



TUGAS PERTEMUAN: 10

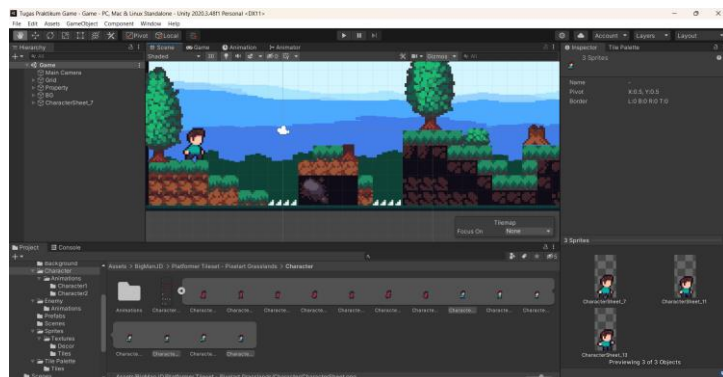
Respawn and AI Enemy Attack

NIM	:	2118099
Nama	:	Indra Syaputra Stiansyah
Kelas	:	E
Asisten Lab	:	Nur Aria Hibnastiar (2118078)
Baju Adat	:	
Referensi	:	

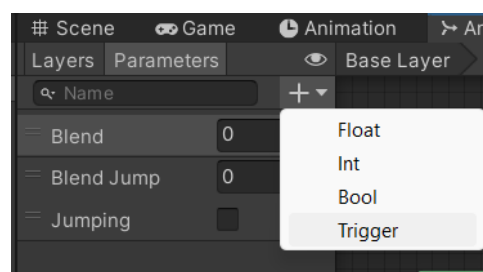
10.1 Tugas 10 : Membuat Respawn and AI Enemy Attack

A. Membuat Mekanisme Attack

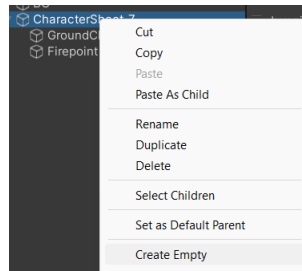
1. Buka Project Bab 9 Untuk melanjutkannya.



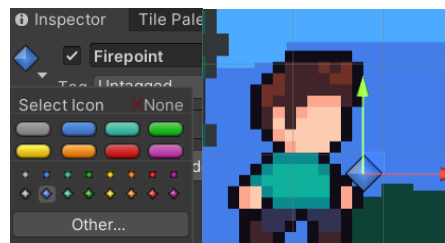
2. Kemudian pada menu Tab **Animator** Tambahkan Parameter Trigger, Rename Menjadi **Attack**



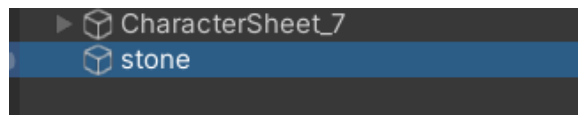
3. Setelah menambahkan parameter Attack, Langkah selanjutnya adalah membuat Layer *Game object* baru didalam *CharacterSheet_7*, Klik kanan pilih *Create Empty* lalu Rename menjadi *Firepoint*



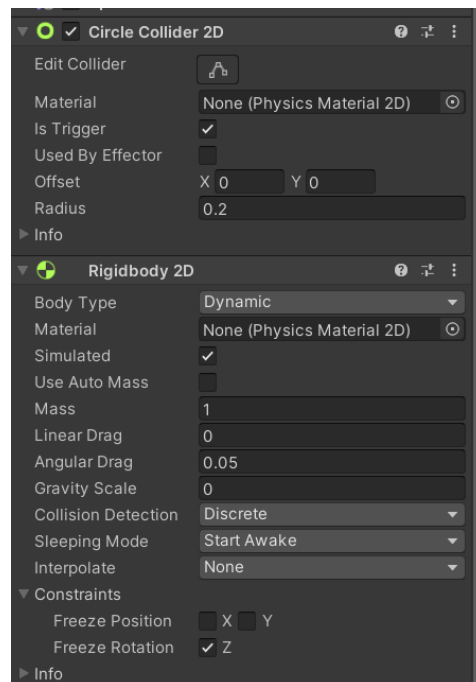
4. Pada menu Hierarchy klik Firepoint untuk setting pada Inspector, Ubah Icon Menjadi titik, atur letak titik didepan player



