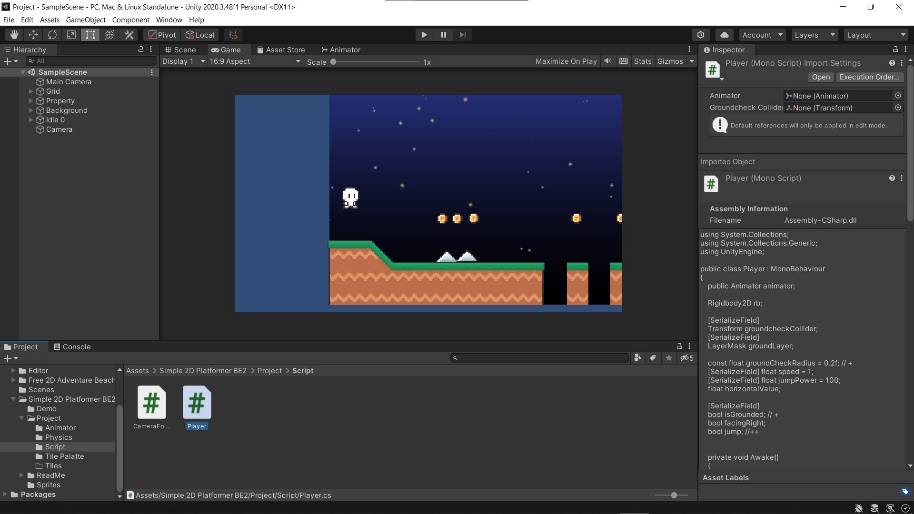
# 10 RESPAWN AND AI ENEMY ATTACK

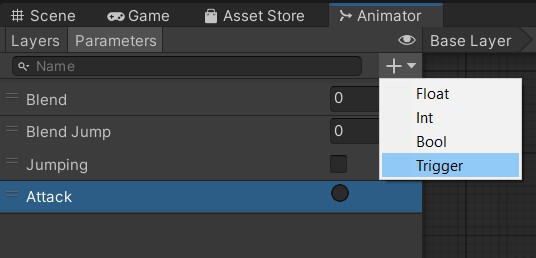
|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118012 |
| **Nama** | : | Adrianus Vianto Eban Kia |
| **Kelas** | : | A |
| **Asisten Lab** | : | Naufal Dhiaurrafif (2218059) |

## Tugas 10 : Membuat AI pada Game

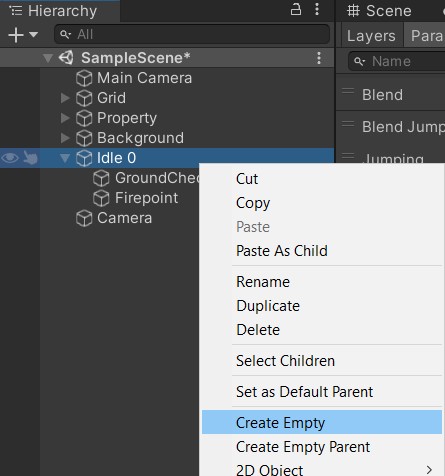
1. Membuat Mekanisme Attack
2. Buka Project Bab 9 Untuk melanjutkannya.



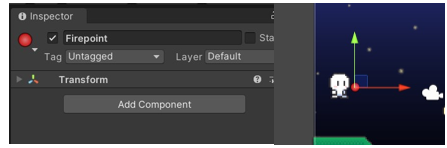
1. Kemudian pada menu Tab *Animator* Tambahkan Parameter *Trigger*, Rename Menjadi *Attack*



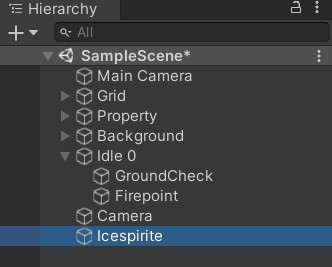
1. Setelah menambahkan parameter *Attack*, Langkah selanjutnya adalah membuat Layer *Game object* baru didalam *player-idle-1*, Klik kanan pilih *Create Empty* lalu Rename menjadi *Firepoint.*



