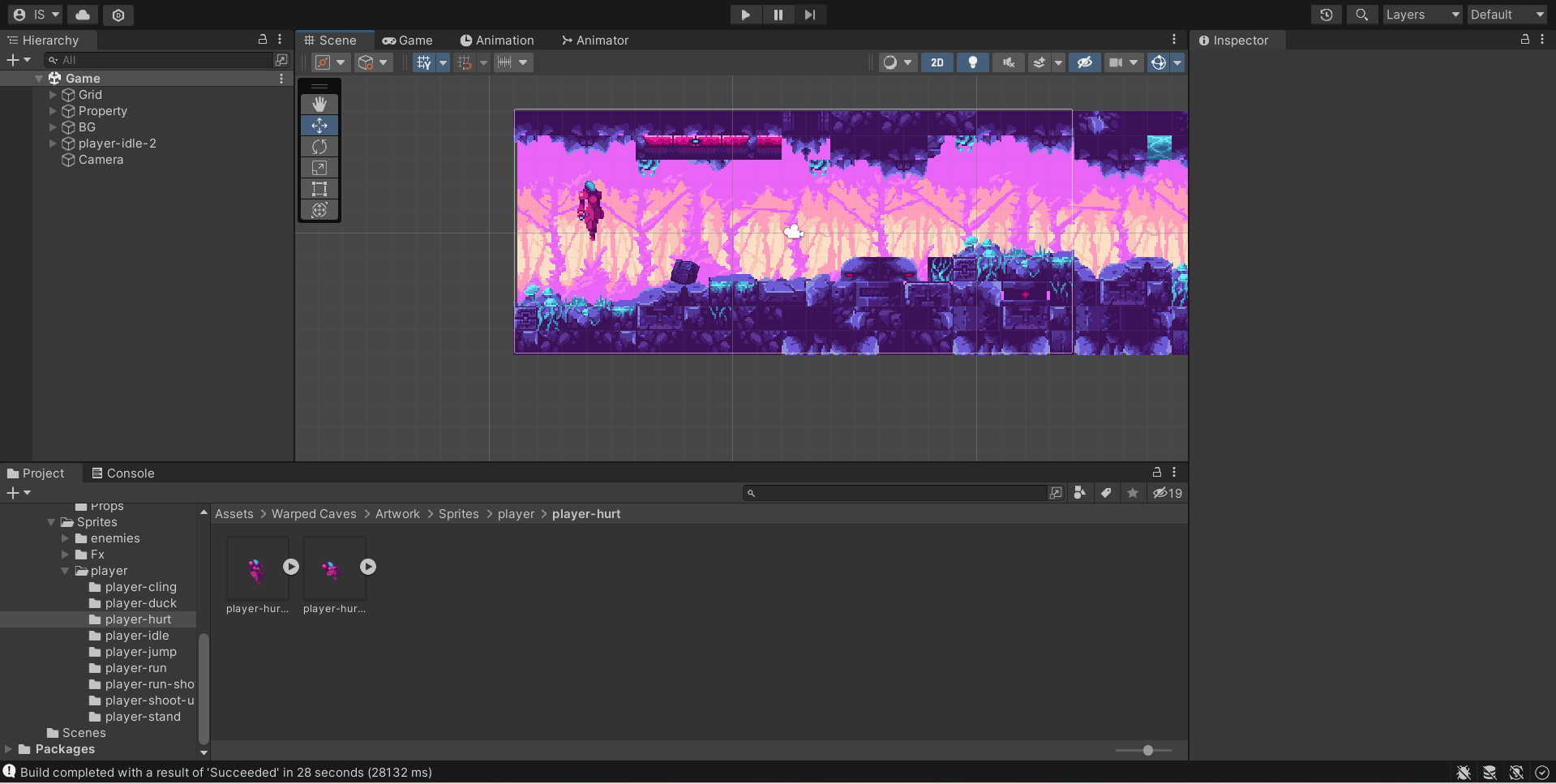
# 10 MEMBUAT KARAKTER

|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118112 |
| **Nama** | : | I Kadek Oka Sukrisna |
| **Kelas** | : | B |
| **Asisten Lab** | : | Mohammad Akbar Ilham 2118091 |

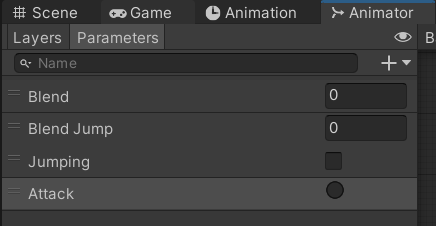
## Tugas 10 : Judul Tugas 1

1. **Membuat mekanisme Attack**
2. Buka Project Bab 9 Untuk melanjutkannya.



