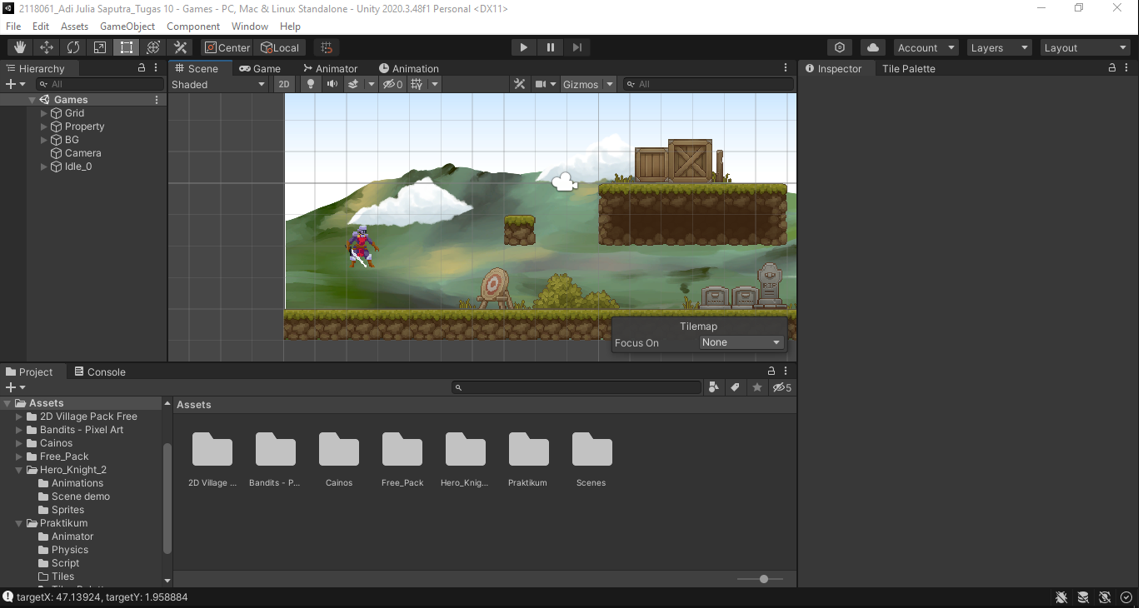
# 10 GAME ANIMATION

|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118061 |
| **Nama** | : | Adi Julia Saputra |
| **Kelas** | : | Informatika B |
| **Asisten Lab** | : | Bagas Anardi (2118004) |