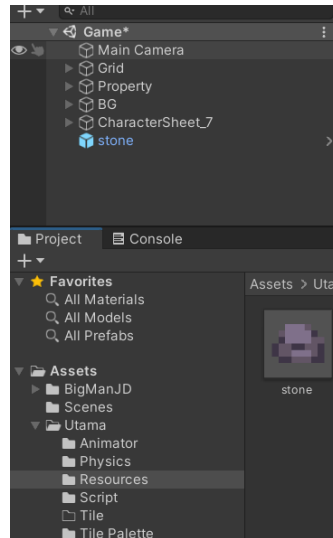
5. Pada menu Hierarchy Tambahkan Decor_3, di folder Sprites > Decor > Décor_3 , *rename* menjadi *stone*.



6. Klik stone untuk menambahkan Component Circle Collider 2d, dan Riggidbody 2D, Setting sesuai gambar dibawah ini



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



8. Pada Script Player Tambahkan Script dibawah ini

```
#Pada class Player
// Deklarasi variabel
public Animator animator;
public GameObject bullet;
public Transform firePoint;
```

```
#Tambahkan dibawah fungsi fixedUpdate
IEnumerator Attack()
{
    animator.SetTrigger("Attack");
    yield return new WaitForSeconds(0.25f);

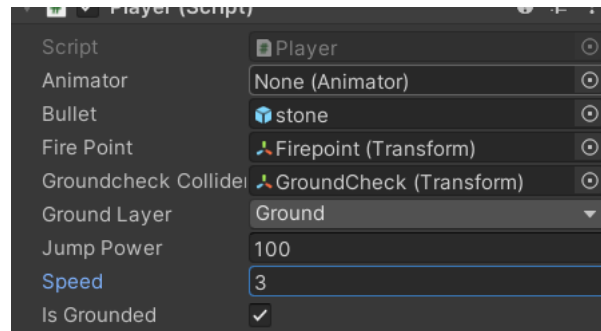
    float direction = 1f;

    GameObject fireball = Instantiate(bullet,
    firePoint.position, Quaternion.identity);

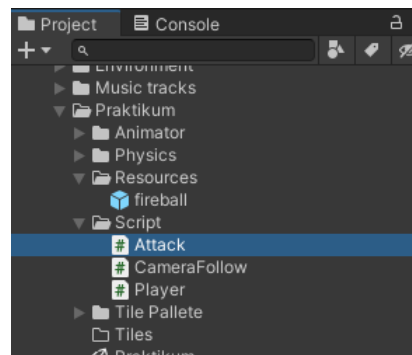
    fireball.GetComponent<Rigidbody2D>().velocity =
    new Vector2(direction * 10f, 0);

    Destroy(fireball, 2f);
}
#Tambahkan pada Function Void Update
if (Input.GetKeyDown(KeyCode.C))
{
    StartCoroutine(Attack());
}
```