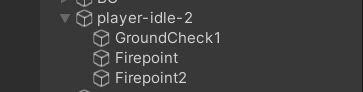
Gambar 10. 1 Membuka project bab9

1. Kemudian pada menu Tab **Animator** Tambahkan Parameter Trigger, Rename Menjadi ***Attack***



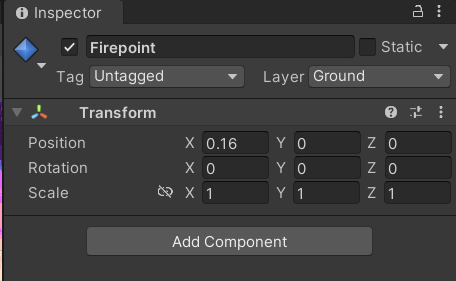
Gambar 10. 2 menambahan parameter attack

1. Setelah menambahkan 2 parameter Attack, Langkah selanjutnya adalah membuat Layer *Game object* baru didalam *player-idle-1*, Klik kanan pilih *Create Empty* lalu Rename menjadi *Firepoint dan Firepoint2*



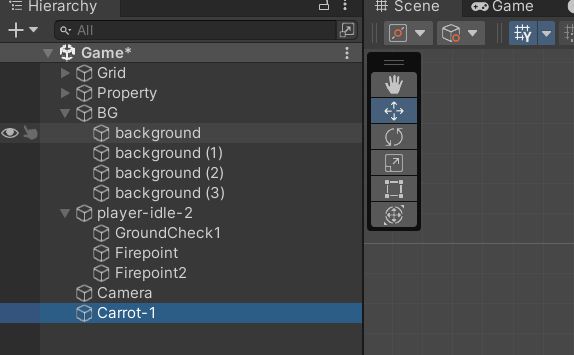
Gambar 10. 3 menambahakan 2 firepoint

1. Pada menu *Hierarchy* klik Firepoint untuk setting pada Inspector, Ubah *Icon* Menjadi titik, atur letak titik didepan dan belakang player



