



## TUGAS PERTEMUAN: 8

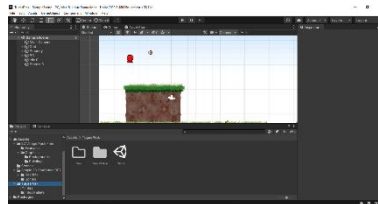
### CAMERA & CHARACTER MOVEMENT

NIM	:	2118139
Nama	:	Farhan Maulana Ridho
Kelas	:	C
Asisten Lab	:	Bagas Anardi Surya W (2118004)

#### 8.1 Tugas 8 : Membuat Character Movement, Detect Ground, Jumping, & Camera Movement

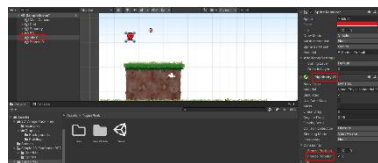
##### A. Membuat Pergerakan Player

1. Buka project Unity sebelumnya yang telah diimpor.



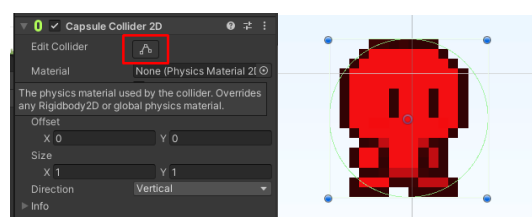
Gambar 8.1 Tampilan *Project Unity*

2. Berikutnya, tambahkan komponen Rigidbody 2D pada player. Kemudian, tambahkan juga komponen Capsule Collider dan sesuaikan ukurannya dengan objek player.



Gambar 8.2 *Capsule Collider Player*

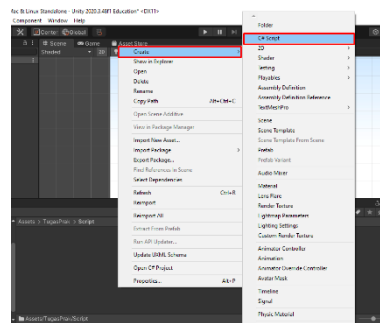
3. Selanjutnya pada *Hierarchy player*, pergi ke *inspector>Capsule Collider 2D>Edit Collider*. Tekan tombol yang di *edit collider* kemudian pilih *scale tool* untuk menyesuaikan garis lingkaran dengan *player*.



Gambar 8.3 *Capsule Collider Scale*

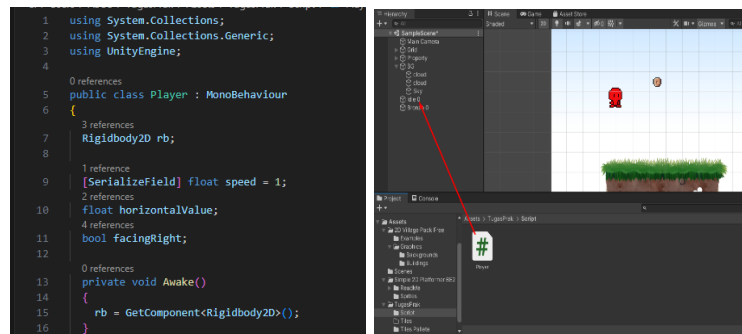


4. Langkah selanjutnya adalah membuat skrip untuk player. Buat folder bernama "Script" di dalam folder "TugasPrak". Di dalam folder tersebut, buat skrip dan beri nama "Player".



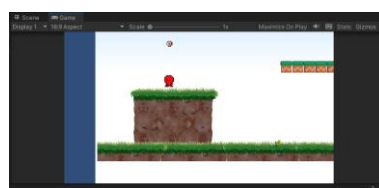
Gambar 8.4 Membuat Script

5. Buka skrip dan masukkan kode sumber yang telah disediakan oleh modul. Setelah itu, seret dan jatuhkan skrip tersebut ke objek player di Hierarchy.



Gambar 8.5 Souce code

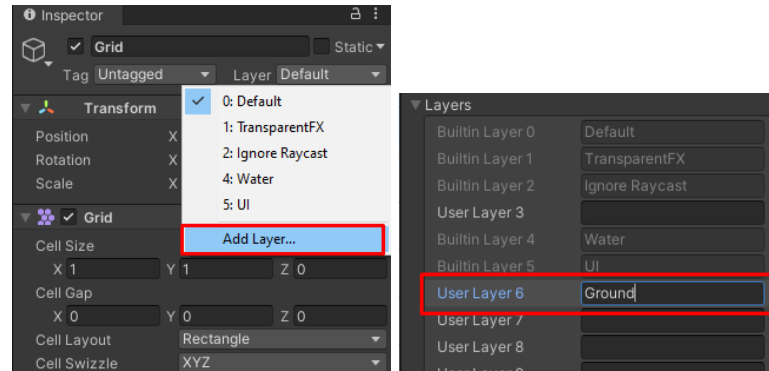
6. Uji apakah kodenya berfungsi dengan menekan tombol keyboard a, d, untuk bergerak ke arah kiri dan kanan.