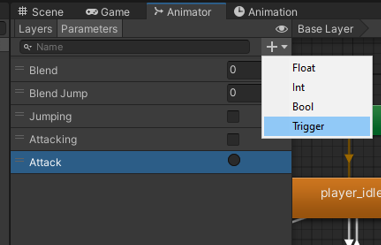
## Tugas 10 : Game Animation

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



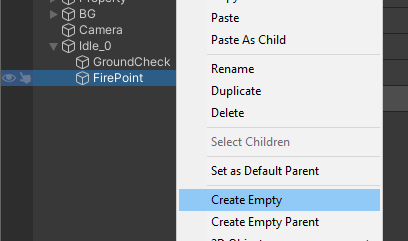
### 10.1 Membuka *Project*

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



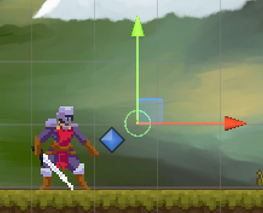
### 10.2 *Attack*

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



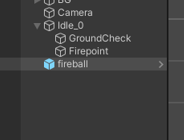
### 10.3 *Firepoint*

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



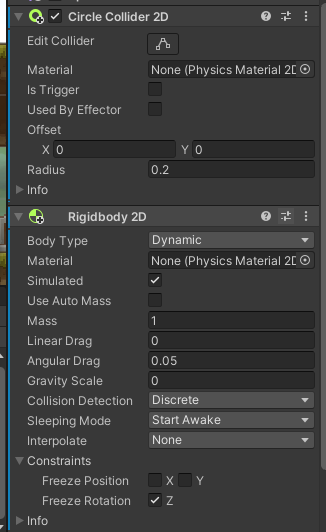
### 10.4 *Firepoint*

1. Pada menu *Hierarchy* Tambahkan *fireball* yang diambil dari *assets.*



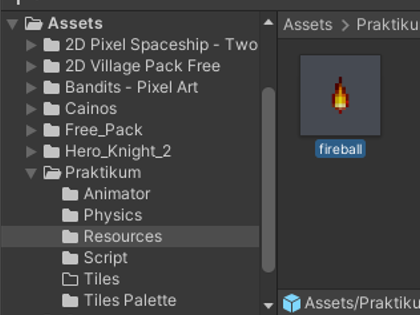
### 10.5 *Fireball*

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



### 10.6 *Add Component*

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

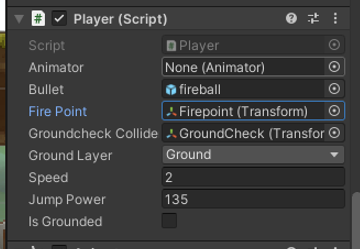


### 10.7 *Fireball Resources*

1. Pada *Script Player* Tambahkan *Script* dibawah ini

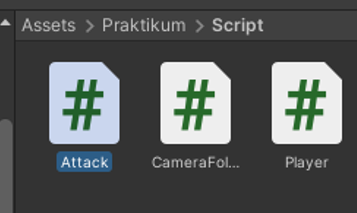
|  |
| --- |
| #pada class public Animator animator;      public GameObject bullet;      public Transform firePoint;  #dibawah fungsi fixedupdate  IEnumerator Attack()      {          animator.SetTrigger("Attack");          yield return new WaitForSeconds(0.25f);          float direction = facingRight ? 1f : -1f; // Determine direction based on facingRight          GameObject fireball = Instantiate(bullet, firePoint.position, Quaternion.identity);          fireball.GetComponent<Rigidbody2D>().velocity = new Vector2(direction \* 10f, 0);          Destroy(fireball, 2f);          attacking = true;          attackArea.SetActive(attacking);          animator.SetBool("Attacking", true);          Debug.Log("Attack initiated");      }  #didalam update  if (Input.GetKeyDown(KeyCode.C))          {              Debug.Log("C key pressed - starting attack");              StartCoroutine(Attack());          } |

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



### 10.8 *Inspector Player*

1. Buat *Script* *Attack* pada *folder Script*

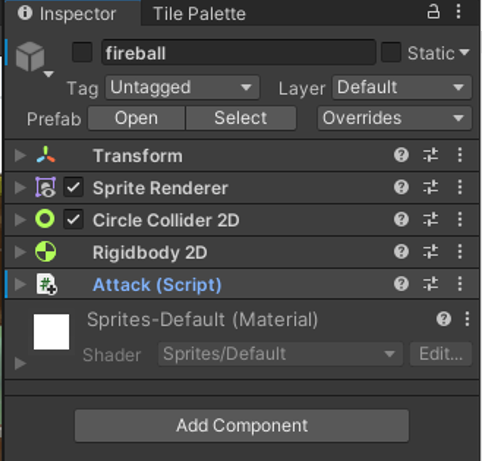


### 10.9 *Script Player*

1. Tambahkan *Script Attack* dibawah ini

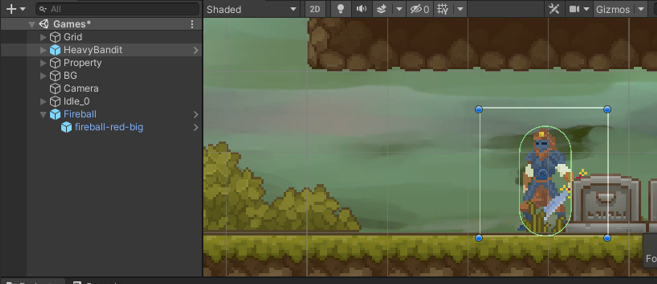
|  |
| --- |
| using System.Collections;  using System.Collections.Generic;  using UnityEngine;  public class Attack : MonoBehaviour  {      private void OnTriggerEnter2D(Collider2D collision)      {          if (collision.gameObject.CompareTag("Enemy"))          {              Destroy(gameObject); 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



