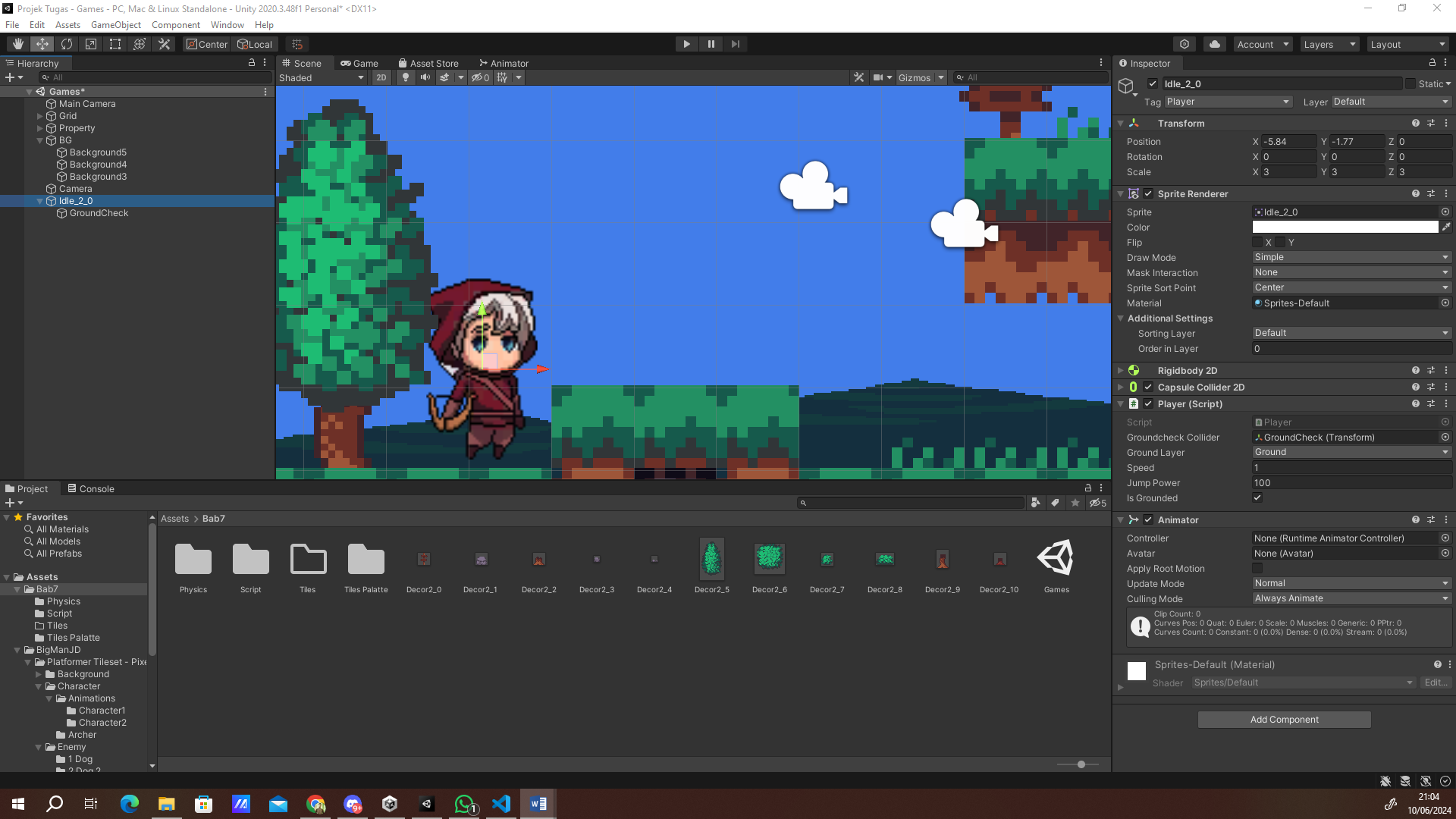
# 10 RESPAWN AND AI ENEMY ATTACK

|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118043 |
| **Nama** | : | Kurniawan Yoga Pratama |
| **Kelas** | : | C |
| **Asisten Lab** | : | Difa Fisabililah (2118052) |

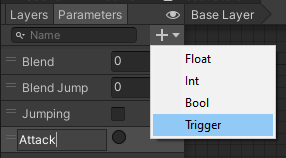
## 10.1 Tugas 1: Character Movement

1. Buat Pergerakan Karakter
   * + 1. Buka kembali *project* yang kemarin kita buat melalui *Unity HUB*

****

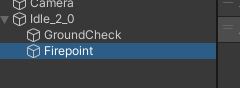
### 10.1 Tampilan Membuka File Unity

* + - 1. Pergi ke tabb animator, tambahkan parameter dengan mana Attack dengan tipe data trigger.



### 10.2 Tampilan Menambahkan Parameter Attack

* + - 1. Selanjutnya buat layer game object baru di dalam Idle\_0, klik kanan pilih Create Empty dan ubah nama menjadi Firepoint.



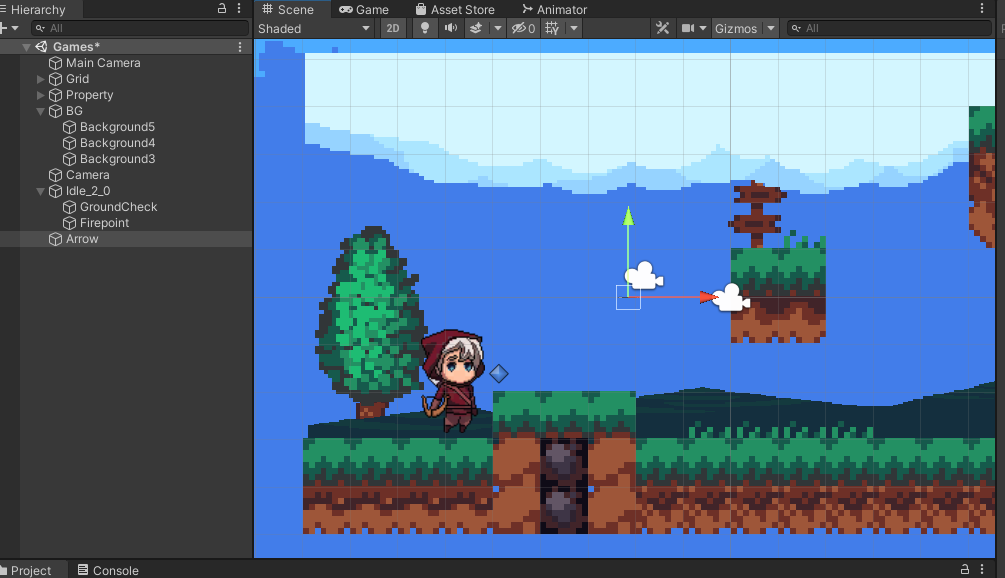
### Gambar 10.3 Tampilan Menambahkan Firepoint

* + - 1. Kemudian atur icon Firepoint menjadi titik melalui tab Inspector dan letakkan di depan player.