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



10. Buat Script Attack pada folder Script

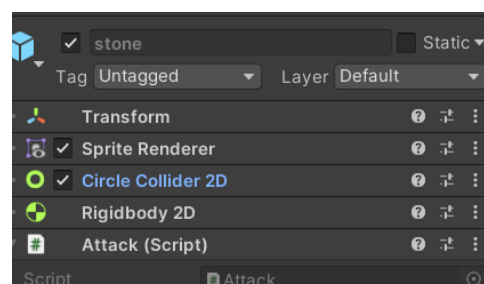


11. Tambahkan Script Attack dibawah ini

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

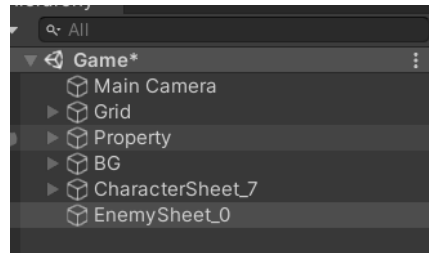
public class Attack : MonoBehaviour
{
    private void OnTriggerEnter2D(Collider2D collision)
    {
        if (collision.gameObject.CompareTag("Enemy"))
        {
            Destroy(gameObject);
            Destroy(collision.gameObject);
        }
    }
}
```

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

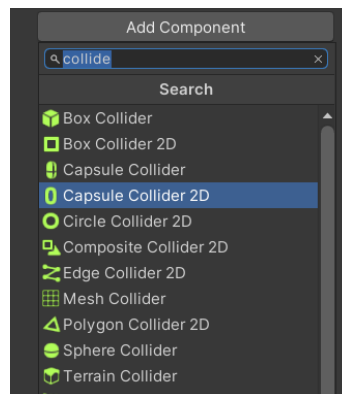




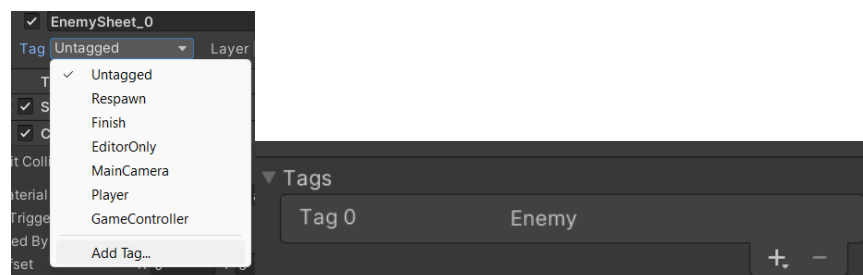
13. Tambahkan Enemy EnemySheet_0 pada hierarchy di folder Enemy



14. Kemudian klik pada EnemySheet_0, lalu pada menu tab inspector tambahkan collider 2D untuk mendeteksinya



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



16. Buka script player dan isikan source code dibawah ini agar saat menyerang serangan player dapat mengikuti arah player

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Player : MonoBehaviour
{
    public int nyawa;
    [SerializeField] Vector3 respawn_loc;
    public bool play_again;
    public Animator animator;
    public GameObject bullet;
    public Transform firePoint;
    [SerializeField] Transform groundcheckCollider;
    [SerializeField] LayerMask groundLayer;
```



```
[SerializeField] float jumpPower = 100;
const float groundCheckRadius = 0.2f;
[SerializeField] float speed = 1;
[SerializeField] bool isGrounded;
bool facingRight = true;
bool jump;
float horizontalValue;
Rigidbody2D rb;

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;
    }
}

void Update()
{
    horizontalValue
Input.GetAxisRaw("Horizontal");
    if (Input.GetButtonDown("Jump"))
    {
        animator.SetBool("Jumping", true);
        jump = true;
    }
    else if (Input.GetButtonUp("Jump"))
    {
        jump = false;
    }

    if (Input.GetKeyDown(KeyCode.C))
    {
        StartCoroutine(Attack());
    }

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

    if(transform.position.y < -10){
        play_again = true;
        playagain();
    }
}

void FixedUpdate()
{
    GroundCheck();
    Move(horizontalValue, jump);
    animator.SetFloat("Blend",
Mathf.Abs(rb.velocity.x));
}
```



```
        animator.SetFloat("Blend
rb.velocity.y);
    }

    IEnumerator Attack()
    {
        animator.SetTrigger("Attack");
        yield return new WaitForSeconds(0.25f);

        // Tentukan arah serangan berdasarkan
orientasi pemain
        float direction = facingRight ? 1f : -1f;

        GameObject fireball = Instantiate(bullet,
firePoint.position, Quaternion.identity);
        fireball.GetComponent<Rigidbody2D>().velocity
= new Vector2(direction * 10f, 0);

        Destroy(fireball, 2f);
    }

    void GroundCheck()
    {
        isGrounded = false;
        Collider2D[] colliders =
Physics2D.OverlapCircleAll(groundcheckCollider.positi
on, groundCheckRadius, groundLayer);
        if (colliders.Length > 0)
        {
            isGrounded = true;
        }

        animator.SetBool("Jumping", !isGrounded);
    }

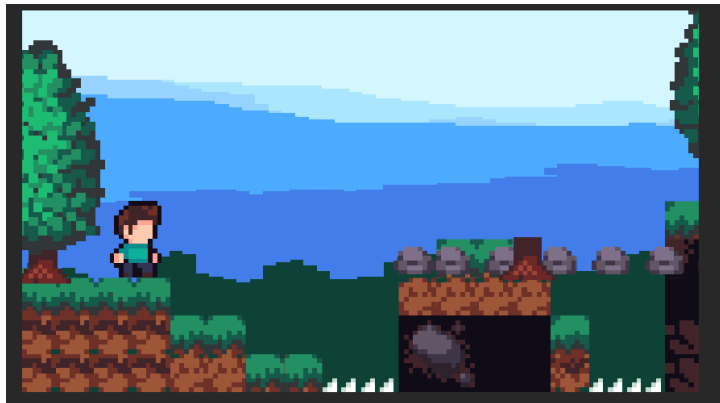
    void Move(float dir, bool jumpflag)
    {
        #region gerak kanan kiri
        float xVal = dir * speed * 100 *
Time.fixedDeltaTime;
        Vector2 targetVelocity = new Vector2(xVal,
rb.velocity.y);
        rb.velocity = targetVelocity;

        if (facingRight && dir < 0)
        {
            // ukuran player
            transform.localScale = new Vector3(-1, 1,
1);
            facingRight = false;
        }
        else if (!facingRight && dir > 0)
        {
            // ukuran player
            transform.localScale = new Vector3(1, 1,
1);
            facingRight = true;
        }
    }
    #endregion
```



