



## TUGAS PERTEMUAN: 8

### Playeracter and Camera Movement

NIM	:	2118104
Nama	:	Andhika Wira Sakti
Kelas	:	C
Asisten Lab	:	Rifal Rifqi Rhomadon ( 2218106 )

#### 8.1 Tugas 8: Menggerakkan Karakter dan Kamera

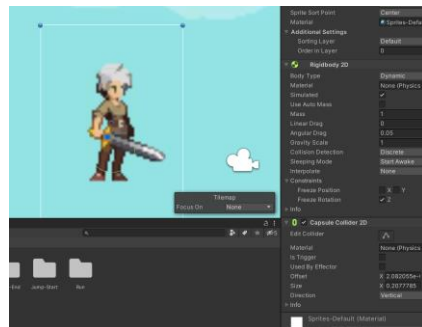
##### A. Menggerakkan Karakter

1. Buka proyek sebelumnya



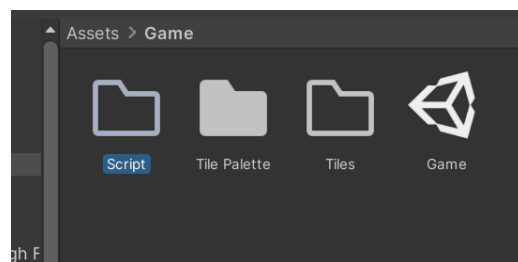
Gambar 6.1 Proyek Sebelumnya

2. Pastikan komponen Player hanya *Rigidbody 2D* dan *Capsule Collider 2D*



Gambar 6.2 Komponen Karakter

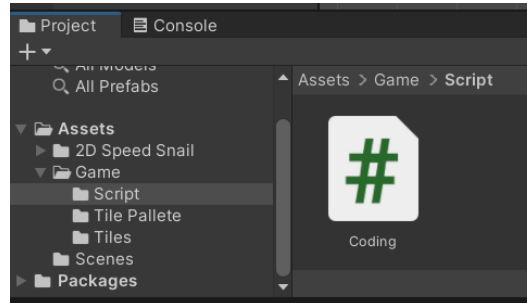
3. Buat folder 'Script' di dalam *folder* Game





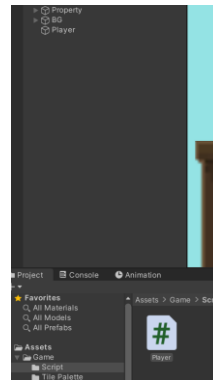
Gambar 6.3 Membuat Folder 'Script'

4. Buat berkas C# di dalam folder Script



Gambar 6.4 Membuat Berkas C# Coding

5. Sisipkan berkas Player ke Player dengan *drag drop*



Gambar 6.5 Menyisipkan Berkas Coding ke Player

6. Buka berkas Coding dan sisipkan skrip di bawah ini

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

public class Player : MonoBehaviour
{
    Rigidbody2D rb;

    [SerializeField] Transform groundcheckCollider;
    [SerializeField] LayerMask groundLayer;

    const float groundCheckRadius = 0.2f; // +
    [SerializeField] float speed = 1;
    [SerializeField] float jumpPower = 100;
    float horizontalValue;

    [SerializeField] bool isGrounded; // +
    bool facingRight;
    bool jump;
```



```
private void Awake()
{
    rb = GetComponent<Rigidbody2D>();
}

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

void FixedUpdate()
{
    GroundCheck();
    Move(horizontalValue, jump);
}

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

void Move(float dir, bool jumpflag)
{
    if(isGrounded && jumpflag)
    {
        isGrounded = false;
        jumpflag = false;
        rb.AddForce(new Vector2(0f, jumpPower));
    }

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

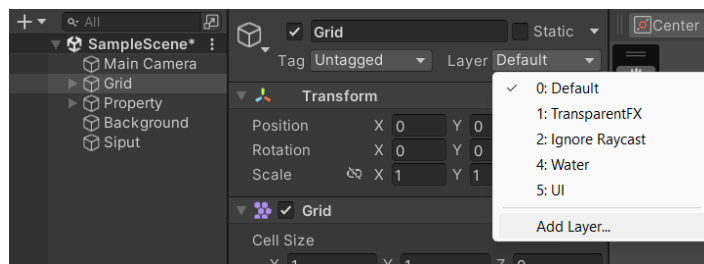
    if (facingRight && dir < 0)
    {
        // ukuran player
        transform.localScale = new Vector3(-5, 5, 5);
        facingRight = false;
    }
}
```



```
else if (!facingRight && dir > 0)
{
    // ukuran player
    transform.localScale = new Vector3(5, 5, 5);
    facingRight = true;
}