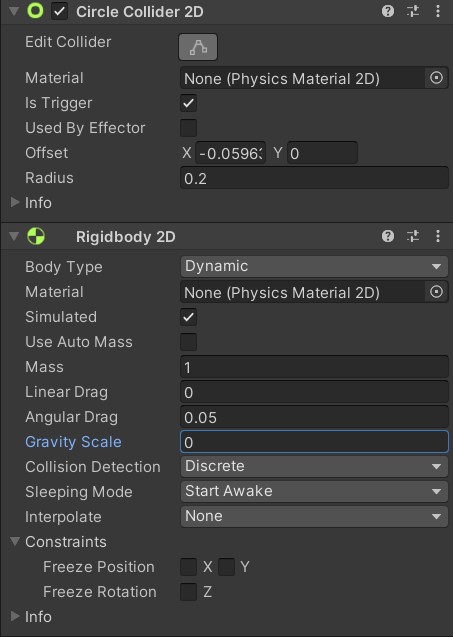
1. Pada menu *Hierarchy* klik *Firepoint* untuk setting pada *Inspector*, Ubah *Icon* Menjadi titik, atur letak titik didepan player.



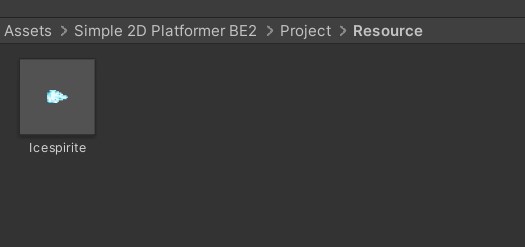
1. Pada menu Hierarchy Tambahkan item-feedback-1, di folder *Attack* > I5050-7 , *rename* menjadi *IceSpirites*



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



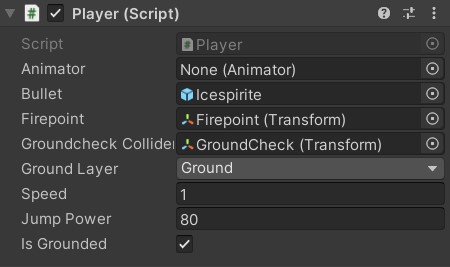
1. Buat Folder baru Resources di menu Project, kemudian drag and drop fireball kedalam folder *Resources*, dan hapus *IceSpirites* pada *Hierarchy*.



1. Pada Script Player Tambahkan Script dibawah ini.

|  |
| --- |
| public GameObject bullet;  public Transform Firepoint;  #Tambahkan dibawah 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);  }  #Tambakan pada Function Void Update  if (Input.GetKeyDown(KeyCode.C))  {      StartCoroutine(Attack());  } |

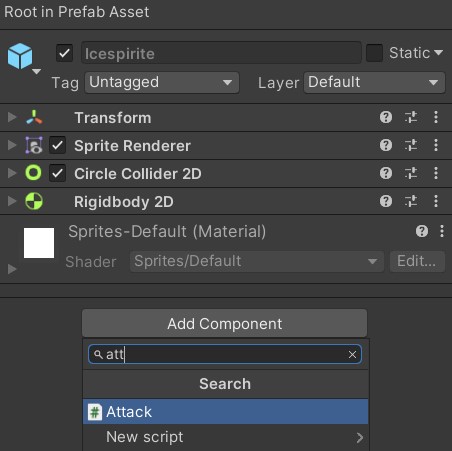
1. Pada *Inspector* *Player*, Ubah seperti dibawah ini, Dimana *Bullet* berisi object yang akan ditembak sedangkan fire point adalah titik tembak pertama.



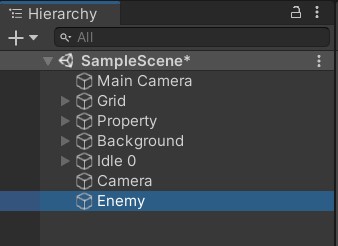
1. Buat Script Attack pada folder Script, lalu tambahkan *source code* ini

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

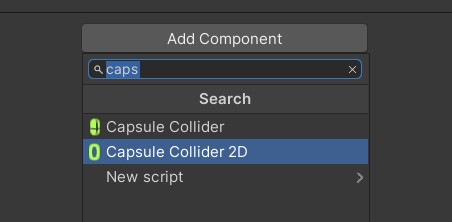
1. Didalam folder resource Tambahkan *Script Attack* di *Prefab Icespirites*, dengan cara Klik *Icespirites* kemudian pada menu *Inspector* *Add Component Script Attack*



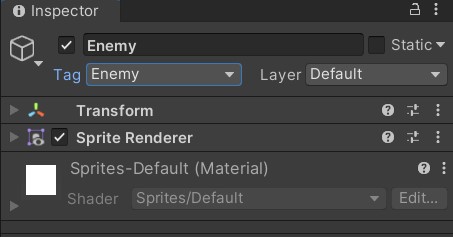
1. Tambahkan Enemy pada hierarchy di folder Sprites Enemy.