### Gambar 9.4 Tampilan Mengatur Firepoint

* + - 1. Tambahkan satu item sebagai peluru beri nama menjadi arrow.



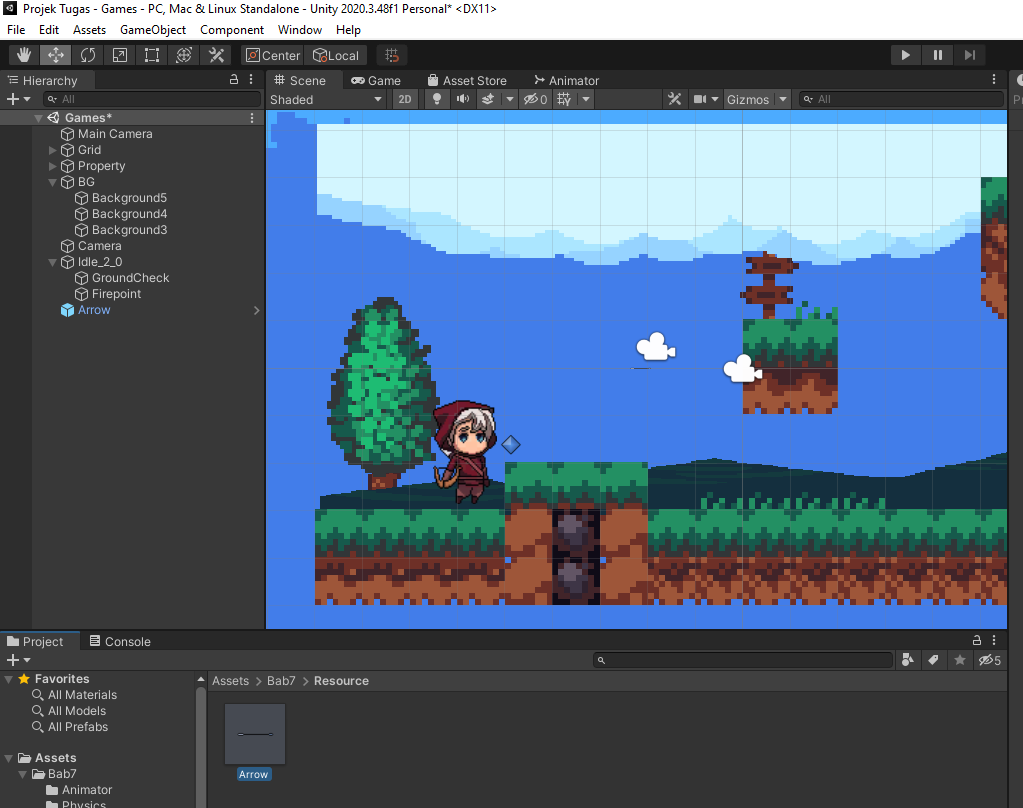
### Gambar 9. 5 Tampilan Menambahkan Arrow

* + - 1. Tambahkan comoponen *Circle Collider 2D* dan *Rigidbody 2D*



### Gambar 9. 6 Tampilan Menambahkan Componen Circle Collider 2D

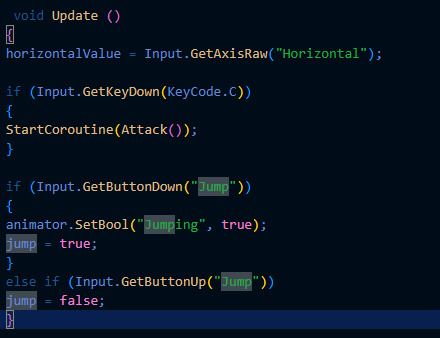
* + - 1. *Drag and drop Arrow* pada *folder Resource* yang dibuat pada folder BAB7, lalu hapus arrow pada Hierarchy.



### Gambar 9. 7 Tampilan Menambahkan Arrow ke Folder Resource

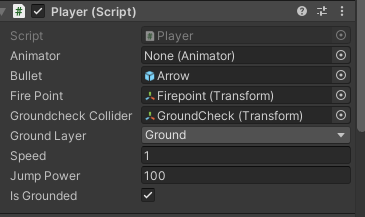
* + - 1. Tambahkan source code sepert ini pada deklarasi dan void update

|  |
| --- |
| public Animator animator;  public GameObject bullet;  public Transform firepoint   1. void Update () 2. { 3. horizontalValue = Input.GetAxisRaw("Horizontal"); 5. if (Input.GetKeyDown(KeyCode.C)) 6. { 7. StartCoroutine(Attack()); 8. }    if (Input.GetButtonDown("Jump"))   {   animator.SetBool("Jumping", true); jump = true;}   else if (Input.GetButtonUp("Jump")   jump = false;   } |



### Gambar 9. 8 Tampilan Syntax Update

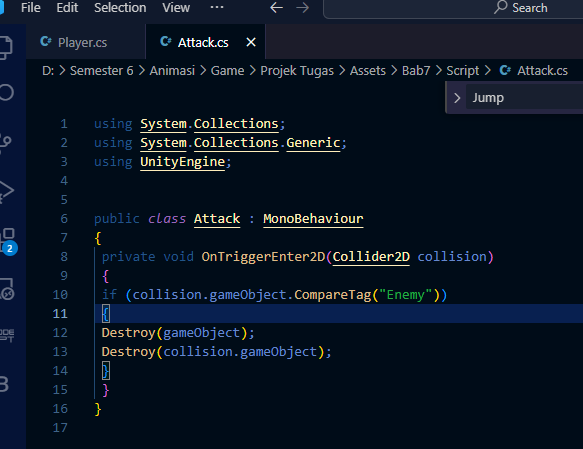
* + - 1. Pada inspector drag and drop arrow pada folder resource ke tab bullet lalu firepoint ke firepoint juga



### Gambar 9.9 Tampilan Inspector add Bullet

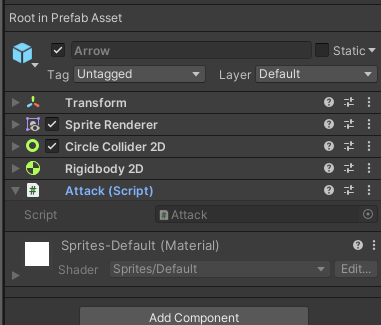
* + - 1. Buat script Attack pada folder Script dan tambahkan script berikut.

|  |
| --- |
| 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);  }  }  } |

****

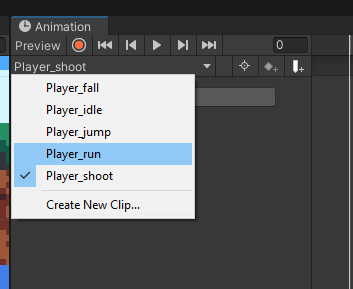
### Gambar 9.10 Tampilan Menambahkan Script Attack

* + - 1. Drag and drop script attack ke inspector arrow yang ada di folder resource.



