



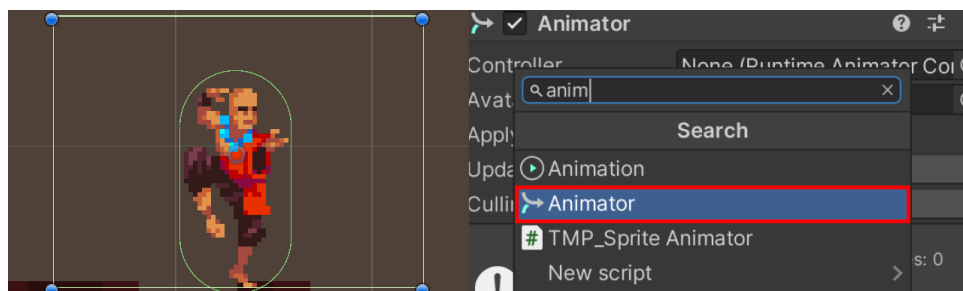
## TUGAS PERTEMUAN: 9

### Game Animation

NIM	:	2118105
Nama	:	Yonanda Haryono
Kelas	:	C
Asisten Lab	:	Rifal Rifqi Rhomadon

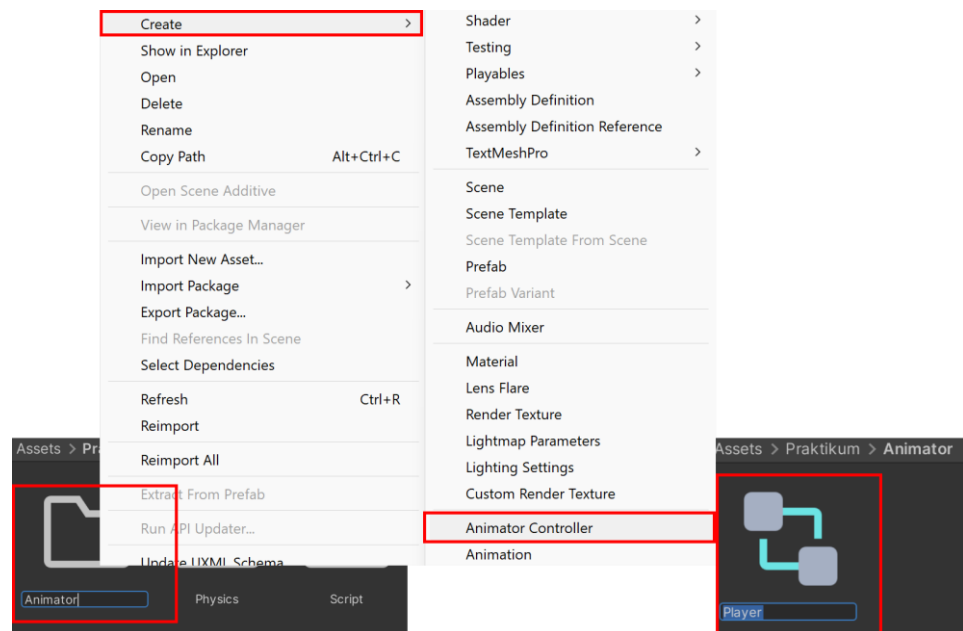
#### 9.1 Tugas 9 : Membuat Game Animation

1. Pada karakter klik inspector kemudian pilih Add Component Animator.



Gambar 9.1 Menambahkan Component Animator

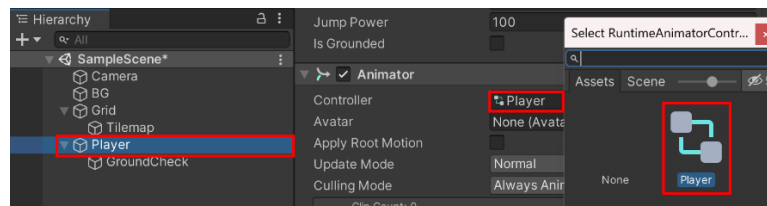
2. Pada folder Praktikum Buat Folder baru “Animator”, Kemudian buat File Animator Controller pada folder Animator , ubah namanya menjadi Player