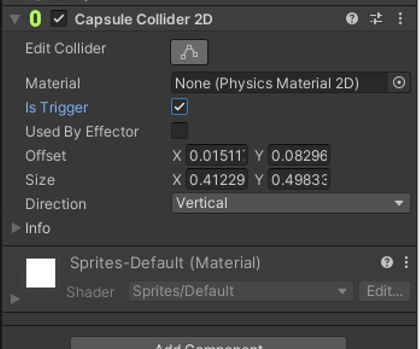
### 10.10 *Script Attack*

1. Tambahkan *Enemy* *HeavyBandit* pada *hierarchy* di *folder Sprites*/



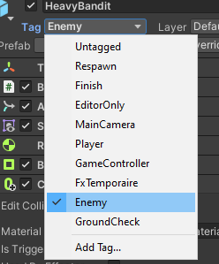
### 10.11 *Heavy Bandit*

1. Kemudian klik pada *HeavyBandit*, lalu pada *menu* *tab* *inspector* tambahkan c*apsule* *collider* 2D untuk mendeteksinya



### 10.12 *Capsule Collider 2d*

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



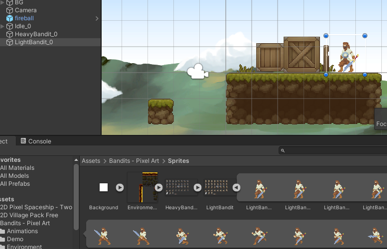
### 10.13 *Tag Enemy*

1. Tembak *Enemy* dengan menekan Tombol C untuk menghancurkan musuh



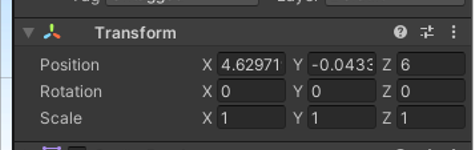
### 10.14 Tembak *Enemy*

1. **Membuat Enemy Behavior NPC**
2. Tambahkan *sprite* baru bernama *LightBandit* ke dalam *hierarchy*.



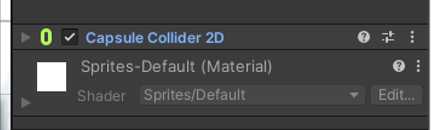
### 10.15 *LightBandit*

1. Pada *inspector* atur *transform scale position* z rubah menjadi 6.



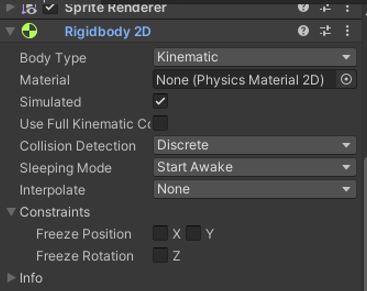
### 10.16 *Transform Scale*

1. Tambahkan sebuah komponen bernama *Capsule Colider* 2D dan *Rigidbody* dalam *inspector* *game* objek *LightBandit*.



### 10.17 *Capsule Collider 2D*

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



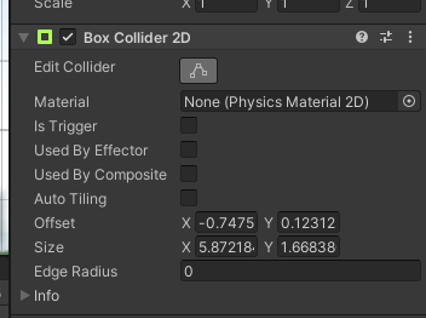
### 10.18 *Rigidbody 2D*

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



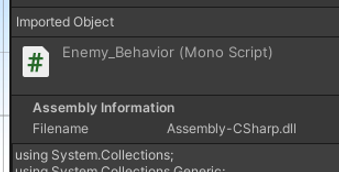
### 10.19 *Boundary*

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



### 10.20 *Box Collider 2D*

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

### 10.21 *Script Enemy Behavior*

1. Tambahkan Script dibawah ini di dalam script Enemy\_Behavior

|  |
| --- |
| 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. Jalankan Program