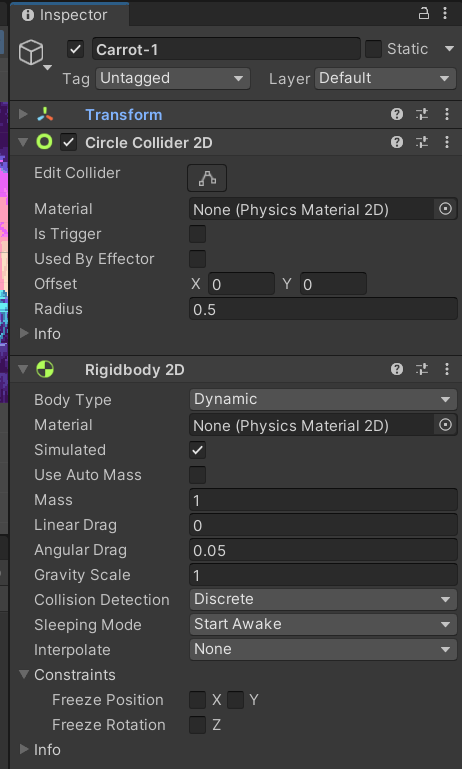
Gambar 10. 4 Mengatur titik Firepoint

1. Pada menu Hierarchy Tambahkan carrot-1, di folder Sprites > Misc > carrot > carrot-1, *rename* menjadi *fireball*



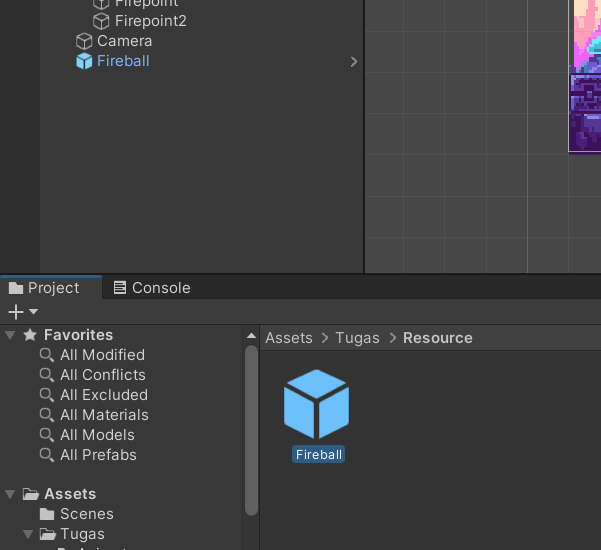
Gambar 10. 5 menambahkan fireball

1. Klik fireball(carrot-1) untuk menambahkan Component Circle Collider 2d, dan Riggidbody 2D, Setting sesuai gambar dibawah ini



Gambar 10. 6 menambahkan colide dan ridig body

1. Buat Folder baru *Resources* di menu Project, kemudian drag and drop fireball kedalam folder Resources, dan sembunyikan *fireball* pada *Hierarchy*

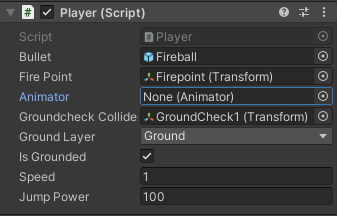


Gambar 10. 7 memindahkan lokasi fireball

1. Pada Script Player Tambahkan Script dibawah ini

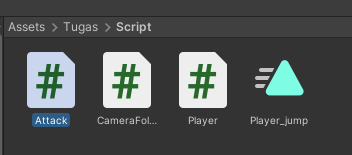
|  |
| --- |
| #Buat variabel baru  public Animator animator;  public GameObject bullet;  public Transform firePoint;  #Tambahkan dibawah fungsi fixedUpdate  IEnumerator Attack()  {  animator.SetTrigger("Attack");  yield return new WaitForSeconds(0.25f);  float direction = Mathf.Sign(transform.localScale.x);  GameObject fireball = Instantiate(bullet, firePoint.position, Quaternion.identity);  fireball.GetComponent<Rigidbody2D>().velocity = new Vector2(direction \* 10f, 0);  Destroy(fireball, 2f);  }  #Tambakan pada Function Void Update  if (Input.GetKeyDown(KeyCode.C))  {  StartCoroutine(Attack());  } |

1. Pada Inspector Player, Ubah seperti dibawah ini, Dimana Bullet berisi object yang akan ditembak sedangkan fire point adalah titik tembak pertama



Gambar 10. 8 import firepoint

1. Buat Script Attack pada folder Script

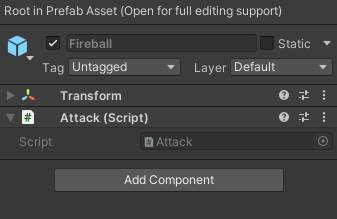


Gambar 10. 9 menambhakan file script

1. Tambahkan Script Attack dibawah ini

|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class Attack : MonoBehaviour  {  private void OnTriggerEnter2D(Collider2D collision)  {  if (collision.gameObject.CompareTag("Enemy"))  {  Destroy(gameObject);  Destroy(collision.gameObject);  }  }  } |

1. Didalam folder resource Tambahkan Script Attack di Prefab fireball, dengan cara Klik fireball kemudian pada menu Inspector arahkan Script Attack kedalam Inspector



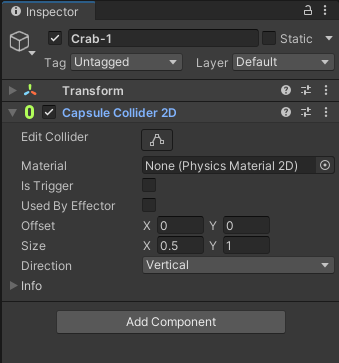
Gambar 10. 10 tambahkan attack pada inspector fireball