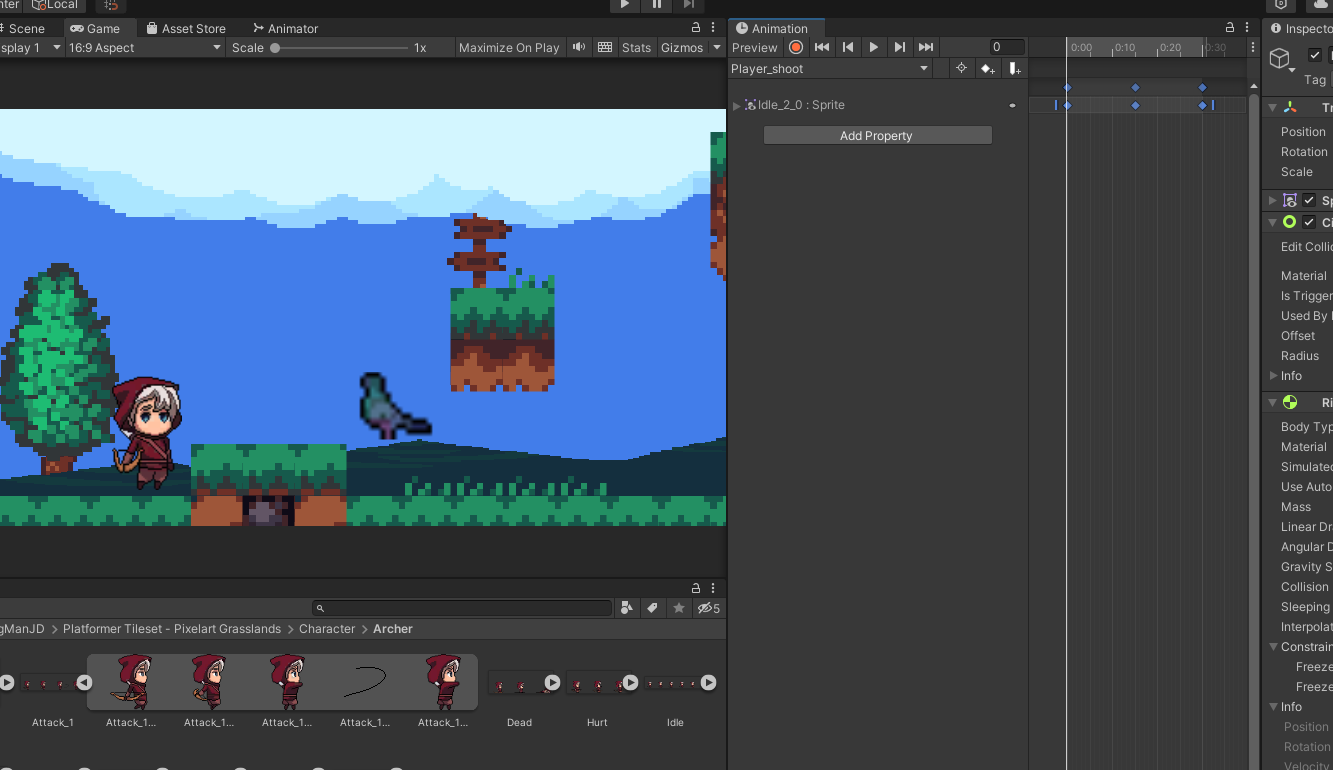
### Gambar 9. 11 Tampilan Menambahkan Script Attack Di Arrow

* + - 1. Tambahkan animasi ketika menembak, dengan menambah new clip dengan nama player\_shoot.



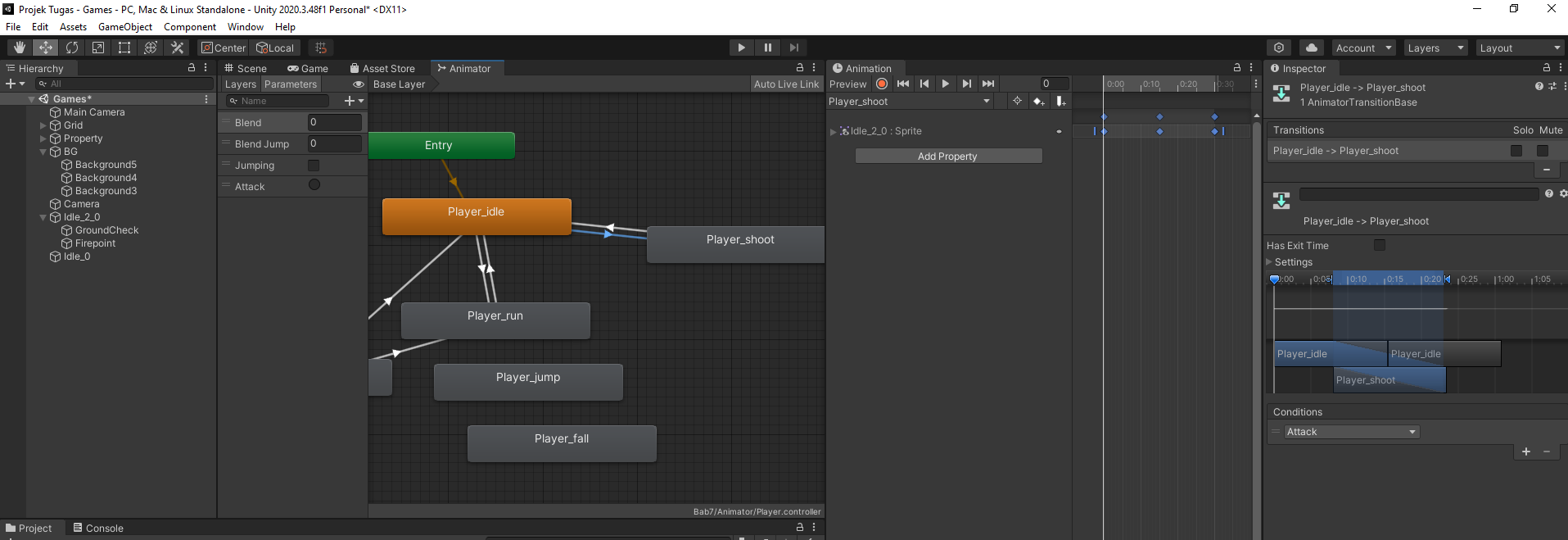
### Gambar 9. 12 Tampilan Menambahkan Player Shoot