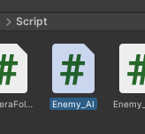
### 10.22 *Enemy Behavior*

1. **Membuat Enemy Enemy AI**
2. Cari sebuah sprite pack bernama *'HeavyBandit'* dan buka *folder* bernama *'HeavyBandit'*. Tambahkan ‘*HeavyBandit’* pada *Hierarchy*.



### 10.23 *HeavyBandit*

1. Buat *Script* *Enemy\_AI* pada *folder* Praktikum – *Script.*

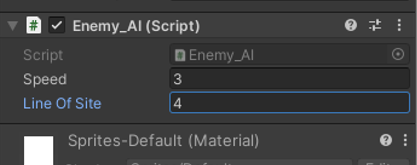


### 10.24 *Script Enemy\_AI*

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      private Animator animator; // Animator untuk musuh      private bool facingRight = true; // Apakah musuh menghadap kanan      // Use this for initialization      void Start()      {          // Mencari pemain berdasarkan tag          player = GameObject.FindGameObjectWithTag("Player").transform;          // Menyimpan posisi awal musuh          initialPosition = GetComponent<Transform>().position;          // Mendapatkan komponen Animator          animator = GetComponent<Animator>();      }      // 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              Vector2 targetPosition = new Vector2(player.position.x, transform.position.y);              transform.position = Vector2.MoveTowards(transform.position, targetPosition, speed \* Time.deltaTime);              // Set parameter animator untuk berlari              animator.SetBool("isRunning", true);              // Balik arah musuh sesuai arah gerakan              if ((player.position.x > transform.position.x && facingRight) || (player.position.x < transform.position.x && !facingRight))              {                  Flip();              }          }          else          {              // Musuh kembali ke posisi awal              Vector2 targetPosition = new Vector2(initialPosition.x, transform.position.y);              transform.position = Vector2.MoveTowards(transform.position, targetPosition, speed \* Time.deltaTime);              // Set parameter animator untuk berlari              animator.SetBool("isRunning", true);              // Balik arah musuh sesuai arah gerakan              if ((initialPosition.x > transform.position.x && facingRight) || (initialPosition.x < transform.position.x && !facingRight))              {                  Flip();              }              // Jika musuh sudah kembali ke posisi awal, set animasi idle              if (Mathf.Approximately(transform.position.x, initialPosition.x))              {                  animator.SetBool("isRunning", false);              }          }      }      // Fungsi untuk membalik arah musuh      void Flip()      {          facingRight = !facingRight;          Vector3 theScale = transform.localScale;          theScale.x \*= -1;          transform.localScale = theScale;      }      // Untuk menggambar jarak penglihatan musuh di editor      private void OnDrawGizmosSelected()      {          Gizmos.color = Color.red;          Gizmos.DrawWireSphere(transform.position, lineOfSite);      }  } |

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



### 10.25 *Script Enemy AI*

1. *Running* *Game*, maka *enemy* akan mengikuti Gerakan *Player.*



### 10.26 *Enemy AI*

1. **Respawn**
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 di dalam *void 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();          } |

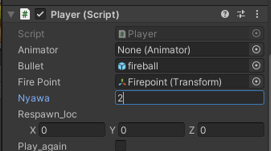
1. Tambahkan fungsi *playagain*() dalam *script Player.cs*.

|  |
| --- |
| 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* *enemy* Tambahkan *Script enemy attack*, arahkan object pada *player-idle-1*. Klik game *object* *Player*, pergi ke *Inspector* dan ubah nilai Nyawa menjadi 3 pada *Player*(*Script*).