Gambar 9.2 Membuat Animator Controller

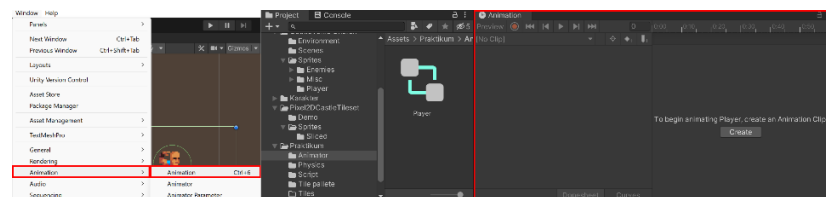


3. Klik player pada Hierarchy, kemudian cari Component Animator, pada setting Controller ubah menjadi Player.



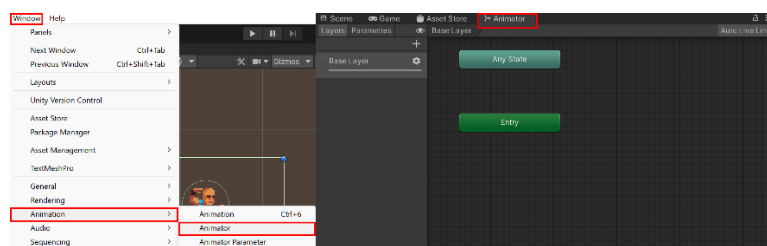
Gambar 9.3 Menambahkan Animator Controller Pada Player

4. Tambahkan menu panel Animation di menu Window, pilih Animation, pilih Animation atau tekan Shortcut CTRL + 6. Maka Akan muncul menu panel baru, geser panel tersebut dibawah sendiri seperti gambar dibawah ini, Split Panel Project dan Animation.



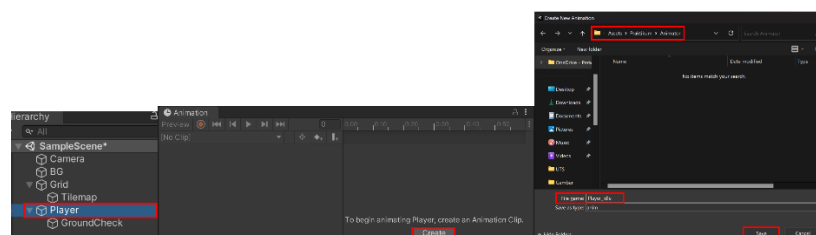
Gambar 9.4 Tampilan Panel Animation

5. Tambahkan menu panel Animator, Kemudian geser Panel tersebut sesuai dengan gambar.



Gambar 9.5 Tampilan Panel Animator

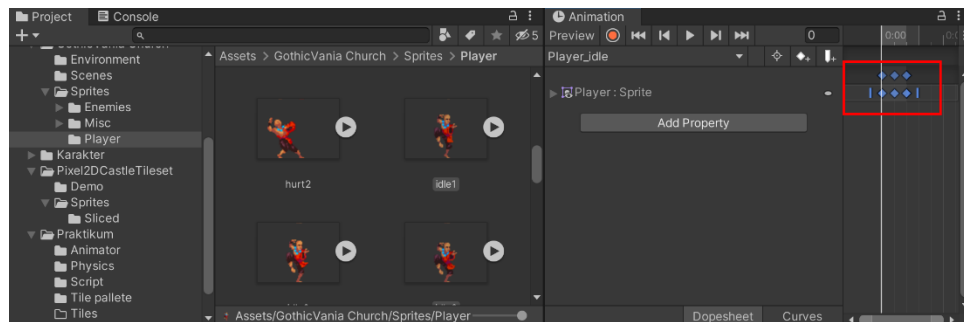
6. Untuk membuat animasi klik Player pada Hierarchy, kemudian ke menu panel Animation, pilih Create. Kemudian simpan pada folder Animator dan beri nama "Player\_idle".