* + - 1. Drag and Drop animasi tke tab animation



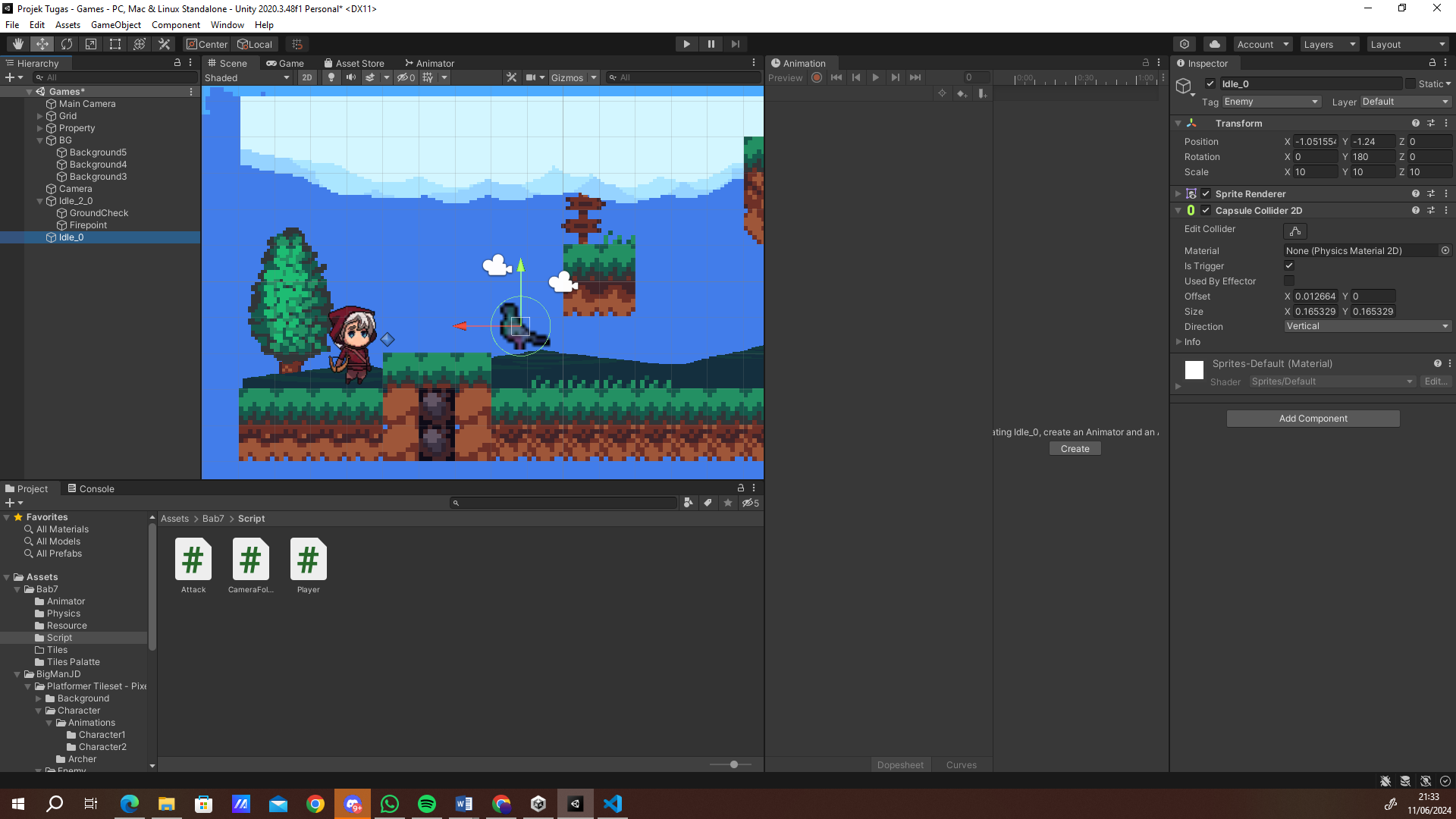
Gambar 9. 13 Tampilan Menambahkan Animasi Shoot

* + - 1. Make Transition dari player idle to player shoot begitu pula sebaliknya.



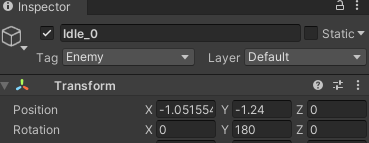
Gambar 9. 13 Tampilan Menambahkan Transisi Idle Ke Shoot

* + - 1. Tambahkan 1 enemy disini saya memlilh burung sebagai enenmy. Lalu tambahkan component capsule collider 2d pada enemy



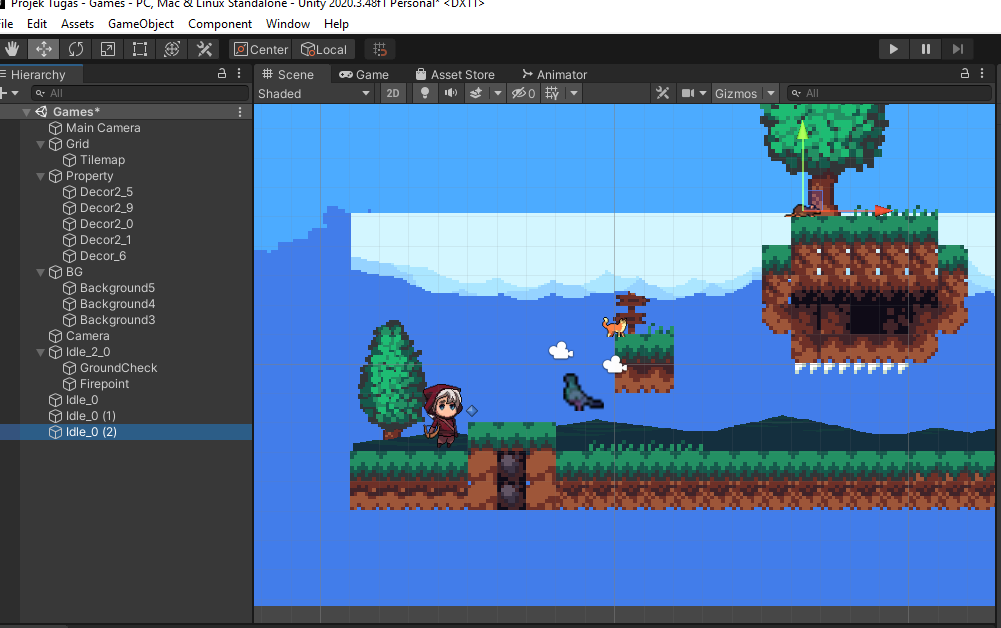
### Gambar 9. 12 Tampilan Menambahkan Enemy Burung 1