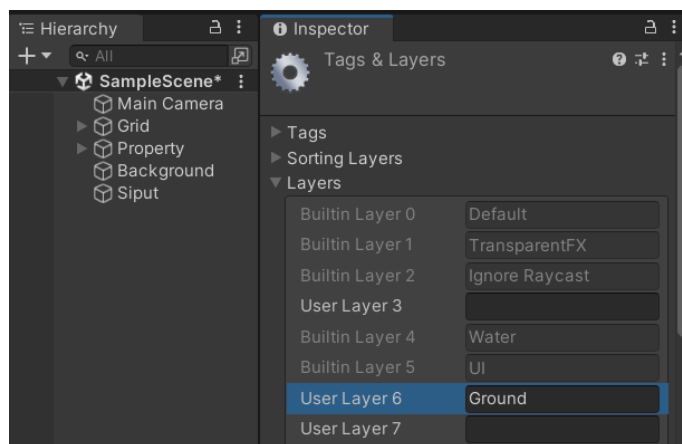
#endregion
}
```

7. Buka kembali Grid dan tambahkan *layer* baru



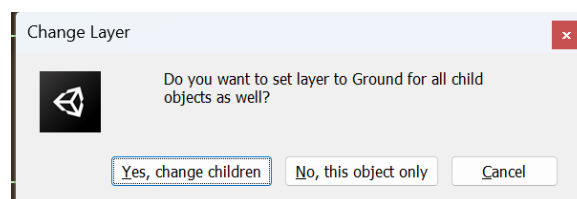
Gambar 6.6 Membuat *Layer* Baru

8. Beri nama 'Ground' pada *layer* ke-6



Gambar 6.7 Memberi Nama Baru *Layer*

9. Kembali ke Grid dan ubah *layer* menjadi 'Ground'. Ketika muncul notifikasi seperti ini, tekan 'Yes'





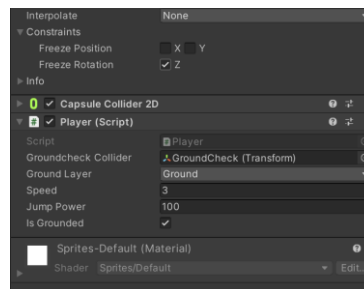
Gambar 6.8 Mengatur *Layer Grid*

10. Buat hierarki baru di dalam Player, beri nama 'GroundCheck'



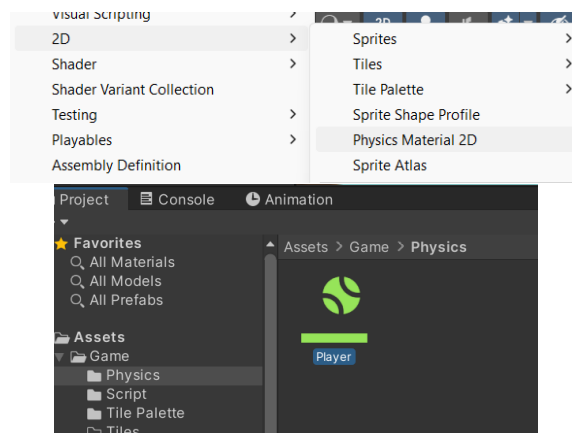
Gambar 6.9 Menambahkan Hierarki 'GroundCheck'

11. Atur komponen Player bagian Coding seperti di bawah ini



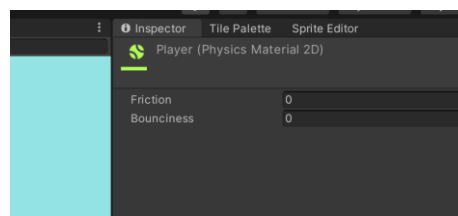
Gambar 6.10 Mengatur Komponen Player bagian Coding

12. Buat folder 'Physics' dan buat *Physics Material 2D* baru



Gambar 6.11 Membuat *Physics Material 2D* Baru

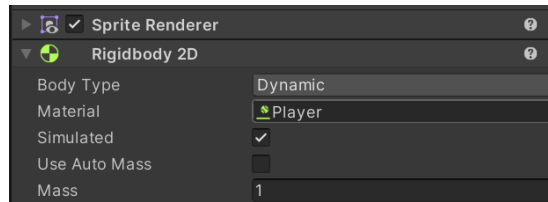
13. Konfigurasi Player di inspector dengan nilai 0 semua



Gambar 6.12 Konfigurasi Player

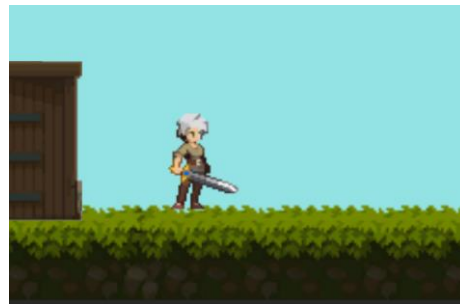


14. Ganti material pada komponen *Rigidbody 2D* Player dengan Player



Gambar 6.13 Mengganti Material *Rigidbody 2D*

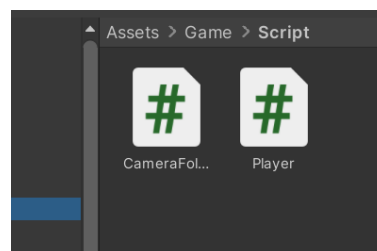
15. Jalankan proyek dan coba untuk lompat dengan spasi



Gambar 6.14 Uji Coba Lompat

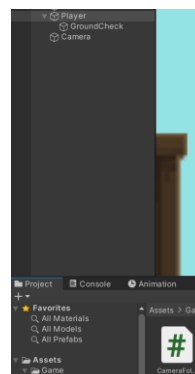
## B. Menggerakkan Kamera

1. Tambahkan *script* baru dengan nama 'CameraFollow'



Gambar 6.15 Buat berkas C# baru untuk Kamera

2. Sisipkan CameraFollow ke Main Camera



Gambar 6.16 Menyisipkan CameraFollow ke Kamera

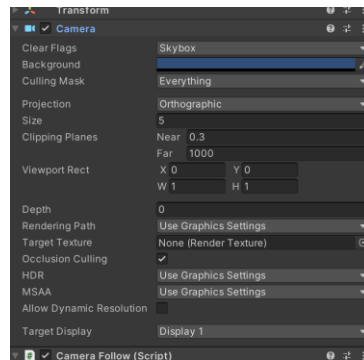


### 3. Masuk ke berkas CameraFollow dan masukkan skrip di bawah ini

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class CameraFollow : MonoBehaviour{
    public float xMargin = 0.5f;
    public float yMargin = 0.5f;
    public float xSmooth = 4f;
    public float ySmooth = 4f;
    public Vector2 maxXAndY;
    public Vector2 minXAndY;
    private Transform player;
    void Awake(){
        player =
GameObject.FindGameObjectWithTag("Player").transform;
    }
    bool CheckXMargin(){
        return Mathf.Abs(transform.position.x -
player.position.x) > xMargin;
    }
    bool CheckYMargin(){
        return Mathf.Abs(transform.position.y -
player.position.y) > yMargin;
    }
    void FixedUpdate(){
        TrackPlayer();
    }
    void TrackPlayer(){
        float targetX = transform.position.x;
        float targetY = transform.position.y;
        if (CheckXMargin())
            targetX = Mathf.Lerp(transform.position.x,
player.position.x,
            xSmooth * Time.deltaTime);
        if (CheckYMargin())
            targetY = Mathf.Lerp(transform.position.y,
player.position.y,
            ySmooth * Time.deltaTime);
        targetX = Mathf.Clamp(targetX, minXAndY.x,
maxXAndY.x); targetY =
        Mathf.Clamp(targetY, minXAndY.y, maxXAndY.y);
        transform.position = new Vector3(targetX, targetY,
transform.position.z);
    }
}
```