Gambar 9.6 Menambahkan Animation Clip

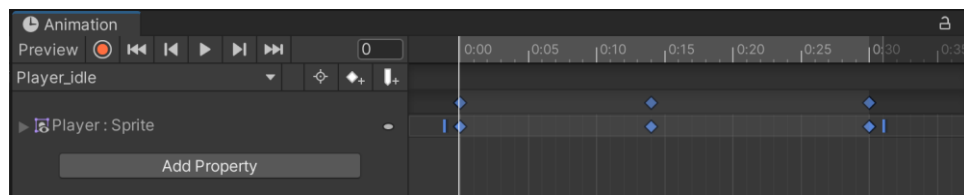


7. Pada menu Project buka folder player dan pilih gambar idle1, idle-3 dan idle4, kemudian drag ke tab Animation.



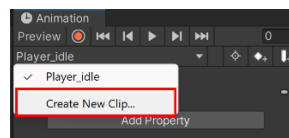
Gambar 9.7 Tampilan Menambahkan Player Idle

8. Tekan CTRL + A pada menu panel Animation geser kotak kecil pada timeline sampai frame 0:30 agar animasinya tidak terlalu cepat.



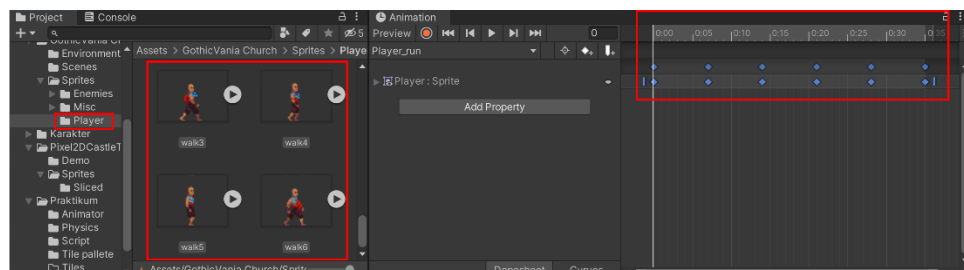
Gambar 9.8 Tampilan Mengubah Timeline Player Idle

9. Buat animasi baru, Klik pada “Player\_idle” kemudian pilih Create New Clip, dan beri nama “Player\_run”, Simpan pada Folder Animator.



Gambar 9.9 Tampilan Membuat New Clip Player\_run

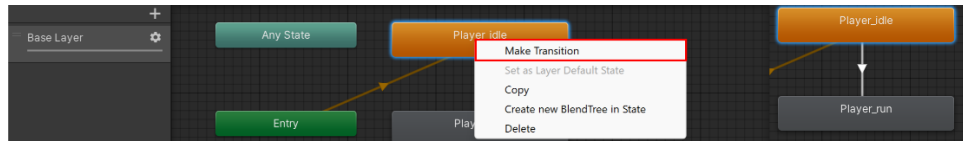
10. Pada menu Project buka folder player dan pilih gambar walk1 sampai walk6, kemudian drag ke tab Animation. Kemudian pada panel timeline tekan Ctrl+A di keyboard, klik bagian kotak kecil disamping keyframe terakhir dan geser sampai waktu 0:35



Gambar 9.10 Tampilan Menambahkan Player Run

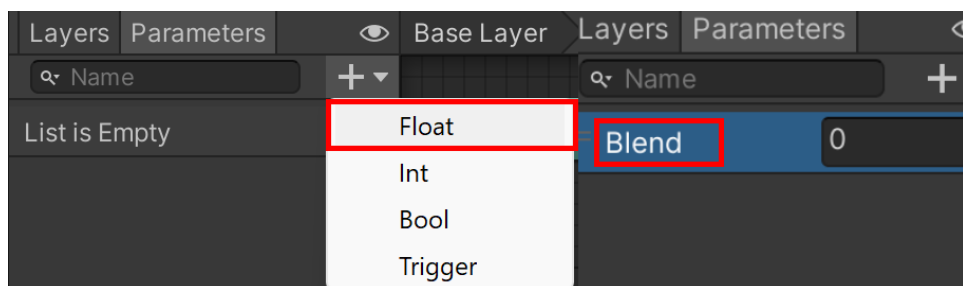


11. Pilih ke menu Animator yang telah dibuka sebelumnya, Kemudian buat transisi antara `player_idle` dan `player_run` dengan cara klik kanan pada `player_idle` dan pilih `Make Transition` dan tarik ke `player_run`.