1. Buat tag baru bernama enemy, lalu ganti tag menjadi enemy.



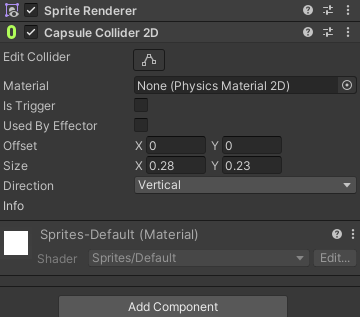
### Gambar 9. 13 Tampilan Menambahkan Tag Enemy

1. Enemy Behavior
2. Tambahkan Enemy cat dan rat sebagai enemy behaviour.



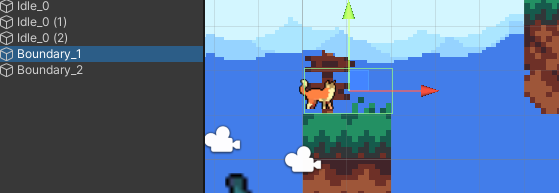
### Gambar 9. 13 Tampilan Menambahkan Enemy Cat

1. Tambahkan *capsule collider* pada masing masing enemy yang barusan ditambahkan.



### Gambar 9. 15 Tampilan Menambahkan Komponen

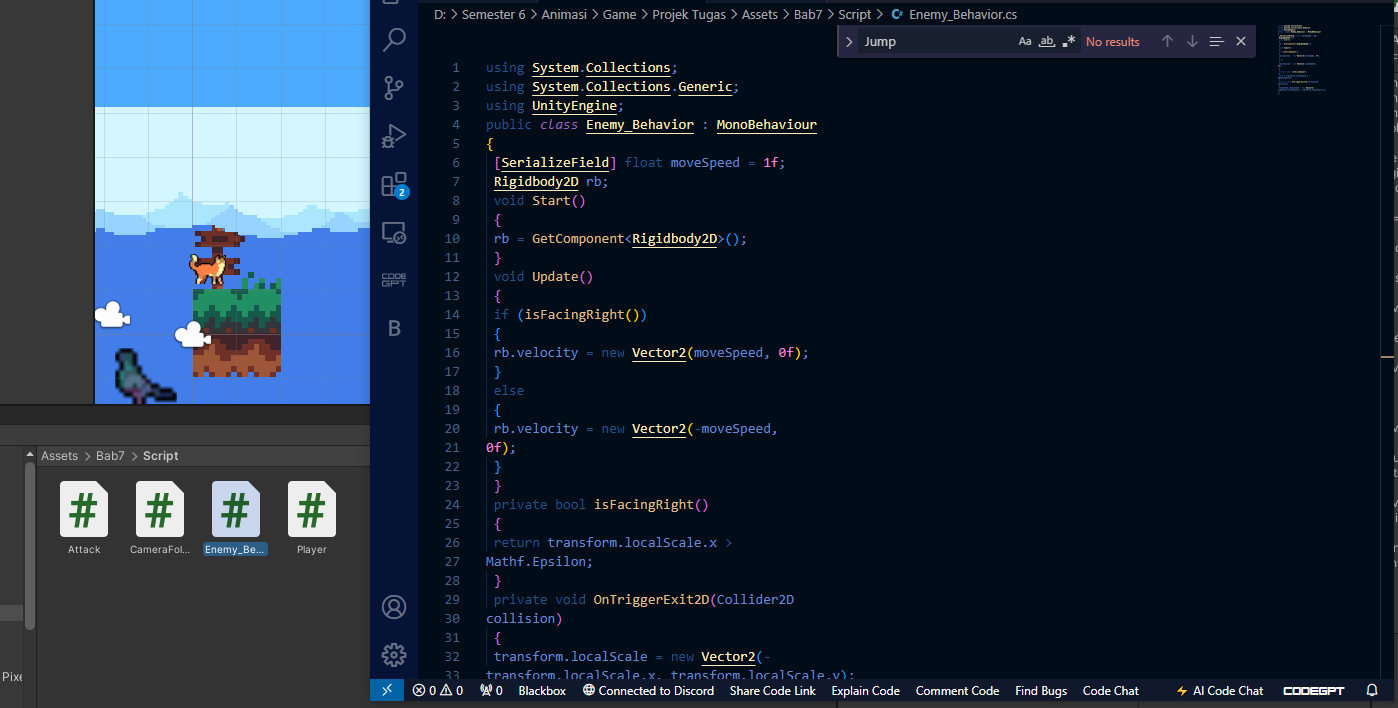
1. Create empty pada hierarchy lalu baru nama Boundary 1 dan boundary 2, lalu tambahkan component box collider sebagai batasan enemy bergerak.