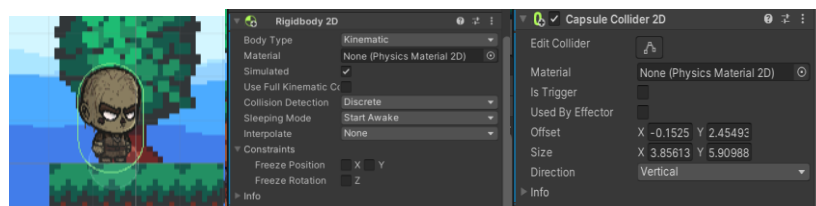
```
if (isGrounded && jumpflag)
{
    isGrounded = false;
    jumpflag = false;
    rb.AddForce(new Vector2(0f, jumpPower));
}
}
```

17. Tembak Enemy dengan menekan Tombol C untuk menghancurkan musuh

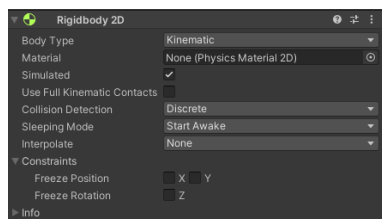


B. Membuat Enemy AI

1. Tambahkan satu enemy lagi dengan nama Zombie_Villager dan tambahkan komponen bernama Rigidbody 2D dan Capsule Collider 2D.

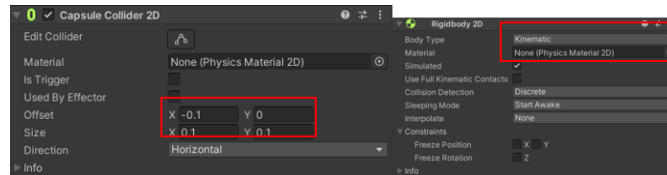


2. Klik pada enemy yang telah ditambah tadi, kemudian tambahkan sebuah komponen bernama Rigidbody 2D dalam inspector game objek EnemySheet_0