1. Tambahkan Enemy crab-1 pada hierarchy di folder Sprites> enemies>Crab



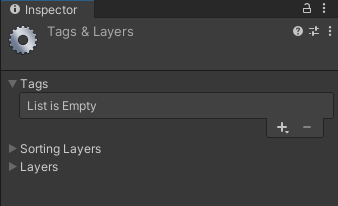
Gambar 10. 11 menambahkan enemy

1. Kemudian klik pada crab-1, lalu pada menu tab inspector tambahkan collider 2D untuk mendeteksinya



Gambar 10. 12 menambhakan capsule colide pada bee-2

1. Tambahkan Tag Enemy dengan cara Pilih Add Tag, kemudian add tag to the list, Tuliskan Enemy



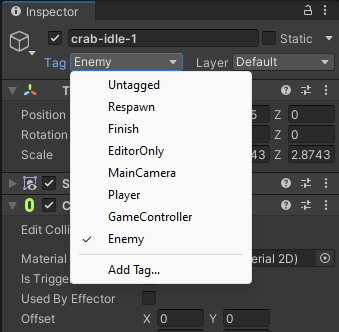
Gambar 10. 13 menambhakan tak enemy

1. Tembak Enemey dengan menekan Tombol C untuk menghancurkan musuh



Gambar 10. 14 hasil enemy

1. Tambahkan Tag Enemy dengan cara Pilih Add Tag, kemudian add tag to the list, Tuliskan Enemy



Gambar 10. 15 hasil enemy

1. Tembak Enemey dengan menekan Tombol C untuk menghancurkan musuh



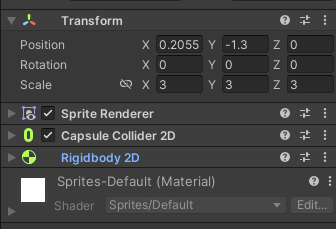
Gambar 10. 16 hasil enemy

1. **Membuat Enemy Behavior**
2. Cari sebuah sprite pack Bernama enemies dan buka folder bernama “jumper-idle-1” ,Tambahkan “jumper-idle-1” ke Hierarchy.



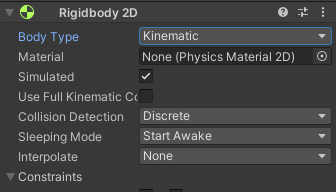
### jumper-idle-1

1. Tambahkan sebuah komponen bernama Capsule Colider 2D dan Rigidbody dalam inspector game objek jumper-idle-1

****

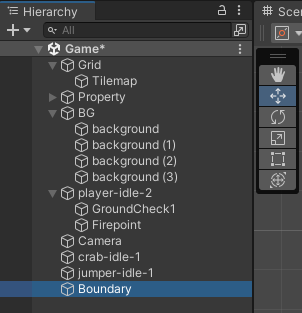
### Capsule Colider 2D

1. Atur sedikit collider tersebut seperti ukurannya diubah jika terlalu besar, dan pada *Body Type* Ubah menjadi *Kinematic*

****

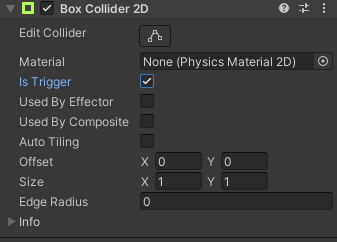
### Menambahkan Body “kinematics”

1. Create Empty object pada Hierarchy, Rename Menjadi *Boundary*

****

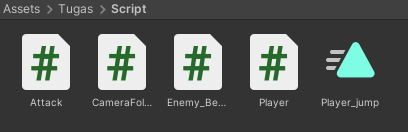
### Menambahkan boundary

1. Tambahkan Box Collider 2d pada Boundary, centang pada Is Trigger lalu atur sesuai keinginan pada size dan offside

****

### Box collider 2D

1. Buat sebuah file script didalam folder Script beri nama “Enemy\_Behavior”, kemudian drag dan masukkan ke dalam game object “jumper”