### Gambar 9. 1 Tampilan Menambahkan Boundary

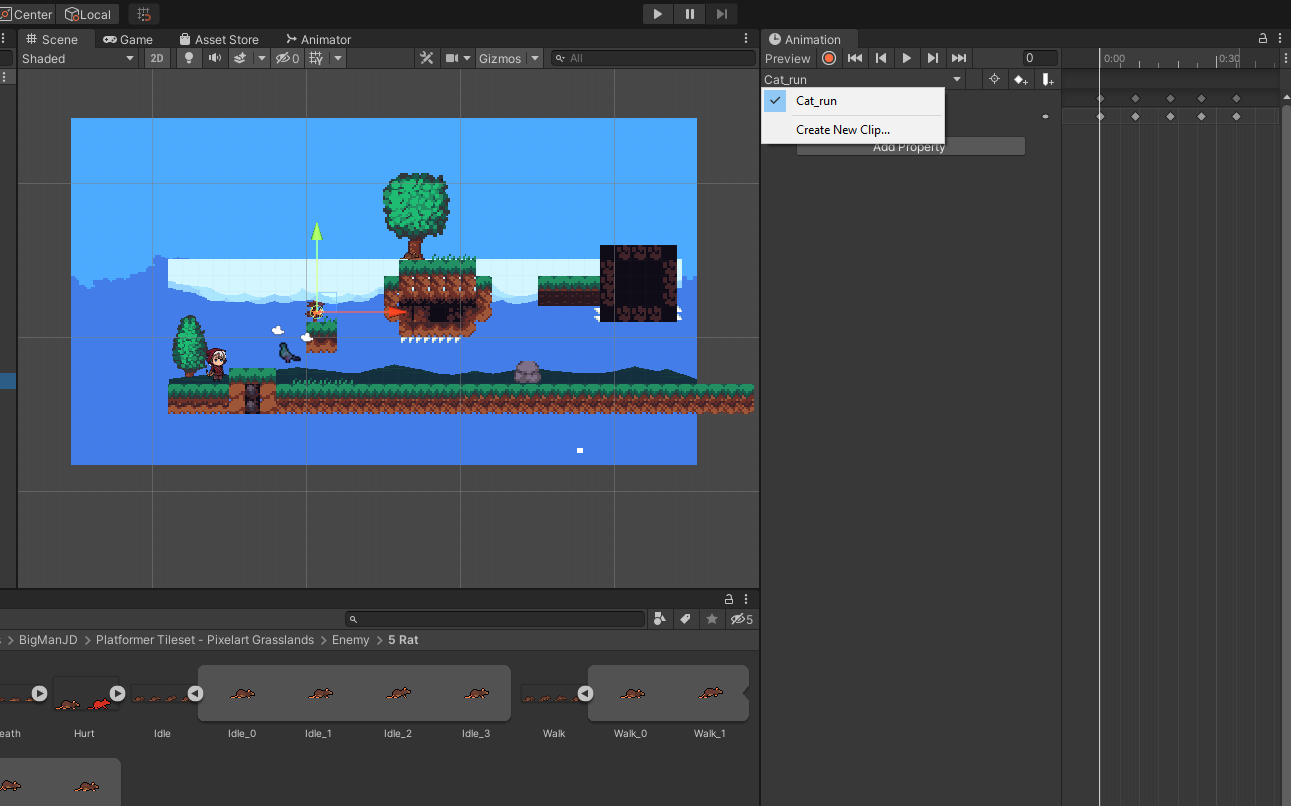
1. Tambahkan Script dengan nama enemy\_behavior, lalu drag and drop ke kedua enemy yang tadi ditambahkan

|  |
| --- |
| 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);  }  } |



### Gambar 9. 2 Tampilan Memanbahkan Script Enemy Behavior

1. Lalu tambahkan animasi untuk enemy bergerak, lalu drag and drop animasi ke klip, maka enemy saat berjalan akan ada animasi berjalannya.



### Gambar 9. 18 Tampilan Menambahkan Animasi Enemy

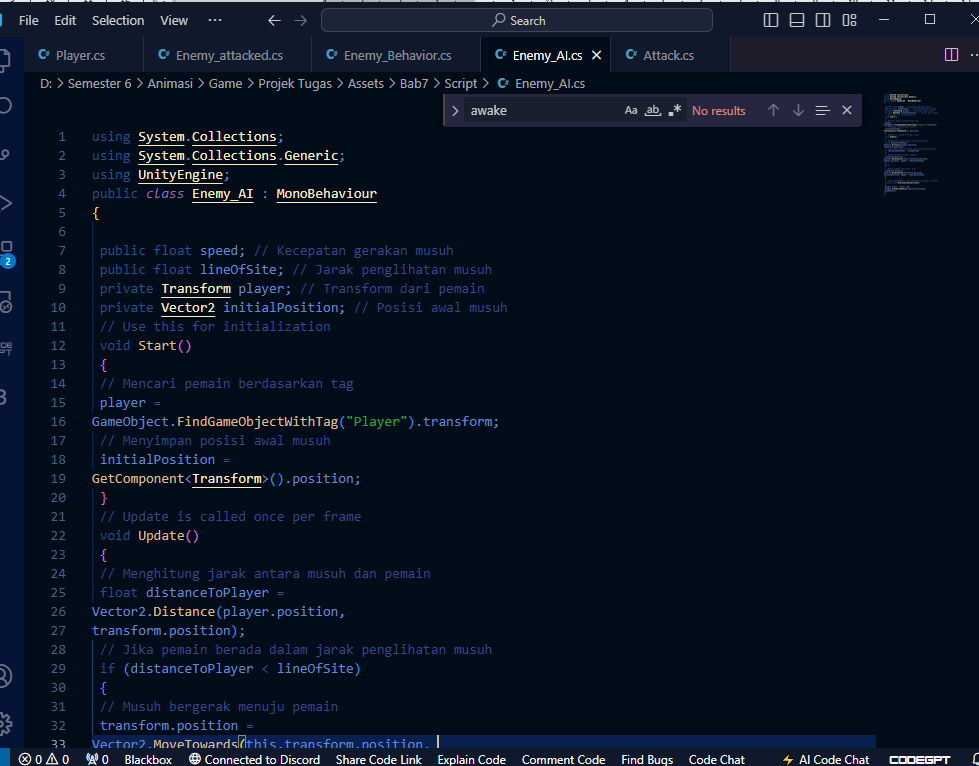
1. Membuat Enemy AI
2. Tambhakan satu enemy lagi sebagai enemy AI.



Gambar 9. 20 Tampilan Menambahakan Enemy Burung 2

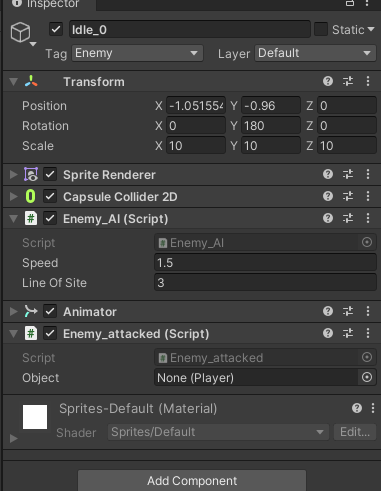
1. Buat script Enemy\_AI dan tambahkan script berikut

|  |
| --- |
| 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);  }  } |



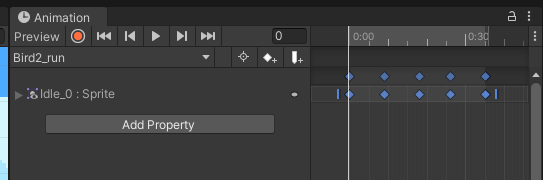
Gambar 9. 20 Tampilan Menambahkan Script Enemy\_AI

1. Drag and drop script tersebut ke enemy burung 1 dan 2, lalu atur speed dan line off site untuk mengatur kecepatan dan jarak.