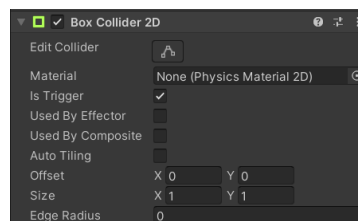
3. Atur sedikit collider pada dua enemy tersebut seperti ukurannya diubah jika terlalu besar, dan pada *Body Type* Ubah menjadi *Kinematic*



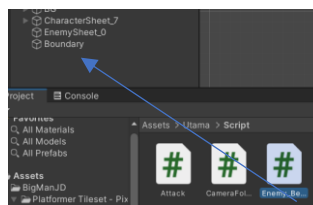
4. Create Empty object pada Hierarchy, Rename Menjadi *Boundary* lalu duplicate dan rubah nama menjadi *Boundary1*



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



6. Buat sebuah file script didalam folder Script beri nama “Enemy_Behavior”, kemudian drag dan masukkan ke dalam game object “EnemySheet_0” dan “Zombie_Villager”



7. 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>();
    }
}
```

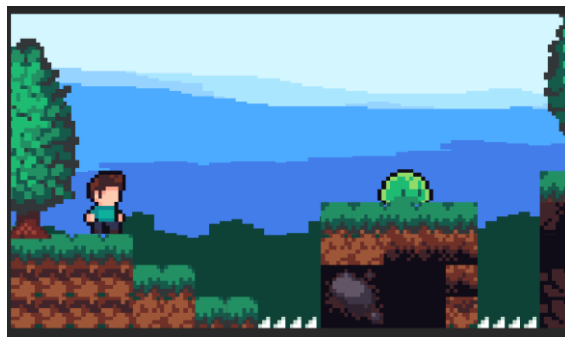


```
void Update()
{
    if (isFacingRight())
    {
        rb.velocity = new Vector2(moveSpeed, 0f);
    }
    else
    {
        rb.velocity = new Vector2(-moveSpeed, 0f);
    }
}

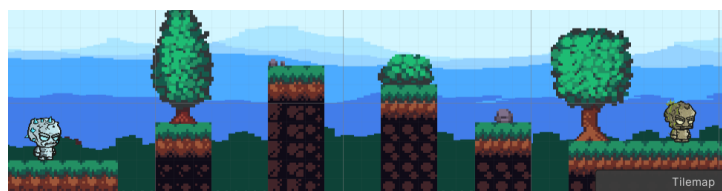
private bool isFacingRight()
{
    return transform.localScale.x > Mathf.Epsilon;
}

private void OnTriggerExit2D(Collider2D collision)
{
    transform.localScale = new Vector2(-transform.localScale.x, transform.localScale.y);
}
}
```

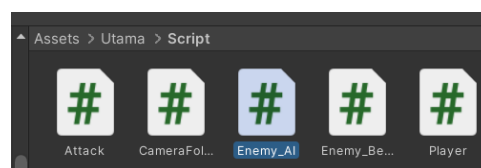
8. Jalankan Program



9. Cari sebuah folder Bernama enemy dan Tambahkan 'Golem' dan 'Golem1' pada Hierarchy



10. Buat Script Enemy_AI pada folder umum - Script



11. 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);
        }
    }

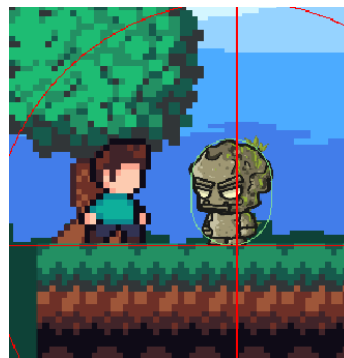
    // Untuk menggambar jarak penglihatan musuh di
    editor
    private void OnDrawGizmosSelected()
    {
        Gizmos.color = Color.red;
        Gizmos.DrawWireSphere(transform.position,
        lineOfSite);
    }
}
```



12. Pada Inspector Enemy_Ai, Atur Speed juga Line of Site untuk menentukan jarak dan speed pada enemy



13. Running Game, maka golem akan mengikuti Gerakan Player



C. Respawn

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

```
public int nyawa;  
[SerializeField] Vector3 respawn_loc;  
public bool play_again;
```