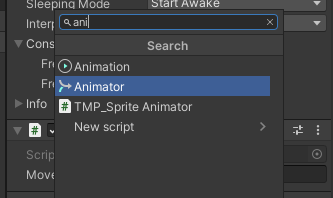
****

### Memuat Enemy\_behaviour

1. Tambahkan Script dibawah ini

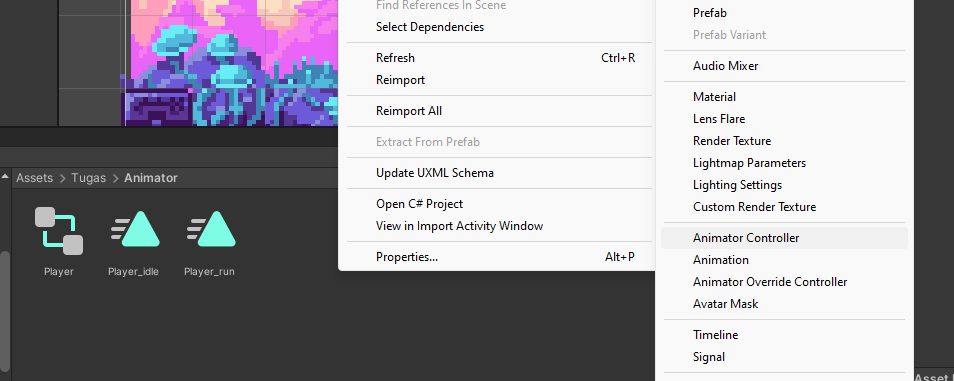
|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class Enemy\_Behavior : MonoBehaviour  {  [SerializeField] float moveSpeed = 1f;  Rigidbody2D rb;  void Start()  {  rb = GetComponent<Rigidbody2D>();  }  void Update()  {  if (isFacingRight())  {  rb.velocity = new Vector2(moveSpeed, 0f);  }  else  {  rb.velocity = new Vector2(-moveSpeed, 0f);  }  }  private bool isFacingRight()  {  return transform.localScale.x > Mathf.Epsilon;  }  private void OnTriggerExit2D(Collider2D collision)  {  transform.localScale = new Vector2(-transform.localScale.x, transform.localScale.y);  }  } |