Gambar 9. 20 Tampilan Drag and Drop Script

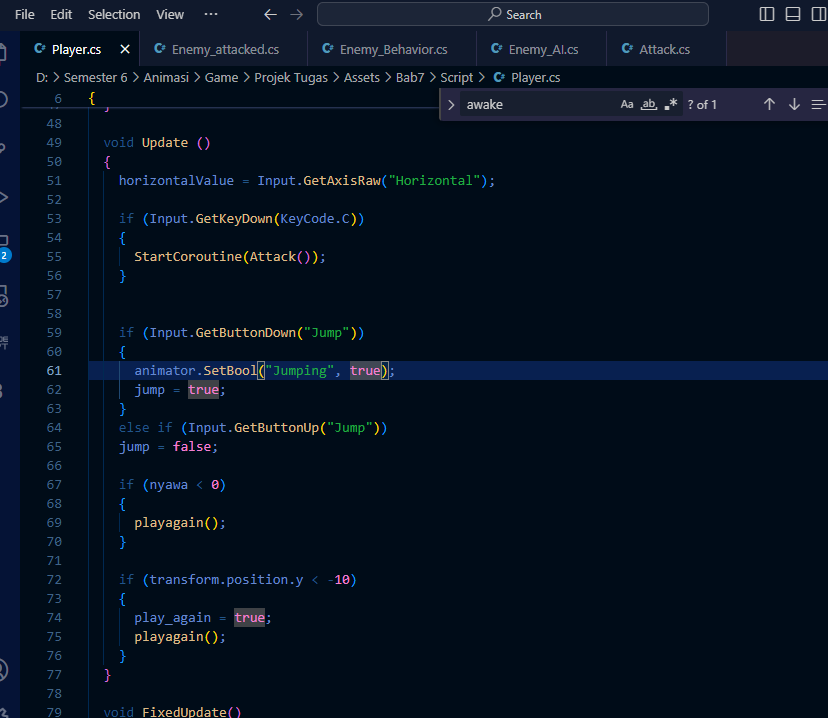
1. Tambahkan animasi bergerak pada ke dua enemy dengan membuat clip baru dan tambahkan animasi sprite bergerak ke dalam timeline serta atur waktunya.



Gambar 9. 20 Tampilan Menambahkan Animasi Enemy Burung

1. Respawn
2. Tambahkan script berikut ke dalam script Player untuk mengembalikan player ke tempat awal ketika jatuh atau nyawa kurang dari 0

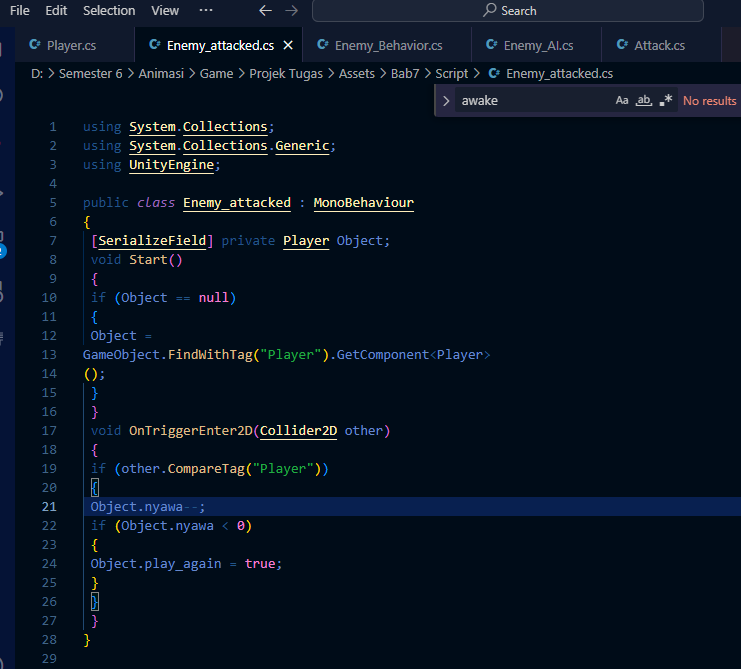
|  |
| --- |
| public int nyawa;  [SerializeField] Vector3 respawn\_loc;  public bool play\_again;  private void Awake()  {  rb = GetComponent<Rigidbody2D>();  animator = GetComponent<Animator>();  respawn\_loc = transform.position;  }  void playagain()  {  if (play\_again == true)  {  nyawa = 3;  transform.position = respawn\_loc;  play\_again = false;  }  }  if (nyawa < 0)  {  playagain();  }  if (transform.position.y < -10)  {  play\_again = true;  playagain();  } |



### Gambar 9. 20 Tampilan Script Respawn

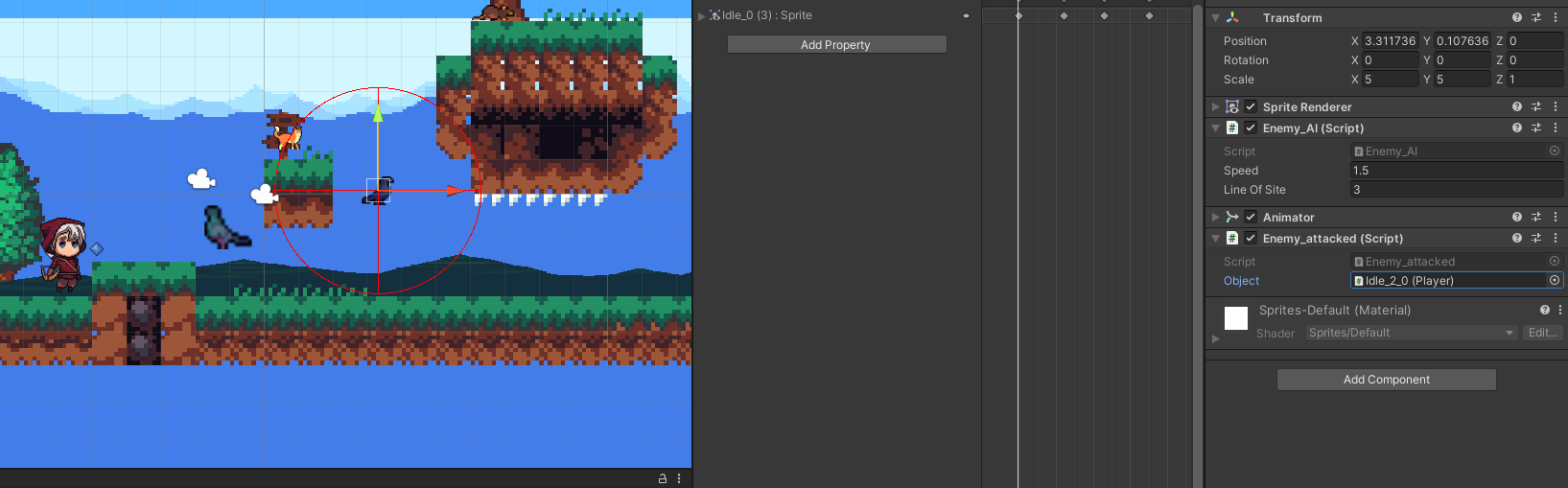
1. Kemudian buat script baru dengan nama Enemy\_Attack dan tambahkan script berikut.