1. Kemudian klik pada Enemy, lalu pada menu tab *inspector* tambahkan *Capsule Collider 2D* untuk mendeteksinya.



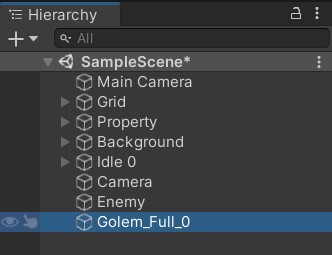
1. Tambahkan Tag Enemy dengan cara Pilih Add Tag, kemudian add tag to the list, Tuliskan Enemy



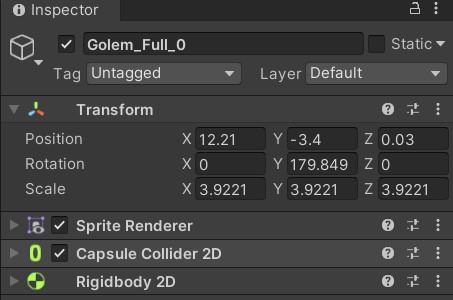
1. Tembak Enemey dengan menekan Tombol C untuk menghancurkan musuh.



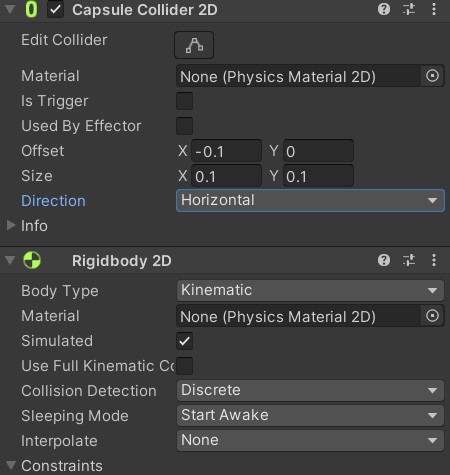
1. Membuat Enemy AI
2. Cari sebuah sprite pack Bernama *Enemy* dan buka folder bernama “Golem”, lalu tambahkan kedalam *hierarchy*.



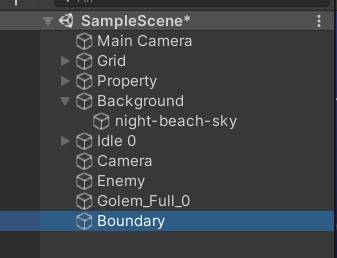
1. Tambahkan sebuah komponen bernama *Capsule Colider 2D* dan *Rigidbody 2D* dalam *inspector* game objek *Golem-Full-0*.



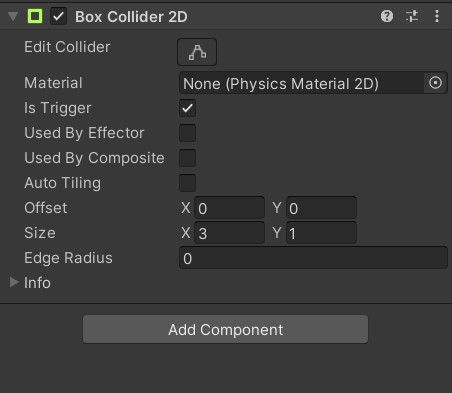
1. Atur sedikit *collider* tersebut seperti ukurannya diubah jika terlalu besar, dan pada *Body Type* Ubah menjadi *Kinematic***.**



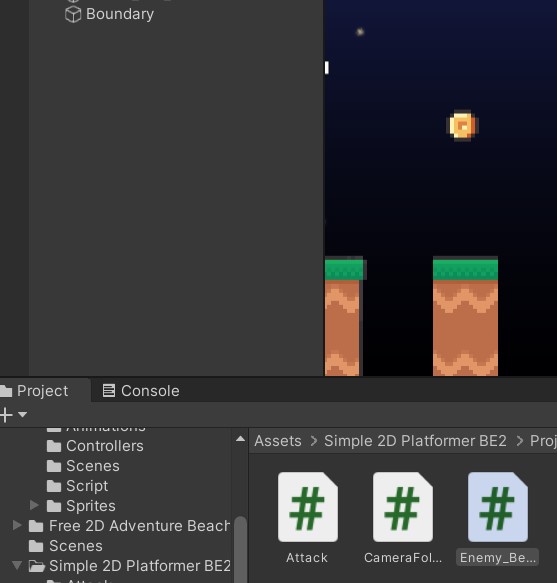
1. Create Empty object pada *Hierarchy*, Rename Menjadi *Boundary*.



