



## TUGAS PERTEMUAN: 9

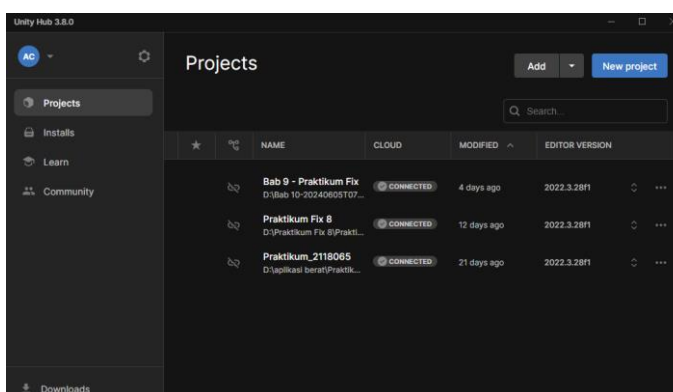
### ANIMATION

NIM	:	2118065
Nama	:	Alga Candra Kusuma
Kelas	:	B
Asisten Lab	:	DEVINA DORKAS MANUELA (2218108)
Referensi	:	Contoh : <a href="#">FREE Pixel Art Kit</a>   <a href="#">2D Environments</a>   <a href="#">Unity Asset Store</a>

### 9.1 Tugas 1 : ANIMATION

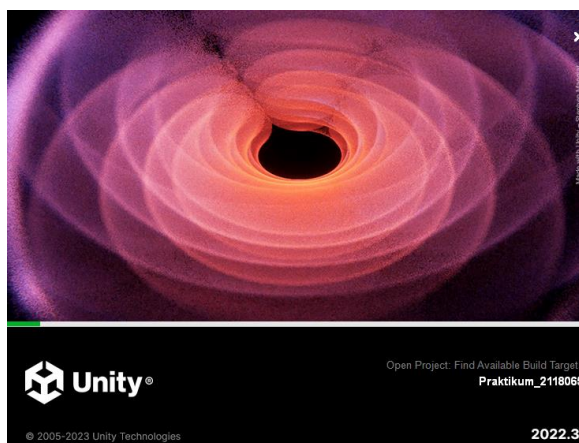
#### A. Animation

1. Masuk ke dalam aplikasi unity



Gambar 9.1 Membuka Unity

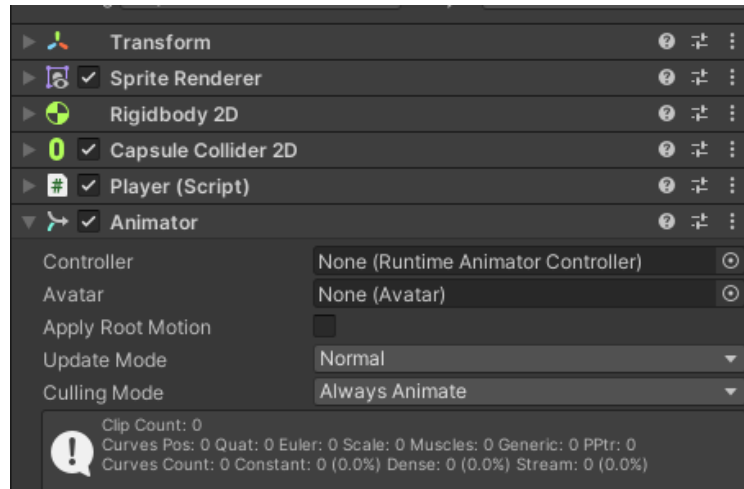
2. Pilih folder yang ingin dibuka atau yang sudah diunduh property