|  |
| --- |
| 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;  }  }  }  } |



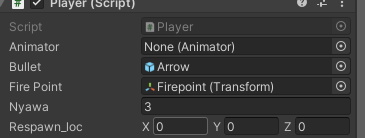
### Gambar 9. 21 Tampilan Script Enemy\_Attack

1. Drag and drop script Enemy\_Attack ke semua musuh dan arahkan object pada Idle\_2\_0(player)



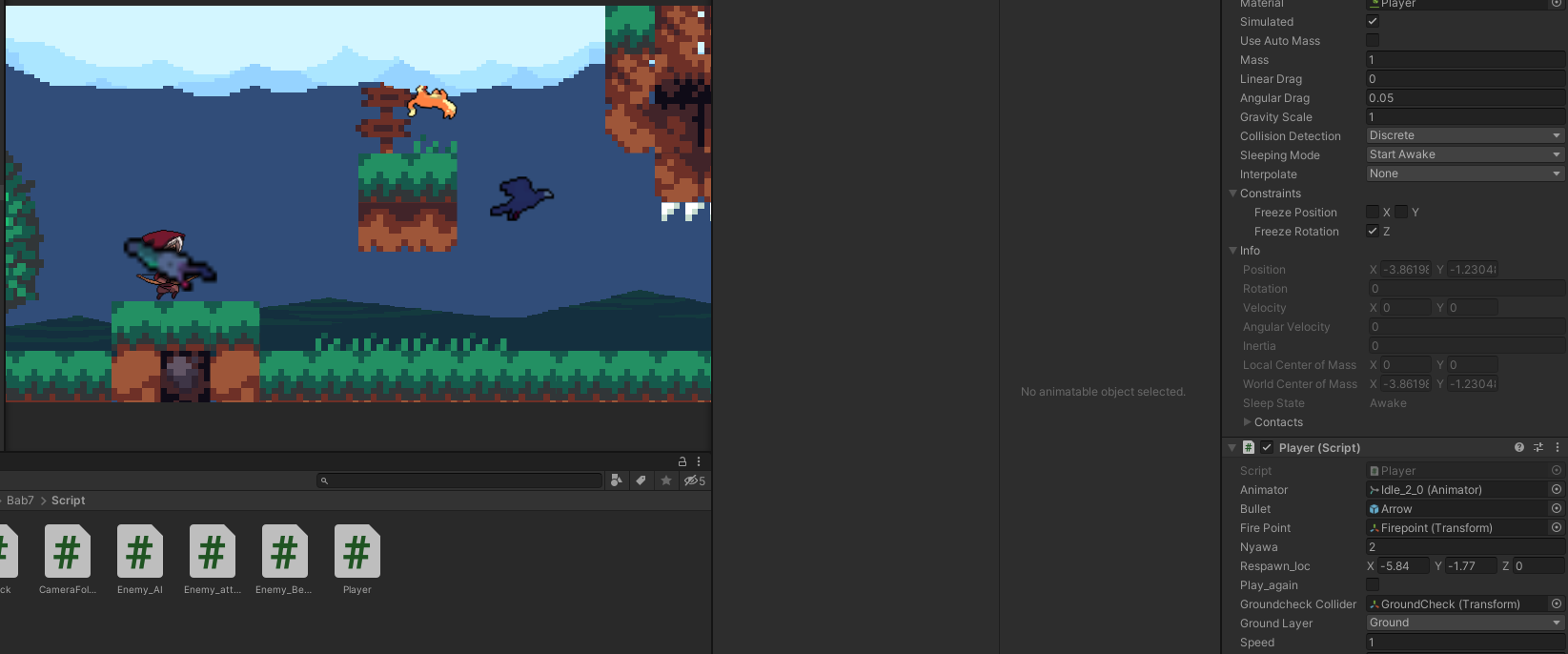
### Gambar 9. 22 Tampilan Inspector Script Enemy\_Attack

1. Pergi ke tap Inspector game object Player dan ubah nilai Nyawa menjadi 3 pada Player (Script).



### Gambar 9. 22 Tampilan Inspector Nilai Nyawa Player

1. Lalu coba jalankan game, maka ketika player menyentuh musuh, nyawa akan berkurang 1 dan jika nyawa kurang dari 0 atau player jatuh maka akan respawn ke titik awal.



Gambar 9. 22 Tampilan Isi dari comoponent camera

1. Kuis : Melengkapi Source code

|  |
| --- |
| using UnityEngine;  public class PlayerAttack : MonoBehaviour  {  public float attackRange = 2.0f; // Changed type to float for precision  public int attackDamage = 10;  void Update()  {  if (Input.GetButtonDown("Fire1"))  {  PerformMeleeAttack();  }  }  void PerformMeleeAttack()  {  RaycastHit hit;  if (Physics.Raycast(transform.position, transform.forward, out hit, attackRange))  {  EnemyHealth enemy = hit.transform.GetComponent<EnemyHealth>();  if (enemy != null)  {  enemy.TakeDamage(attackDamage);  }  }  }  } |

Penjelasan:

Script ini memiliki dua kelas utama yaitu PlayerAttack dan KesehatanMusuh. Pada PlayerAttack, terdapat variabel jarakSerang untuk menentukan jarak serangan dan seranganDamage untuk menentukan besaran damage. Di metode Update, jika tombol "Fire1" ditekan, metode LakukanSeranganMelee akan dipanggil. Metode ini menggunakan Physics.Raycast untuk mendeteksi objek dalam jarak serang, jika objek tersebut memiliki komponen KesehatanMusuh, maka metode TerimaDamage dipanggil untuk mengurangi kesehatan musuh. Pada kelas KesehatanMusuh, terdapat variabel kesehatan dan metode TerimaDamage yang mengurangi nilai kesehatan sesuai damage yang diterima, serta metode Mati yang akan dipanggil jika kesehatan mencapai nol atau kurang untuk menangani kematian musuh, seperti menghancurkan objek musuh

**Link Github Pengumpulan**

[**https://github.com/Yktama18/2118043\_PRAK\_ANIGAME.git**](https://github.com/Yktama17/2118043_PRAK_ANIGAME.git)