1. Buat animasi untuk jumper-idle-1, klik jumper-idle-1, tambahkan component Animator.



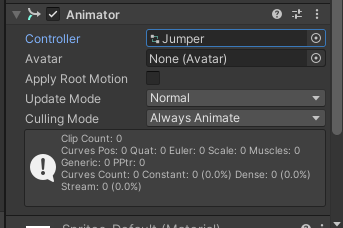
### Componen Animator

1. Buat animator di bagian animator, klik kanan pada folder animator > animator controller , berinama jumper



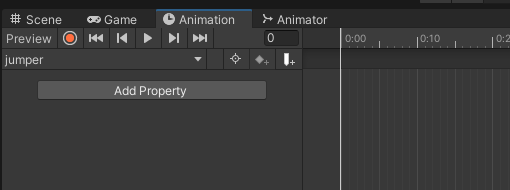
### Membuat animator Controller

1. Buka inspector jumper-idle-1, pada animator inport controller jumper yang sudah dibuat.



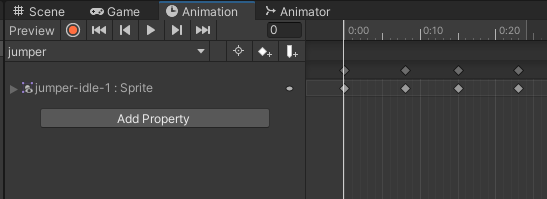
### add controller jumper

1. Kemudian tambah animasi dengan klik create pada animation, kemudian save pada folder animator dengan nama jumper.



### Create animation

1. Kemudian drag and drop animation jumper-idle-2 hingga jumper-idle-4 ke dalam add property



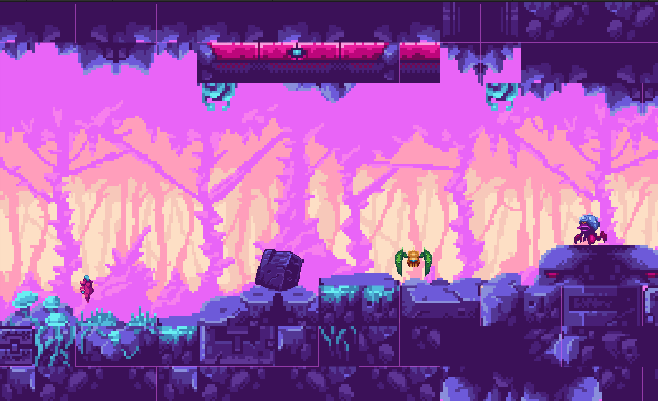
### Import slug ke property

1. Maka ketika di game di play akan terdapat animasi pada property slug.



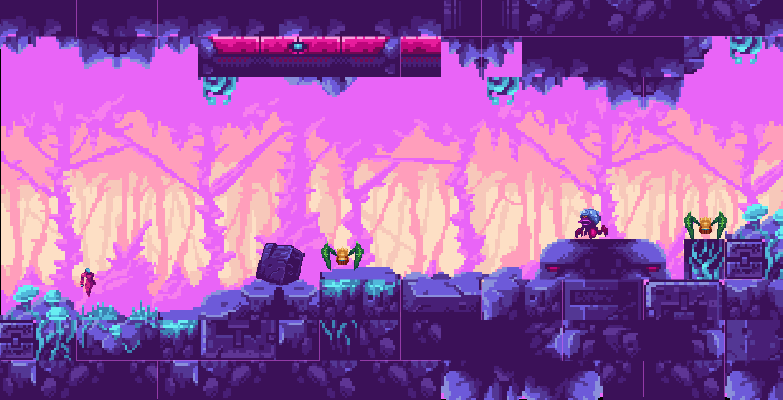
### Animasi slug

1. Duplikat slug ,dan boundary dan posisikan ke tempat yang diinginkan.

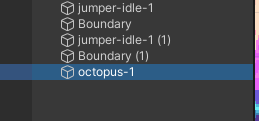


### Duplikat jumper dan boundary

1. Ketika di play akan mendapat 2 buah animasi slug.

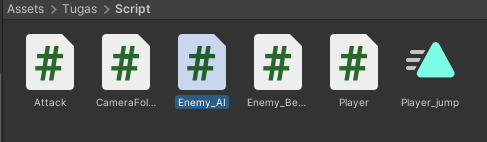


1. **Membuat Enemy Behavior**
2. Cari sebuah sprite pack bernama 'enemy' dan buka folder bernama 'octopus-1'. Tambahkan ''octopus-1' pada Hierarchy