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

```
private void Awake()  
{  
    rb = GetComponent<Rigidbody2D>();  
    animator = GetComponent<Animator>();  
    respawn_loc = transform.position;  
}
```

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



```
void Update()
{
    horizontalValue = Input.GetAxisRaw("Horizontal");
    if (Input.GetButtonDown("Jump"))
    { ...
    }
    else if (Input.GetButtonUp("Jump"))
    { ...
    }
    |
    if (Input.GetKeyDown(KeyCode.C))
    { ...
    }

    if(nyawa < 0){
        playagain();
    }
}
```

4. Tambahkan juga kode berikut di method update code sebelumnya agar ketika player jatuh dibawah platform akan melakukan respawn

```
if(transform.position.y < -10){
    play_again = true;
    playagain();
}
```

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

```
void playagain(){
    if(play_again == true){
        nyawa = 3;
        transform.position = respawn_loc;
        play_again = false;
    }
}
```

6. 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>
            ();
        }
    }

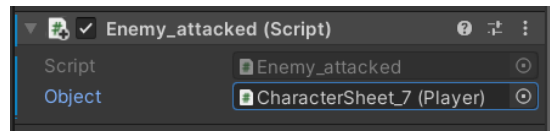
    void OnTriggerEnter2D(Collider2D other)
    {
        if (other.CompareTag("Player"))
        {

```

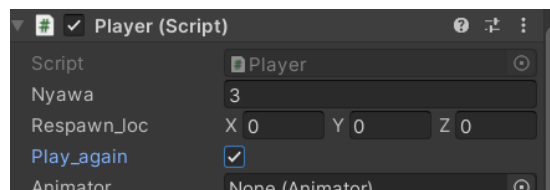


```
Object.nyawa--;\n\nif (Object.nyawa < 0)\n{\n    Object.play_again = true;\n}\n\n}\n\n}
```

7. Pada hierarchy golem dan golem1Tambahkan Script enemy attack, arahkan object pada CharacterSheet_7



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



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



D. Quiz

```
using UnityEngine;\n\npublic class PlayerAttack : MonoBehaviour\n{\n    public float attackRange = 2.0f; // Perbaiki tipe data dari int ke float\n    public int attackDamage = 10; // Perbaiki nama variabel dari attacDamage ke attackDamage\n}
```



```
void Update()
{
    if (Input.GetButtonDown("Fire1")) // Perbaiki
    penulisan fungsi Input.GetButtonDown
    {
        PerformMeleeAttack();
    }
}

void PerformMeleeAttack()
{
    RaycastHit hit;
    if (Physics.Raycast(transform.position,
    transform.forward, out hit, attackRange))
    {
        // Cek apakah objek yang terkena adalah musuh
        EnemyHealth enemyHealth =
        hit.collider.GetComponent<EnemyHealth>();
        if (enemyHealth != null)
        {
            // Kurangi health musuh
            enemyHealth.TakeDamage(attackDamage);
        }
    }
}
```

Penjelasan :

Pada skrip PlayerAttack yang telah diperbarui, beberapa modifikasi dan tambahan dilakukan untuk memungkinkan serangan pemain mengenai musuh dan mengurangi health mereka. Pertama, tipe data attackRange diubah dari int ke float agar lebih sesuai untuk merepresentasikan jarak. Kemudian, kesalahan penamaan variabel attacDamage diperbaiki menjadi attackDamage.

Di dalam metode Update, penulisan fungsi Input.GetButtonDown juga diperbaiki agar benar. Metode PerformMeleeAttack menggunakan Physics.Raycast untuk mendeteksi objek di depan pemain dalam jarak yang ditentukan oleh attackRange. Jika Raycast mengenai sesuatu, skrip memeriksa apakah objek yang terkena memiliki komponen EnemyHealth. Jika ada, maka metode TakeDamage dari EnemyHealth dipanggil dengan parameter attackDamage, yang mengurangi health musuh.

E. Link Github Pengumpulan

https://github.com/IndraSyaputraS/2118099_PRAK_ANIGAME