Gambar 9.2 mebuks file unity

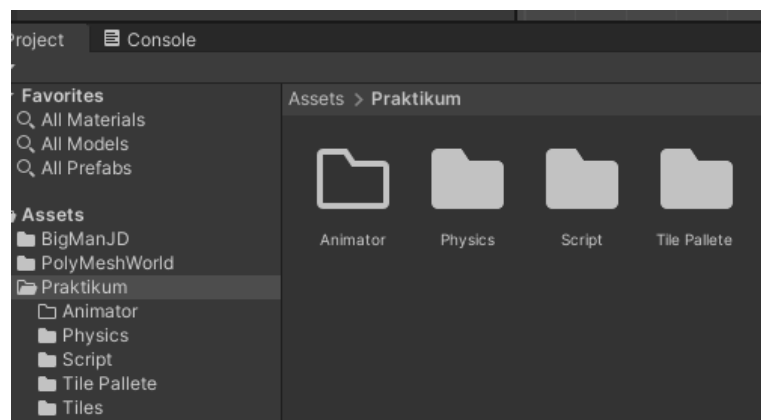


3. Pada hirarki player tambahkan komponen animator



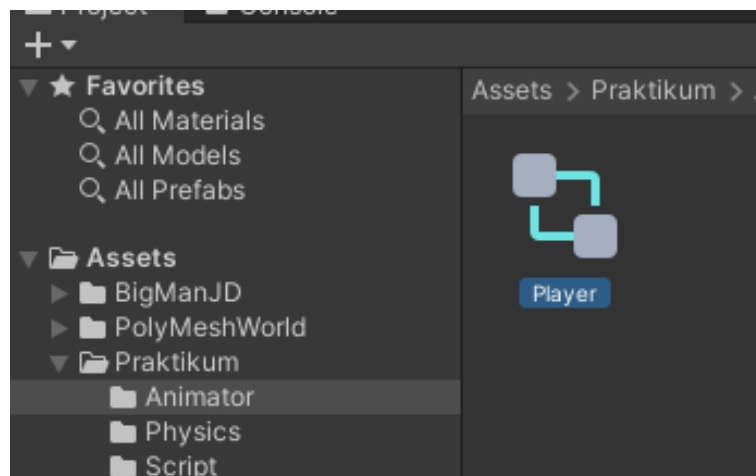
Gambar 9.3 Menambahkan komponen animator

4. Buat folder baru Bernama Animator dalam folder praktikum



Gambar 9.4 menambahkan folder animator

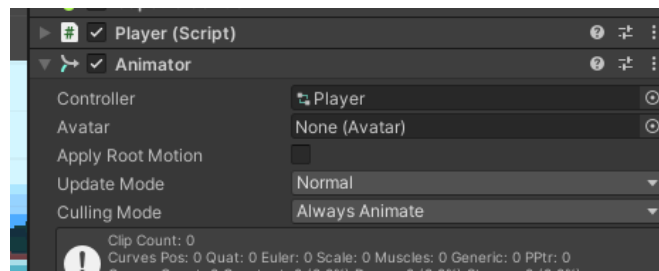
5. Membuat animator kontroler pada folder animator dgn nama player



Gambar 9.5 menambahkan kontrollor

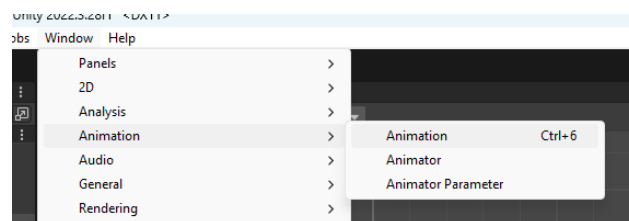


6. Pada komponen hirarki player, ubah target controller ke player



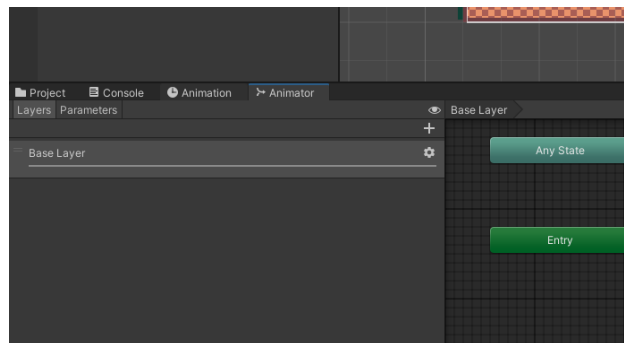
Gambar 9.6 mengubah controller ke player