### Menambahkan property octopus-1

1. Buat Script Enemy\_AI pada folder Tugas - Script

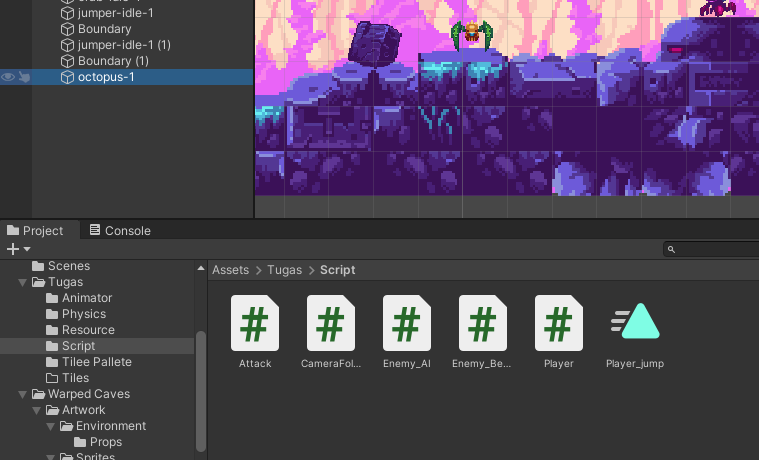


### Script Enemy\_Ai

1. Tambahkan Script dibawah ini

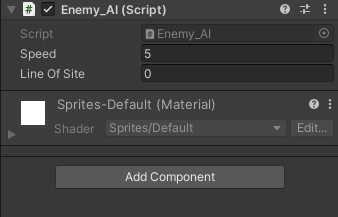
|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class Enemy\_AI : MonoBehaviour  {  public float speed; // Kecepatan gerakan musuh  public float lineOfSite; // Jarak penglihatan musuh  private Transform player; // Transform dari pemain  private Vector2 initialPosition; // Posisi awal musuh  // Use this for initialization  void Start()  {  // Mencari pemain berdasarkan tag  player = GameObject.FindGameObjectWithTag("Player").transform;  // Menyimpan posisi awal musuh  initialPosition = GetComponent<Transform>().position;  }  // Update is called once per frame  void Update()  {  // Menghitung jarak antara musuh dan pemain  float distanceToPlayer = Vector2.Distance(player.position, transform.position);  // Jika pemain berada dalam jarak penglihatan musuh  if (distanceToPlayer < lineOfSite)  {  // Musuh bergerak menuju pemain  transform.position = Vector2.MoveTowards(this.transform.position, player.position, speed \* Time.deltaTime);  }  else  {  // Musuh kembali ke posisi awal  transform.position = Vector2.MoveTowards(transform.position, initialPosition, speed \* Time.deltaTime);  }  }  // Untuk menggambar jarak penglihatan musuh di editor  private void OnDrawGizmosSelected()  {  Gizmos.color = Color.red;  Gizmos.DrawWireSphere(transform.position, lineOfSite);  }  } |

