS!**



Obrigada!

Não te esqueças onde
encontrar este ppt:

motamdaniela.github.io/tajd