7. Tambahkan window animation dan animator



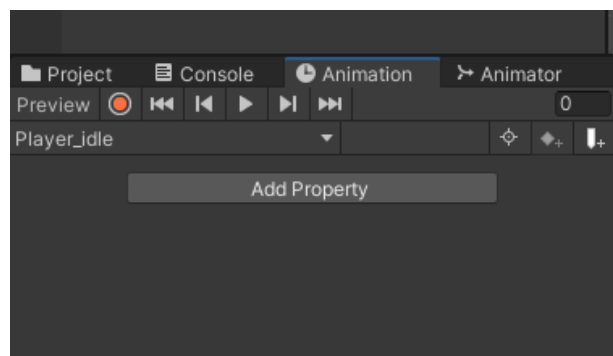
Gambar 9.7 menambahkan animation dan animator

8. Drag and drop window pada tempat yang sesuai keinginan pengguna



Gambar 9.8 penerapan movement

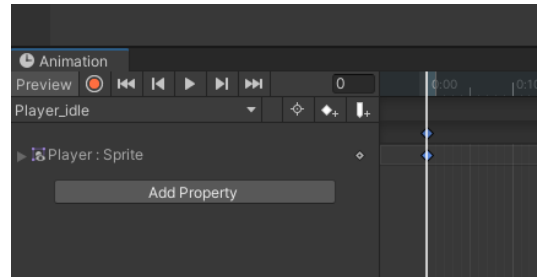
9. Setelah menambahkan window, pada animator klik create dan simpan pada folder animator