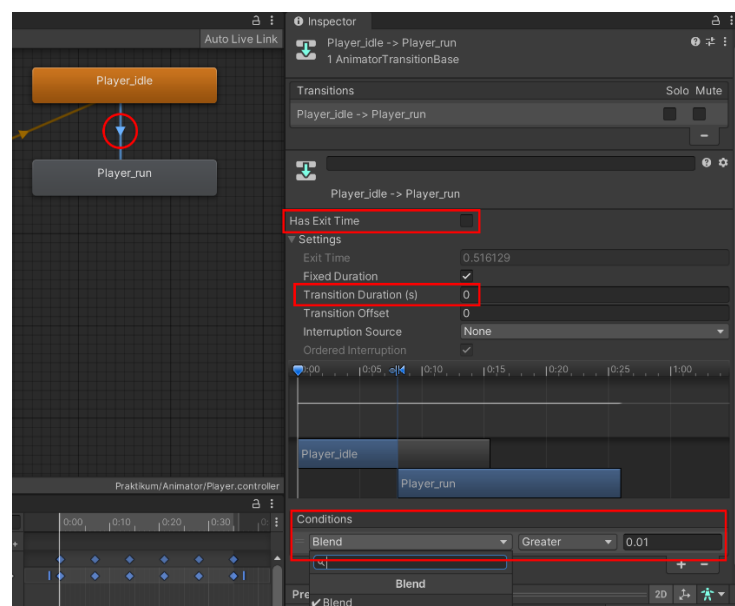
Gambar 9.11 Menambahkan Transition `Player_idle` ke `Player_run`

12. Masuk ke tab parameter, tambahkan tipe data dengan cara tekan icon tambah dan ubah namanya menjadi “Blend”



Gambar 9.12 Tampilan Menambahkan Parameters

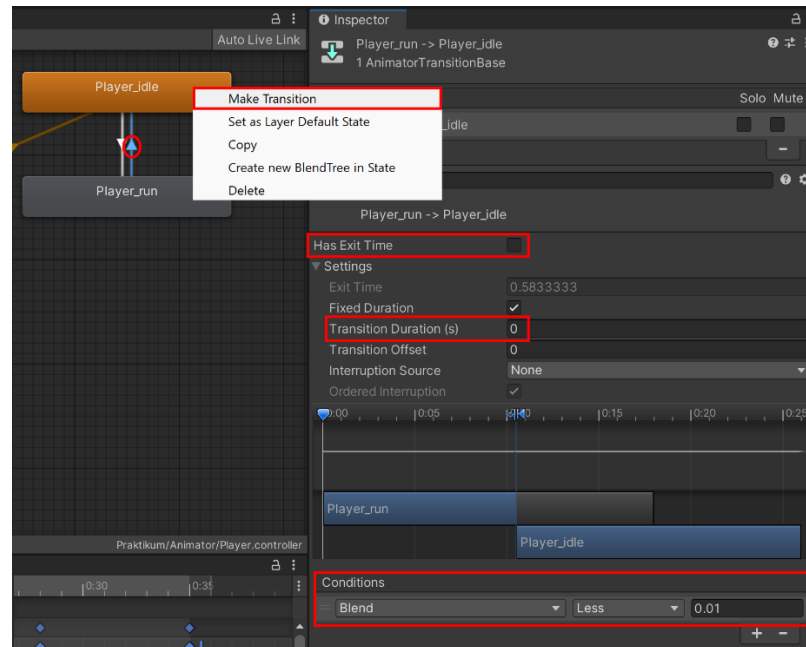
13. Klik panah putih tersebut, pada bagian conditions klik icon tambah kemudian atur menjadi “Blend”. Atur nilai conditions blend tersebut menjadi 0.01. Pada bagian Settings, hilangkan centang pada `Has Exit Time` dan atur nilai `Transition Duration` menjadi 0



Gambar 9.13 Tampilan Menambahkan



14. Buat transisi juga dari player\_run ke player\_idle dengan cara klik kanan pada player\_run dan pilih Make Transition. Tambahkan parameter transisi dengan tipe data Float. Klik ikon tambah dan rename menjadi “Blend”. Setelah itu, ubah operator dari Greater menjadi Less dan atur nilainya menjadi 0.01. Kemudian pada bagian Settings, hilangkan centang pada Has Exit Time dan atur nilai Transition Duration menjadi 0.