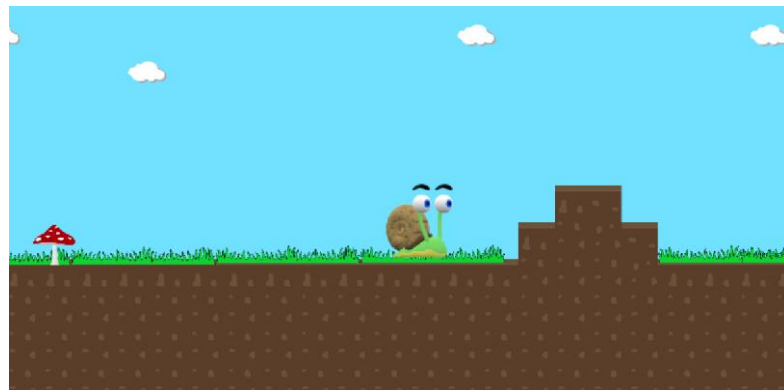


4. Atur komponen di Main Camera seperti ini



Gambar 6.17 Menyesuaikan Komponen di Main Camera

5. Jalankan proyek dan coba untuk bergerak lebih jauh hingga kamera bergerak



Gambar 6.18 Uji Coba Pergerakan Kamera

### C. Link Github Pengumpulan

<https://github.com/andhikatok/praktikum-animasi-game/tree/main/TUGAS%208>





#### **D. Kuis**

Penjelasan source code langkah ke-6 Menggerakkan Karakter

Terdapat 5 function yang berjalan pada karakter, yaitu Awake(), Update(), FixedUpdate(), GroundCheck(), dan Move(). Func Awake() berfungsi membaca komponen Rigidbody 2D di dalam karakter Player. Func Update() berfungsi mengatur kondisi karakter ketika lompat atau tidak. Func FixedUpdate() untuk memberikan perintah ke karakter untuk berjalan atau melompat. Func GroundCheck() berfungsi mengecek karakter bertapak atau tidak, jika tidak bertapak maka karakter akan turun. Func Move() berfungsi menggerakkan karakter khususnya maju dan mundur.

Penjelasan source code langkah ke-6 Menggerakkan Karakter

Terdapat 5 function yang bekerja pada kamera, yaitu Awake(), CheckXMargin(), CheckYMargin(), FixedUpdate(), dan TrackPlayer(). Func Awake() berfungsi mendefinisikan kamera ke obyek yang memiliki tag 'Player'. Func CheckXMargin() berfungsi mengecek margin sumbu X kamera. Func CheckYMargin() berfungsi mengecek margin sumbu Y kamera. Func TrackPlayer() berfungsi mengecek posisi karakter dan mengikuti karakter. Func FixedUpdate() berfungsi menjalankan func TrackPlayer().