Gambar 9.9 penerapan movement

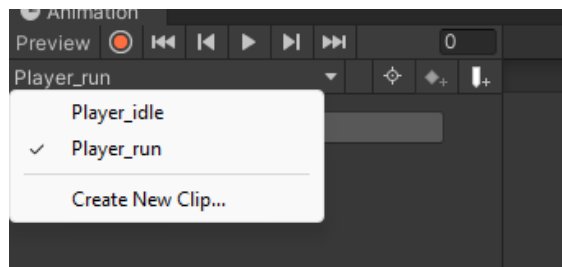


10. Pada grid property asset ambil animasi karakter idle lalu drag and drop pada animation



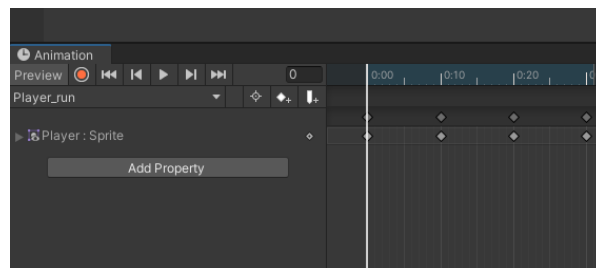
Gambar 9.10 Menambahkan idle animation

11. Create new clip dan rename menjadi player run untuk animasi lain



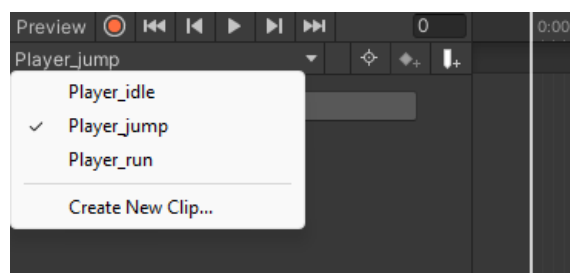
Gambar 9.11 Menambahkan Player run

12. Pada grid property asset ambil animasi karakter run lalu drag and drop pada animation



Gambar 9.12 Menambahkan run animation

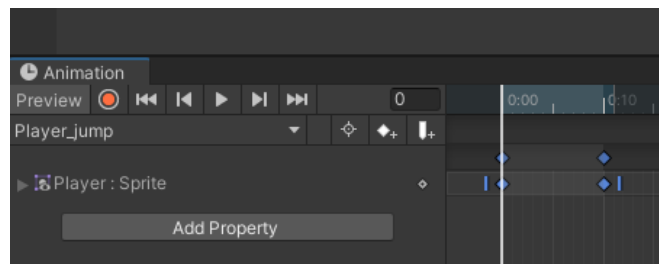
13. Create new clip dan rename menjadi player jump untuk animasi lain



Gambar 9.13 Menambahkan Player\_jump

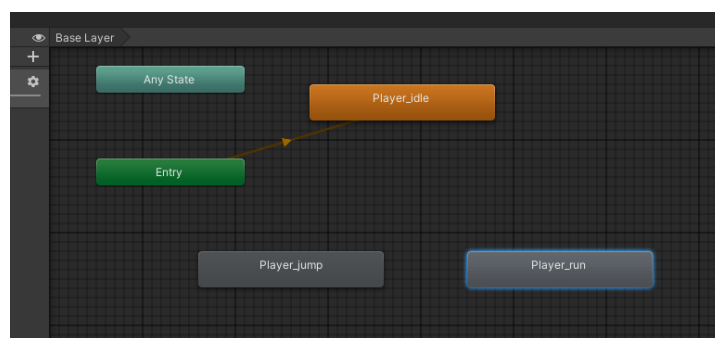


14. Pada grid property asset ambil animasi karakter jump lalu drag and drop pada animation



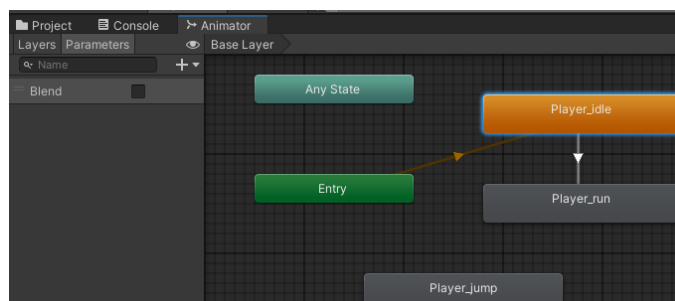
Gambar 9.14 Membuat folder physics

15. Pindah ke dalam window animator



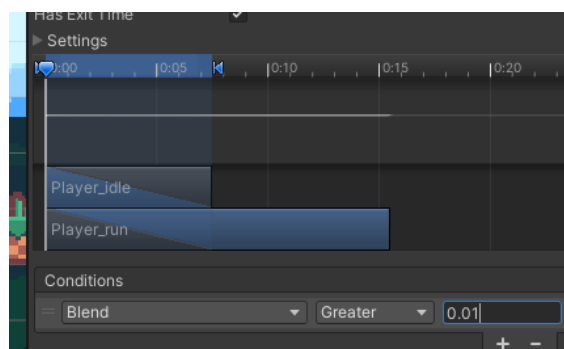
Gambar 9.15 tampilan window animator

16. Pada parameter player idle make transition dan tambahkan tipe data



Gambar 9.16 Menambah parameter dan transition

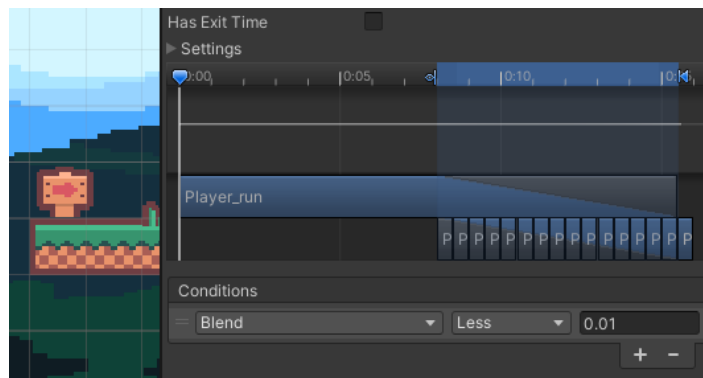
17. Pada klik panah putih transition ubah nilai menjadi 0.01