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



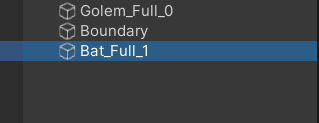
1. Buat sebuah file script didalam folder *Script* beri nama “*Enemy\_Behavior*”, kemudian drag dan masukkan ke dalam game object “*Golem-Full-0*”.



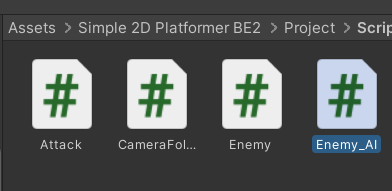
1. Tambahkan Script dibawah ini.

|  |
| --- |
| 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. Membuat Enemy AI
2. Cari sebuah sprite pack bernama 'Enemy' dan buka folder bernama 'Bat'. Tambahkan 'Bat-Full-1' pada *Hierarchy*



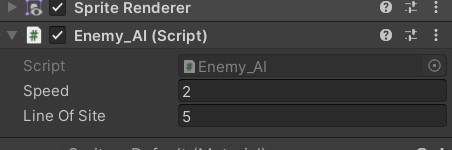
1. Buat Script Enemy\_AI pada folder Praktikum - Script.



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);      }  } |

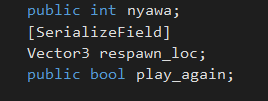
1. Pada Inspector *Enemy\_AI*, Atur *Speed* juga *Line of Site* untuk menentukan jarak dan *speed* pada *enemy*.



1. Running Game, maka Bat akan mengikuti Gerakan Player.



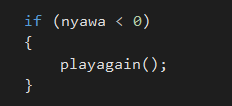
1. Membuat Respawn
2. Buka file script (Player.cs) tambahkan variabel nyawa seperti dibawah ini.



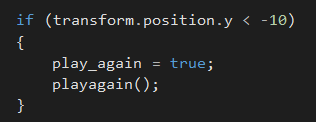
1. Tambahkan kode dibawah ini untuk mengatur posisi *respawn* sesuai dengan posisi awal permainan dimulai.



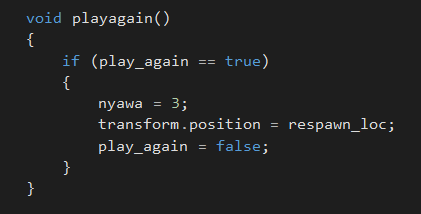
1. Tambahkan kode dibawah ini di dalam *void* *update* *Player.cs* agar ketika nyawa player dibawah 0 maka akan melakukan .



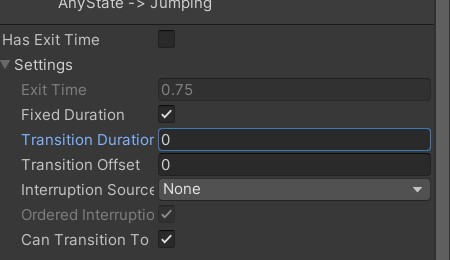
1. Tambahkan juga kode berikut dibawah code sebelumnya agar ketika player jatuh dibawah *platform* akan melakukan *respawn*.



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



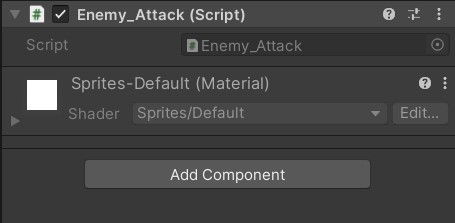
1. Klik Settings dan ubah nilai *Transition Duration* menjadi 0 dan hilangkan centang *Has Exit Time*.



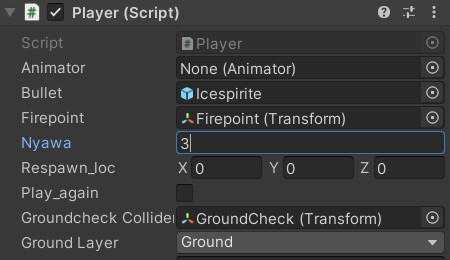
1. Buat file C++ *Enemy\_Attacked.cs* pada folder *Script* dan isikan *source code* dibawah ini.

|  |
| --- |
| 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 Bat-Full-1* Tambahkan *Script Enemy\_Attack*, arahkan object pada *idle-1*.



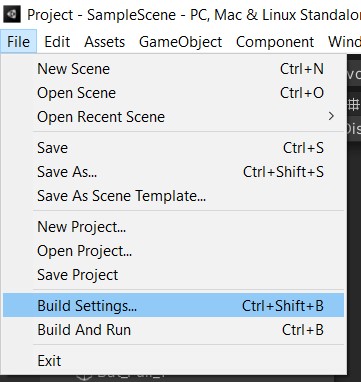
1. Klik game object *Player*, pergi ke *Inspector* dan ubah nilai Nyawa menjadi 3 pada *Script Player*.