****

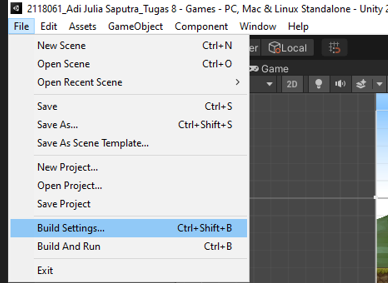
### 10.27 *Script*

1. Jika di *play*, *Player* mengenai atau menyentuh *enemy* sebanyak 2 kali maka nyawa akan berkurang 1 dan jika nyawa kurang dari 0 maka akan *respawn* ke titik awal.

****

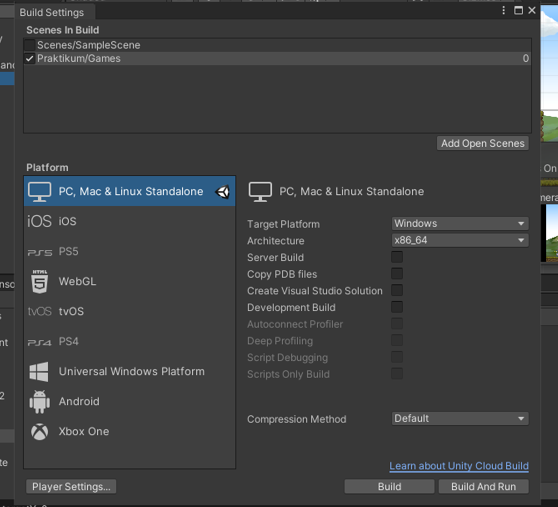
### 10.28 *Respawn*

1. **Render**
2. Pergi ke *menu* *File* kemudian pilih *Build Setting* (Ctrl + Shift + B).



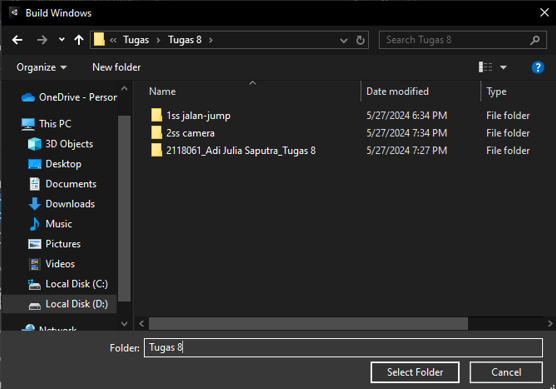
### 10.29 *Build Setting*

1. Pada *Setting Build* ini pilih PC, Mac & Linux, Tekan *Build*, pastikan pada *menu* *Scene in Build* berada pada *project* Tugas Kalian.



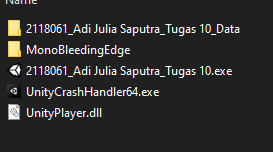
### 10.30 *Build Setting*

1. Pilih dimana *Project* disimpan, dan tunggu hasilnya.



### 10.31 Lokasi Simpan File

1. Hasil Render.



### 10.32 Hasil Render

1. **Repository GitHub**

[2110061\_PRAK\_ANIGAME](https://github.com/AdiJulia/2118061_PRAK_ANIGAME)

1. **Kuis Pertemuan 10**

Lengkapi Source code dibawah ini :

|  |
| --- |
| 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  }  }  } |

Source code yang sudah dilengkapi yang kurang

|  |
| --- |
| using UnityEngine;  public class PlayerAttack : MonoBehaviour  {  public float attackRange = 2.0f;  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 enemyHealth = hit.transform.GetComponent<EnemyHealth>();  if (enemyHealth != null)  {  enemyHealth.TakeDamage(attackDamage);  }  }  }  } |