1. Kemudian drag and drop script pada bee-1



### Drag and drop script ke octopus-1

1. Pada Inspector Enemy\_Ai, Atur Speed juga Line of Site untuk menentukan jarak dan speed pada enemy



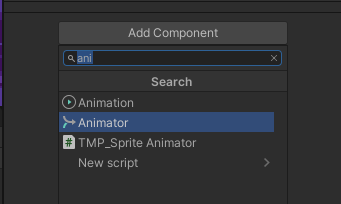
### Mengatur speed dan line of site

1. Running Game, maka bee akan mengikuti Gerakan Player



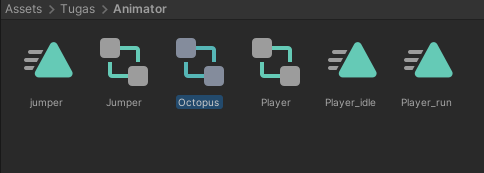
### Running game

1. Buat animasi untuk bee, masuk ke dalam inspector octopus > tambahkan component animator.



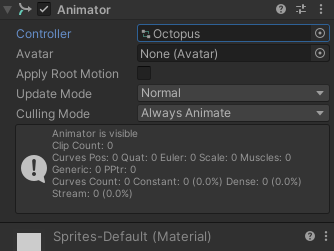
### component animator octopus

1. Tambahkan animator bee pada folder animator.



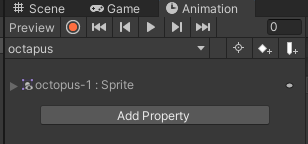
### Menambahkan animator octopus

1. Kemudian import animator yang sudah di baut ke dalam animator inspector octopus, di bagian controller.



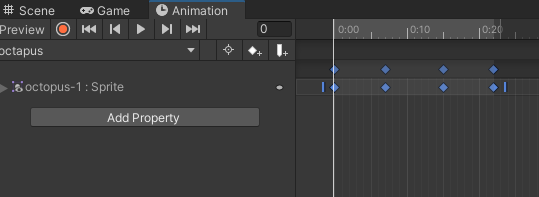
### Import animator controller

1. Kemudian tambah animasi dengan klik create pada animation, kemudian save pada folder animator dengan nama octopus.



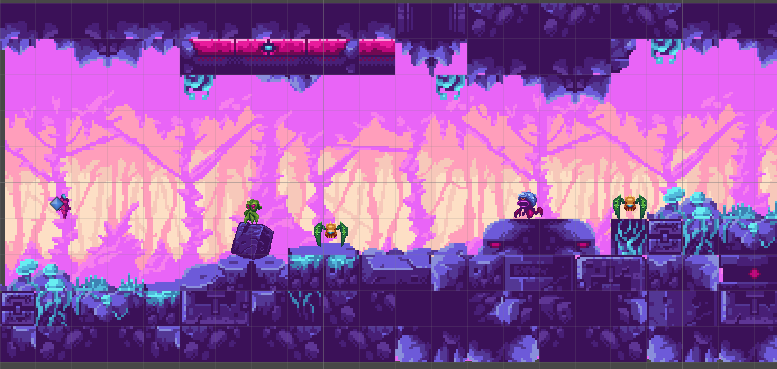
### Create new animator octopus

1. Kemudian drag and drop animation octopus-2 hingga octopus-4 ke dalam add property.



### Drag and drop amination octopus

1. Running game, maka bee akan mengikuti Gerakan player.



### Jalankan game

1. **Respawnbt**
2. Buka file script (Player.cs) tambahkan variabel nyawa seperti dibawah ini

|  |
| --- |
| public int nyawa;  [SerializeField] Vector3 respawn\_loc;  public bool play\_again; |

1. Tambahkan kode dibawah ini untuk mengatur posisi respawn sesuai dengan posisi awal permainan dimulai, tambahkan source code di function Awake()

|  |
| --- |
| respawn\_loc = transform.position; |

1. Tambahkan kode dibawah ini di dalam void update Player.cs agar ketika nyawa player dibawah 0 maka akan melakukan respawn

|  |
| --- |
| if(nyawa < 0){  playagain();  } |

1. Tambahkan juga kode berikut dibawah code sebelumnya agar ketika player jatuh dibawah platform akan melakukan respawn

|  |
| --- |
| if(transform.position.y < -10)  {  play\_again = true;  playagain();  } |

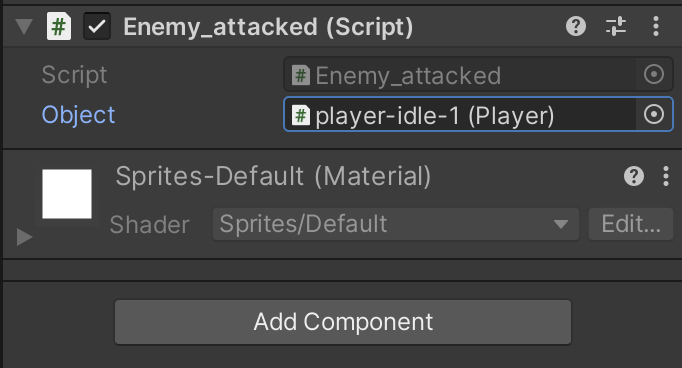
1. Tambahkan fungsi playagain() dalam script Player.cs

|  |
| --- |
| void playagain(){  if(play\_again == true){  nyawa = 3;  transform.position = respawn\_loc;  play\_again = false;  }  } |

1. Tambahkan file script (Enemy\_Attacked.cs) dan isikan source code dibawah ini

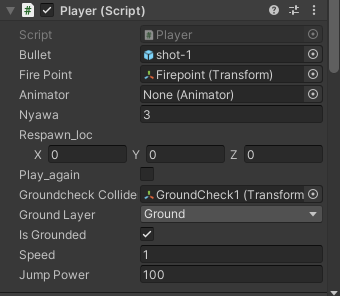
|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class Enemy\_attacked : MonoBehaviour  {  [SerializeField] private Player Object;  void Start()  {  if (Object == null)  {  Object = GameObject.FindWithTag("Player").GetComponent<Player>();  }  }  void OnTriggerEnter2D(Collider2D other)  {  if (other.CompareTag("Player"))  {  Object.nyawa--;  if (Object.nyawa < 0)  {  Object.play\_again = true;  }  }  }  } |

1. Pada hierarchy bee-1 Tambahkan Script enemy\_attack, arahkan object pada player-idle-1



### Enemy attack

1. Klik game object Player, pergi ke Inspector dan ubah nilai Nyawa menjadi 3 pada Player(Script)



### Healt player

1. Jika di play, Player mengenai atau menyentuh opposum-1 sebanyak 3 kali maka nyawa akan berkurang 1 dan jika nyawa kurang dari 0 maka akan reswpawn ke titik awal.



### Play games

1. **Link Github Pengumpulan**

Link…