Gambar 9.14 Tampilan Menambahkan Transisi player\_run ke player\_idle

15. Agar animasi dapat sesuai ketika berjalan, buka script Player dan tambahkan source code berikut pada class Player.

```
public class Player : MonoBehaviour
{
    public Animator animator;
    Rigidbody2D rb;
}
```

Gambar 9.15 Menambahkan Script Dibawah Public Class

```
public Animator animator;
```

16. Tambahkan Script Komponen Animator

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

Gambar 9.16 Menambahkan Script Pada Function Awake

```
animator = GetComponent<Animator>();
```



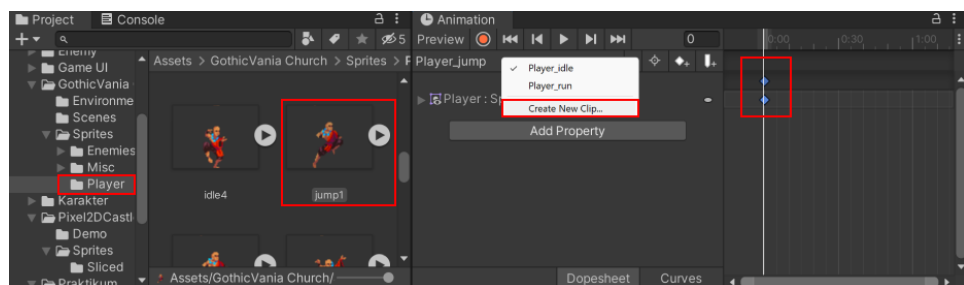
17. Dan pada fungsi FixedUpdate tambahkan source code berikut

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

Gambar 9.17 Menambahkan Script Pada Function FixedUpdate

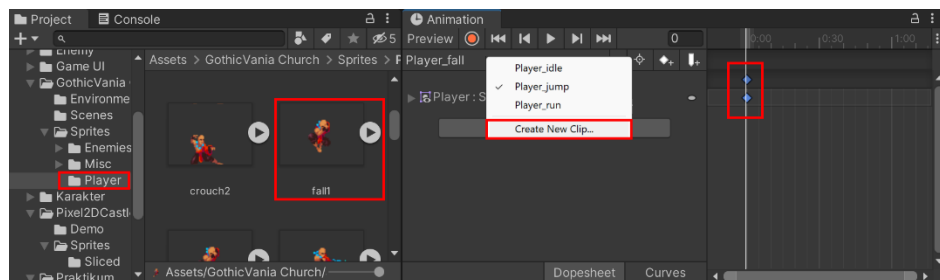
```
animator.SetFloat("Blend", Mathf.Abs(rb.velocity.x));
```

18. Kemudian buat animasi baru tekan tulisan “Player\_run” kemudian pilih Create New Clip, dan beri nama “Player\_jump”. Kemudian pada folder player pilih gambar jump1, kemudian drag ke tab Animation.



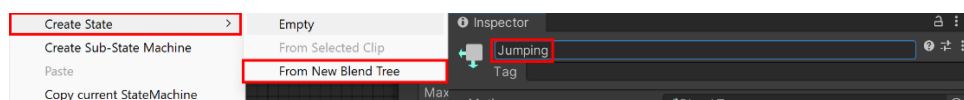
Gambar 9.18 Tampilan Menambahkan Clip Player\_jump

19. Buat animasi baru dengan cara tekan tulisan “Player\_jump” kemudian pilih Create New Clip, dan beri nama “Player\_fall”. Pada tab Project buka folder karakter pilih gambar fall1, kemudian drag ke tab Animation.