Berikan Tanda Merah yang menyebabkan Source code Error

|  |
| --- |
| void HandleJumpInput()  {  if (Input.GetKeyDown(KeyCode.Space))  {  animator.SetBool("isJumping", );  rb.AddForce(Vector2.up \* jumpForce, ForceMode2D.Impulse);  }  else if (Input.GetKey(KeyCode.Space))  {  animator.SetBool("isJumping",);  }  }  void HandleMovementInput()  {  float move = Input.GetAxis("Horizontal");  if (move != 1)  {  animator.SetBool("isIdle", true);  transform.Translate(Vector3.left \* move \* Time.deltaTime);  }  else  {  animator.SetBool("isWalking", false);  }  if (move != 0)  {  transform.localScale = new Vector3(-4, 1, 1);  }  else if (move > 0)  {  transform.localScale = new Vector3(1, 2, 1);  }  } |

Dan ini merupakan source code yang sudah diperbaiki

|  |
| --- |
| void HandleJumpInput()  {  if (Input.GetKeyDown(KeyCode.Space))  {  animator.SetBool("isJumping", true); // Set to true when jumping starts  rb.AddForce(Vector2.up \* jumpForce, ForceMode2D.Impulse);  }  else if (Input.GetKey(KeyCode.Space))  {  animator.SetBool("isJumping", true); // Set to true while the space key is held down  }  else  {  animator.SetBool("isJumping", false); // Set to false when the space key is released  }  }  void HandleMovementInput()  {  float move = Input.GetAxis("Horizontal");  if (move == 0)  {  animator.SetBool("isIdle", true);  animator.SetBool("isWalking", false);  }  else  {  animator.SetBool("isIdle", false);  animator.SetBool("isWalking", true);    transform.Translate(Vector3.right \* move \* Time.deltaTime); // Corrected to Vector3.right  }  if (move < 0)  {  transform.localScale = new Vector3(-1, 1, 1); // Correct scaling for left direction  }  else if (move > 0)  {  transform.localScale = new Vector3(1, 1, 1); // Correct scaling for right direction  }  } |

Analisa :

Pada bagian pertama, fungsi *PerformMeleeAttack* yang ada pada *PlayerAttack* script mengalami beberapa kekurangan. Kesalahan utamanya adalah tidak adanya kode untuk mengurangi *health* musuh setelah terkena serangan. Hal ini disebabkan oleh tidak adanya pemeriksaan komponen musuh (*EnemyHealth*) yang mengelola health mereka. Selain itu, dalam *script* *HandleJumpInput*, kesalahan utama adalah tidak adanya nilai *boolean* pada *animator*.*SetBool*("*isJumping*", );. Kondisi ini menyebabkan *error* kompilasi karena *SetBool* membutuhkan dua parameter: nama parameter animasi dan nilai *boolean* yang akan diatur. Pada *script* *HandleMovementInput*, terdapat beberapa kesalahan *logika*. Pertama, kondisi *if (move != 1)* yang seharusnya memeriksa apakah nilai *move* tidak sama dengan nol (0) untuk mendeteksi gerakan. Kedua, arah *transform.Translate(Vector3.left \* move \* Time.deltaTime);* yang tidak sesuai dengan arah gerakan yang benar.

Setelah perbaikan, pada fungsi *PerformMeleeAttack*, telah ditambahkan logika untuk memeriksa apakah objek yang terkena adalah musuh dan mengurangi *health* mereka menggunakan komponen *EnemyHealth*. *Script HandleJumpInput* telah diperbaiki dengan menambahkan nilai *boolean* yang sesuai (*true atau false*) pada *animator*.*SetBool*, sehingga sekarang dapat berfungsi dengan benar. Pada *script* *HandleMovementInput*, *kondisi if (move != 1)* telah diperbaiki menjadi *if (move == 0)* untuk mendeteksi jika tidak ada pergerakan. Selain itu, arah gerakan telah diperbaiki dengan menggunakan *Vector3.right* untuk mencerminkan arah yang benar berdasarkan nilai *move*. Logika animasi juga telah diperbaiki dengan mengatur *animator.SetBool("isWalking", true)* ketika ada gerakan dan *animator.SetBool("isIdle", false)* ketika tidak ada. Terakhir, pengaturan *transform.localScale* telah disesuaikan untuk mencerminkan arah karakter yang menghadap kiri atau kanan dengan benar, berdasarkan nilai *move*.