Gambar 9.17 ubah bagian setting transition

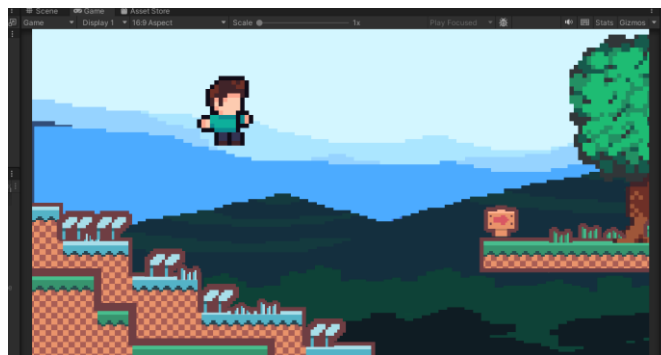


18. Buat transition sebaliknya dengan cara yang sama namun pada pilihan settings ubah greater menjadi less



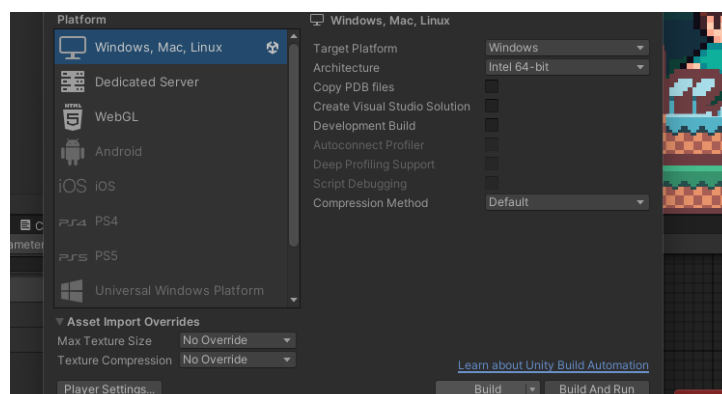
Gambar 9.18 ubah bagian setting transition menjadi less

19. Hasil tampilan penerapan animasi movement mulai dari lompat, jatuh, idle dan run



Gambar 9.19 Tampilan penerapan animasi movement

20. Hasil game yang sudah diberi movement animation disimpan kemudian render



Gambar 9.20 Merender File



## B. Kuis

```
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);
    }
}
```

### Penjelasan

#### 1. Dalam fungsi HandleJumpInput:

- `animator.SetBool("isJumping", );` harus memiliki nilai boolean kedua, seperti true atau false.
- `else if (Input.GetKey(KeyCode.Space))` seharusnya `else if (Input.GetKeyUp(KeyCode.Space))` untuk mendeteksi ketika tombol space dilepas.



## 2. Dalam fungsi HandleMovementInput:

- if (move != 1) seharusnya if (move != 0) untuk mengecek apakah ada input gerakan.
- animator.SetBool("isWalking", false); seharusnya animator.SetBool("isWalking", true); ketika karakter bergerak.
- transform.Translate(Vector3.left \* move \* Time.deltaTime); menggunakan Vector3.left yang tidak perlu dikalikan dengan move karena sudah merepresentasikan arah ke kiri. Sebaiknya gunakan Vector3.right dan kalikan dengan move untuk mendapatkan arah yang benar.
- Blok kode yang mengatur transform.localScale:
  - if (move != 0) dan else if (move > 0) saling bertentangan karena move > 0 adalah subset dari move != 0. Ini harus diatur ulang untuk memastikan skala karakter berubah sesuai dengan arah gerakan.