Gambar 9.19 Tampilan Menambahkan Clip Player\_fall

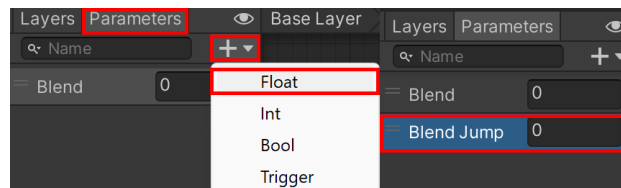
20. Kemudian untuk menambahkan animasi ketika melompat. Klik kanan pada menu Animator, di area kosong, pilih Create State kemudian pilih From New Blend Tree. Pada Animator klik Blend Tree, di menu Inspector, ubah namanya menjadi Jumping.



Gambar 9.20 Tampilan Menambahkan Blend Tree

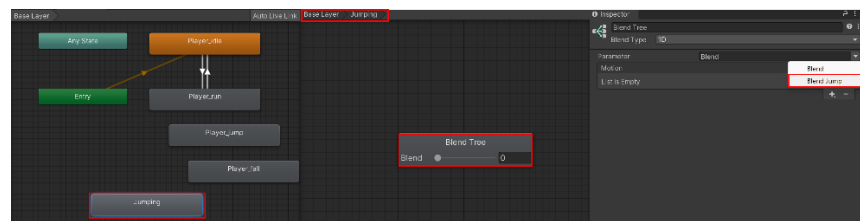


21. Pada menu Parameters tambahkan parameter tipe data Float tekan icon + dan ubah namanya menjadi “Blend Jump”.



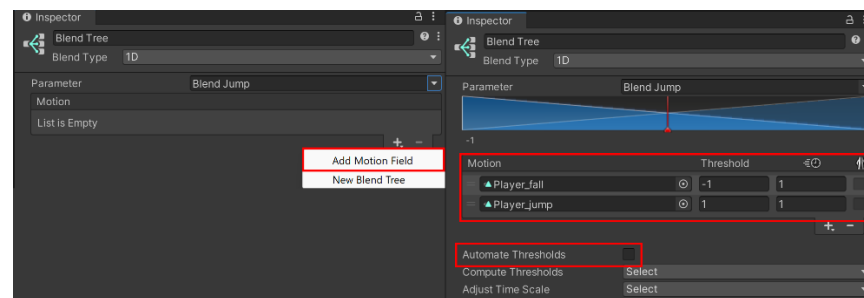
Gambar 9.21 Tampilan Menambahkan Parameter Blend Jump

22. Pada menu Animator, Klik dua kali pada Blend Tree “Jumping”, Tekan pada Blend Tree, Kemudian klik 2X Blend Tree “Jumping”, pada inspector ubah parameter menjadi “Blend Jump”.



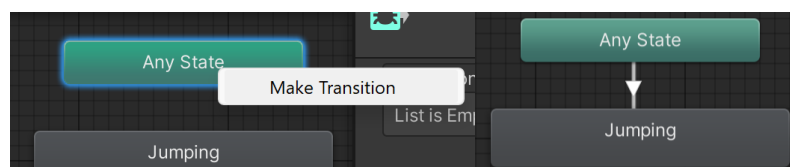
Gambar 9.22 Tampilan Mengubah Parameter Blend Tree Pada Jumping

23. Buka menu Inspector, tekan icon + dan pilih Add Motion Field. Tambahkan dua Motion Field. Hilangkan centang “Automate Thresholds” dan atur nilai Threshold seperti berikut. Klik bagian icon None (Motion), maka akan muncul Windows Motion, Tambahkan Sesuai dengan urutan.



Gambar 9.23 Tampilan Menambahkan Motion Field Pada Blend Tree

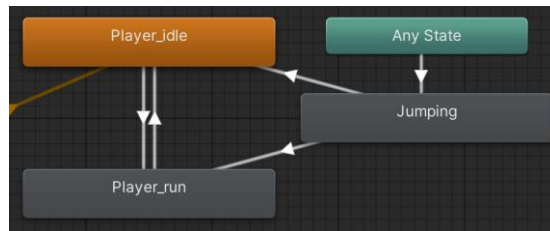
24. Kembali ke Base Layer, klik kanan Any State, pilih Make Transition dan arahkan panahnya ke Jumping.