Gambar 8.6 Mencoba Source Code

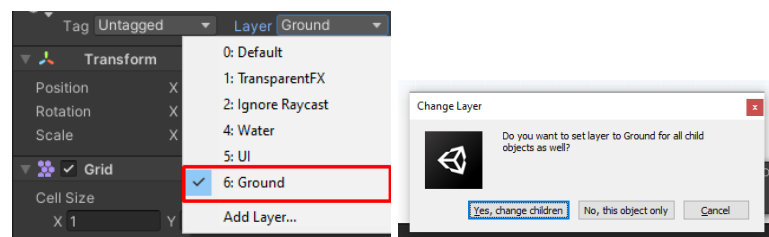


7. Selanjutnya membuat player bisa melompat saat menekan spasi. Pertama, buat GroundCheck dengan cara sebagai berikut: pada Inspector Grid, pilih layer, kemudian klik Add Layer. Isi "Ground" pada layer 6.



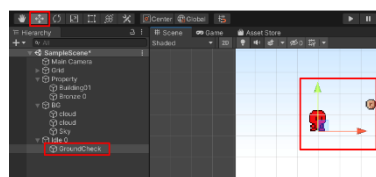
Gambar 8.7 GroundCheck Layer 6

8. Kemudian, ubah layer menjadi "Ground". Jika muncul pop-up Change Layer, klik saja "Yes".



Gambar 8.8 Change Layer

9. Selanjutnya, buat objek GroundCheck pada hierarki player. Klik GroundCheck dan ubah posisinya menggunakan Move Tools untuk memindahkannya ke bagian bawah player.



Gambar 8.9 GroundCheck

10. Tambahkan juga pada script player pada bagian voidFixedUpdate dan voidGroundCheck tambahkan source code ini.



Gambar 8.10 Source code

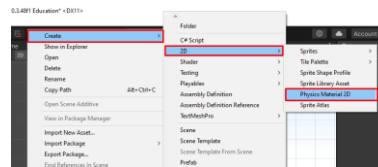


11. Setelah itu, tambahkan kode ini ke dalam skrip untuk player melompat dan tambahkan pada voidUpdate.

```
25 void Update()
26 {
27     horizontalValue = Input.GetAxisRaw("Horizontal");
28     if (Input.GetButtonDown("Jump"))
29     {
30         jump = true;
31     }
32     else if (Input.GetButtonUp("Jump"))
33     {
34         jump = false;
35     }
36 }
37
38 0 references
39 void FixedUpdate()
40 {
41     GroundCheck();
42     Move(horizontalValue, jump);
43 }
```

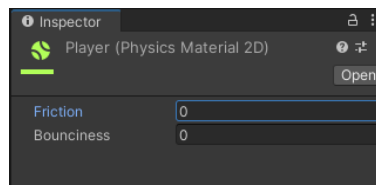
Gambar 8.11 Source code untuk lompat

12. Selanjutnya membuat folder baru di dalam folder "TugasPrak" dan beri nama "Physics". Kemudian, buat Physical Material 2D dan beri nama "Player".



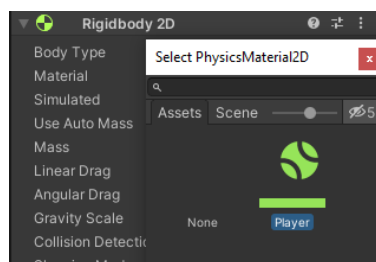
Gambar 8.12 PhysicalMaterial 2D

13. Pada folder player, pergi ke inspector Physics Material 2D ubah *friction* dan *Bounciness* menjadi 0.



Gambar 8.13 Inspector Physics Material 2D

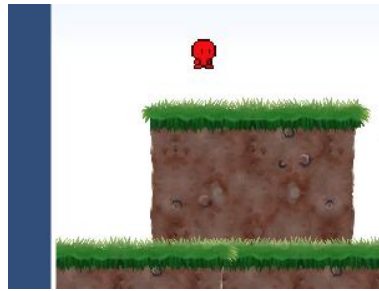
14. Pada hirarki player, kemudian pada inspector bagian Rigidbody2D, kita ubah material menjadi player.



Gambar 8.14 Box Collider 2D Player



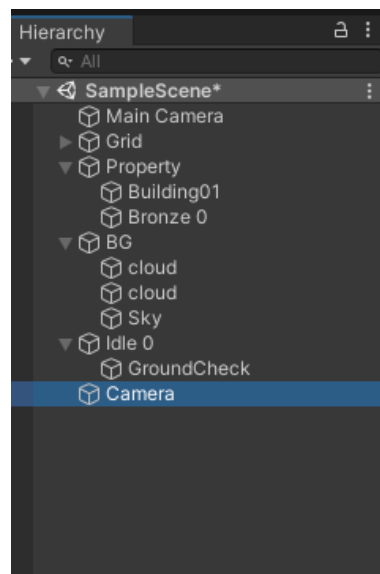
15. Pada tombol play tekan, kemudian tekan spasi maka player bisa melompat.



