# 10 RESPAWN AND AI ENEMY ATTACK

|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118006 |
| **Nama** | : | Mikhael Christian |
| **Kelas** | : | A |
| **Asisten Lab** | : | Naufal Dhiaurrafif (2218059) |

## Tugas 10 : Membuat Langkah-Langkah

1. **Langkah-Langkah Membuat Mekanisme Attack**
2. Buka *Project Unity* sebelumnya.

**A screenshot of a video game

Description automatically generated**

### 10.1 Buka *Project* Sebelumnya

1. Kemudian pada *tab Animator* buat *parameter trigger* dan beri nama “Attack”. Setelah itu *Create Empty* pada Astronaut-idle dan beri nama “Firepoint”. Terakhir buka *tab inspector* dan pada bagian *Firepoint* ubah ikonnya serta letakkan di depan Astronaut-idle.

A screenshot of a video game

Description automatically generated

### 10.2 Tambahkan *Firepoint*

1. Tambahkan *item* pada *asset* menjadi peluru. Kemudian pada bagian *inspector* tambahkan komponen *Circle Collider* 2D dan *Rigidbody* 2D.

A screenshot of a video game

Description automatically generated

### 10.3 Tambahkan Peluru

1. Kemudian buka *script Player* dan ubah *source code* menjadi seperti di bawah ini.

|  |
| --- |
| #Pada class Player  public GameObject bullet;  public Transform firePoint;  #Tambahkan di bawah fungsi fixedUpdate  IEnumerator Attack()  {      animator.SetTrigger("Attack");      yield return new WaitForSeconds(0.25f);      float direction = 1f;    GameObject fireball = Instantiate(bullet, firePoint.position, Quaternion.identity);     fireball.GetComponent<Rigidbody2D>().velocity = new Vector2(direction \* 10f, 0);      Destroy(fireball, 2f);  }  #Tambahkan pada Function Void Update  if (Input.GetKeyDown(KeyCode.C))          {              StartCoroutine(Attack());          } |

1. Setelah itu pada *inspector* Astronaut-idle, pada bagian *Bullet* pilih objek *Lazer4*. Kemudian pada bagian *Fire Point* pilih *Firepoint*.

A screenshot of a computer

Description automatically generated

### 10.4 Atur *Player Script*

1. Buat *script* baru dengan nama “Attack” dan masukkan *source code* di bawah 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. Kemudian *drag and drop script* “Attack” ke dalam *Lazer4*.

A screenshot of a computer program

Description automatically generated

### 10.5 Tambahkan *Script Attack*

1. Kemudian tambahkan objek sebagai musuh. Pada bagian *inspector* tambahkan komponen *Capsule Collider* 2D serta ubah *tag*nya menjadi *enemy*.

A screenshot of a video game

Description automatically generated

### 10.6 Tambahkan *Enemy*

1. **Membuat Enemy Behavior NPC**
2. Selanjutnya tambahkan *enemy* baru, atur ukuran dan letakkan *enemy* pada posisi seperti di bawah ini.

A screenshot of a video game

Description automatically generated

### 10.7 Tambahkan *Enemy*

1. Pada *tab inspector*, tambahkan *Capsule Collider* 2D dan *Rigidbody* 2D. Kemudian atur *collider*nya agar sesuai.

A screenshot of a computer

Description automatically generated

### 10.8 Atur *Inspector Enemy*

1. Selanjutnya *Create Empty* pada hirarki dan beri nama “Boundary”. Pada *tab inspector* tambahkan *Box Collider* 2D kemudian atur menjadi seperti di bawah ini.

A screenshot of a computer game

Description automatically generated

### 10.9 Atur *Boundary*

1. Setelah itu buat *script* baru dengan nama “Enemy\_Behavior” dan tambahkan *source code* di bawah 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. *Drag and drop script* “Enemy\_Behavior” ke *enemy slime*. Kemudian untuk membuat animasi berjalan, pada *tab animation create* animasi baru. Jika dijalankan, maka animasi *enemy* akan terlihat.

A screenshot of a video game

Description automatically generated

### 10.10 Memberi *Script* dan Animasi pada *Enemy*

1. **Membuat Enemy AI**
2. Buat *script* dengan nama “Enemy\_AI” pada *folder* Praktikum dan masukkan *source code* di bawah 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. *Drag and drop script* “Enemy\_AI” pada *enemy* yang terbang, lalu atur jarak pandang dan kecepatan musuh.

A screenshot of a computer

Description automatically generated

### 10.11 Atur *Enemy AI*

1. Kemudian untuk menambahkan animasi, klik *enemy* kemudian pada *tab animation* buat animasi baru. *Drag and drop asset* animasi ke dalam animasi baru yang sudah dibuat.

A screenshot of a video game

Description automatically generated

### 10.12 Memberi *Script* dan Animasi pada *Enemy*

1. **Membuat Respawn**
2. Kemudian buka *script Player* dan tambahkan *source code* di bawah ini.

|  |
| --- |
| #Pada class Player  public int nyawa;  [SerializeField] Vector3 respawn\_loc;  public bool play\_again;  #Pada void Awake  respawn\_loc = transform.position;  #Tambahkan void playagain  void playagain()      {          if (play\_again == true)          {              nyawa = 3;              transform.position = respawn\_loc;              play\_again = false;          }      }  #Pada void Update  if (nyawa < 0)          {              play\_again();          }          if (transform.postition.y < -10)          {              play\_again = true;              play\_again();          } |

1. Selanjutnya buat *script* baru dan beri nama “Enemy\_Attacked” dan masukkan *source code* di bawah 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. *Drag and drop script* “Enemy\_Attacked” ke masing-masing *enemy*, kemudian pada *inspector* bagian *Object* arahkan ke Astronaut-idle (Player).

A screenshot of a computer

Description automatically generated

### 10.13 Atur *Inspector* pada Setiap *Enemy*

1. Klik objek *Player*, kemudian pada *tab inspector* ubah nilai Nyawa menjadi 3.

A screenshot of a computer

Description automatically generated

### 10.14 Atur Nyawa pada *Player*

1. Tekan *play*, kemudian jika *Player* mengenai atau menyentuh *enemy* sebanyak 3 kali, maka nyawa akan berkurang dan jika nyawa *Player* < 0 secara otomatis *Player* akan *respawn* ke titik awal.

A screenshot of a video game

Description automatically generated

### 10.15 Jalankan *Game*

1. **Kuis**

|  |
| --- |
| using UnityEngine;  public class PlayerAttack : MonoBehaviour  {      public int atackRange = 2.0f;      public int attacDamage = 10;      void Update()      {          if (InputGetButtonDown("Fire1"))          {              PerformMeleeAttack();          }      }      void PerformMeleeAttack()      {          RaycastHit hit;          if (Physics.Raycast(transform.position, transform.forward, out hit, attackRange))          {              // Lengkapi kode di sini untuk mengenai musuh dan mengurangi health mereka          }      }  } |

Penjelasan:

Terdapat beberapa kesalahan seperti kesalahan penulisan yaitu pada ”atackRange” yang seharusnya “attackRange” dan “attacDamage” yang seharusnya “attackDamage”. Kesalahan selanjutnya ada pada void Update(), pada bagian “InputGetButtonDown” yang seharusnya “Input.GetButtonDown”. Kesalahan terakhir adalah kurangnya logika pada void PerformMeleeAttack() untuk mengurangi nyawa lawan.