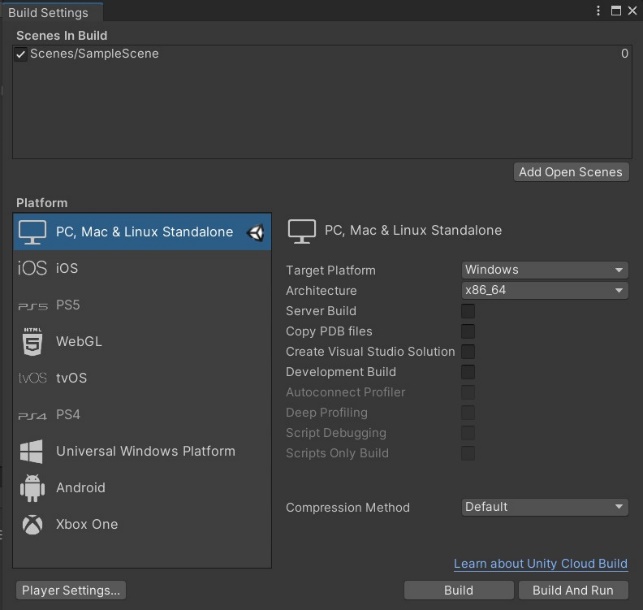
1. Jika di play, *Player* mengenai atau menyentuh *Golem-Full-0* sebanyak 3 kali maka nyawa akan berkurang 1 dan jika nyawa kurang dari 0 maka akan *respawn* ke titik awal.



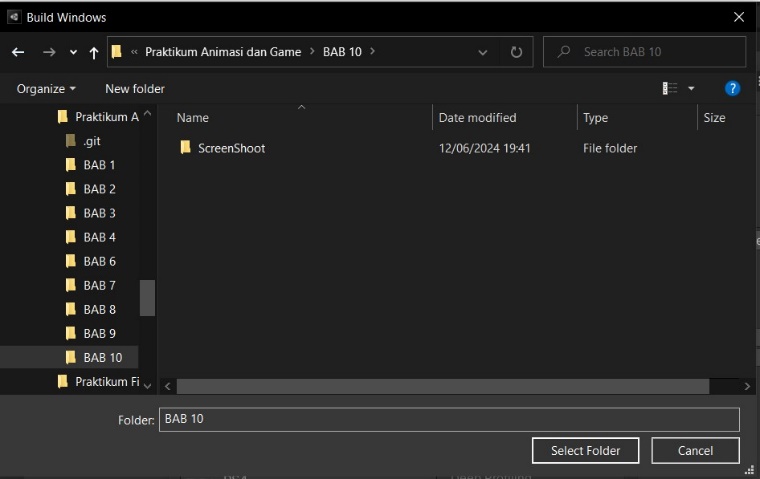
1. Render
2. Pergi ke menu File kemudian pilih *Build Setting* (Ctrl + Shift + B)*.*



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



1. Pilih dimana Project disimpan, dan tunggu hasilnya



QUIZ :

*Source Code* :

|  |
| --- |
| 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* Lengkap *:*

|  |
| --- |
| using UnityEngine;  public class PlayerAttack : MonoBehaviour  {  public float attackRange = 2.0f; // Ubah ke float untuk jarak serangan  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))  {  // Lengkapi kode di sini untuk mengenai musuh dan mengurangi health mereka  EnemyHealth enemyHealth = hit.collider.GetComponent<EnemyHealth>();  if (enemyHealth != null)  {  enemyHealth.TakeDamage(attackDamage);  }  }  }  } |

Analisa *Source Code* :

Kode ini mengimplementasikan mekanisme serangan jarak dekat untuk pemain dalam game Unity. Saat pemain menekan tombol, fungsi *PerformMeleeAttack* dipanggil, yang menggunakan *raycast* untuk mendeteksi objek dalam jarak tertentu di depan pemain. Jika *raycast* mengenai objek yang memiliki komponen *EnemyHealth*, maka fungsi *TakeDamage* pada objek tersebut dipanggil, mengurangi health-nya dengan nilai *damage* yang telah ditentukan. Jika *health* musuh mencapai nol, fungsi *Die* dipanggil untuk menghancurkan objek musuh, mensimulasikan kematiannya dalam game. Skrip *EnemyHealth* mengatur *health* musuh, mengurangi *health* saat terkena damage, dan membunuh musuh.