Gambar 8.15 Spasi untuk melompat

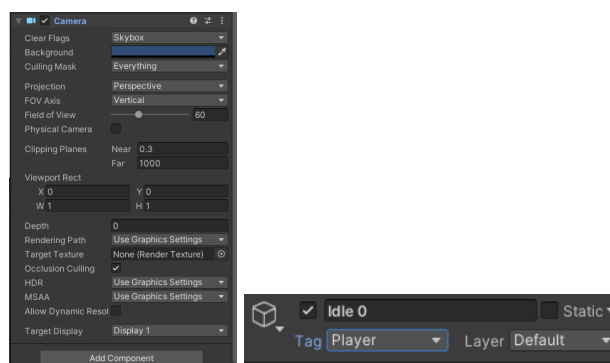
## B. Camera Movement

1. Pada hierarchy klik kanan>create empty beri nama "Camera".



Gambar 8.15 Layer Camera

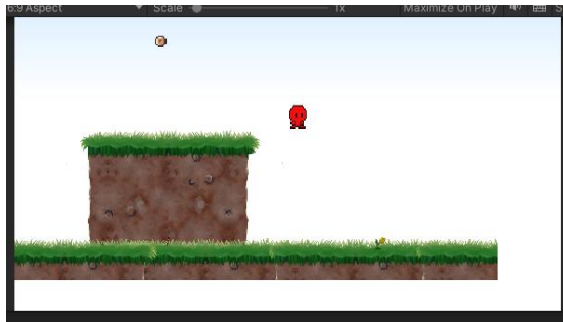
2. Langkah selanjutnya pada hierarchy Camera, tambahkan add component dengan nama camera. Setelah itu sesuaikan dengan yang sudah dibuat. Kemudian pada hierarchy player pada inspector, tag gani dengan "Player".



Gambar 8.16 Camera Follow



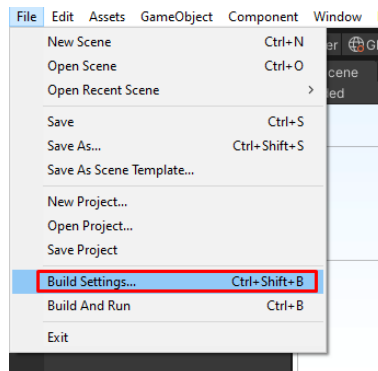
3. Langkah terakhir jalankan dengan tekan tombol play.



Gambar 8.17 Play Project

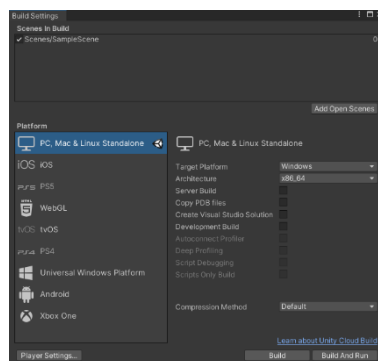
### C. Render

1. Untuk proses merender, buka file kemudian pilih Build Settings. Setelah masuk ke Build Settings, pilih PC, Mac, dan Linux. Jangan lupa mencentang proyek kita pada bagian Scenes in Build. Jika belum ada, tekan Add Open Scenes.



Gambar 8.18 Render Settings

2. Setelah itu kita klik Build dan kita pilih project jadinya akan disimpan dimana. Dan tunggu hasilnya.



Gambar 8.19 Building Render

### D. Link Pengumpulan Github

Link : [https://github.com/farhan2153/2118139\\_PRAK\\_ANIGAME.git](https://github.com/farhan2153/2118139_PRAK_ANIGAME.git)



# KUIS

Menjelaskan Source Code dibawah ini :

## X. Kuis Camera Follow

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class CameraFollow : MonoBehaviour
{
    [SerializeField] private Transform player;
    void Update() {
        transform.position = new Vector3 (player. position.x,
        transform.position.y, transform.position.z);
    }
}
```

Analisa :

Kode di atas adalah script untuk Unity yang mengatur agar kamera mengikuti pemain. Pertama, namespace System.Collections, System.Collections.Generic, dan UnityEngine diimpor untuk menyediakan fungsionalitas dasar. Kelas CameraFollow dideklarasikan sebagai publik dan mewarisi dari MonoBehaviour, yang memungkinkan integrasi dengan sistem komponen Unity. Variabel player dideklarasikan sebagai private dan ditandai dengan atribut [SerializeField], sehingga dapat diatur melalui inspector di editor Unity. Dalam method Update, yang dipanggil setiap frame, posisi kamera diperbarui dengan mengatur nilai x ke posisi x pemain (player.position.x), sedangkan nilai y dan z tetap menggunakan posisi kamera saat ini (transform.position.y dan transform.position.z). Hal ini membuat kamera mengikuti pergerakan pemain pada sumbu x, sementara sumbu y dan z tetap konstan.