Gambar 9.24 Menambahkan Transition Pada Any State ke Jumping

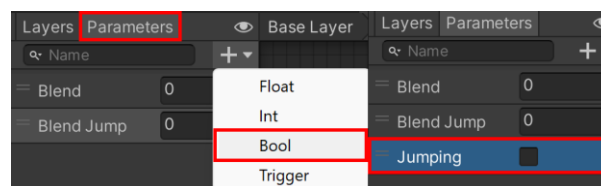


25. Klik kanan Jumping, pilih Make Transition dan arahkan panahnya ke Player\_idle dan Player\_run



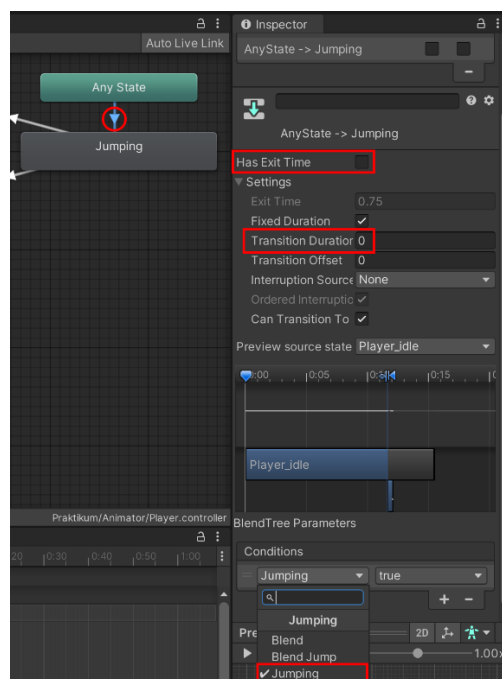
Gambar 9.25 Menambahkan Transition ke Player\_idle dan Player\_run

26. Tambahkan parameter transisi dengan tipe data Bool tekan icon + dan ubah namanya menjadi “Jumping”.



Gambar 9.26 Tampilan Menambahkan Parameter Jumping

27. Klik panah yang mengarah ke Jumping, pada inspector tambahkan condition, pilih condition Jumping dan ubah nilainya menjadi true. Kemudian klik Settings dan ubah nilai Transition Duration menjadi 0 dan hilangkan centang Has Exit Time

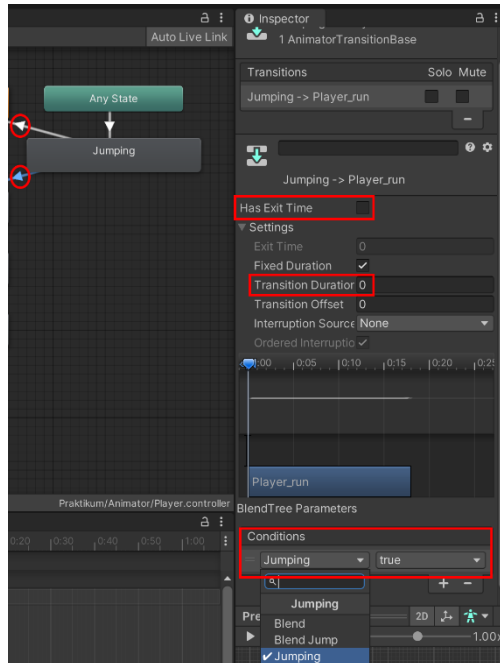


Gambar 9.27 Mengubah Inspector Panah Yang Mengarah Ke Jumping





28. Klik panah yang mengarah ke Player\_idle dan Player\_run, pada inspector tambahkan condition, pilih condition Jumping, pada arah panah ke player\_idle ubah menjadi false, pada arah panah ke player\_run ubah menjadi true. Kemudian klik Settings dan ubah nilai Transition Duration menjadi 0 dan hilangkan centang Has Exit Time



Gambar 9.28 Mengubah Inspector Panah Yang Mengarah Ke player\_idle dan player\_run

29. Buka script Player, dan tambahkan source code berikut pada fungsi update

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

Gambar 9.29 Menambahkan Script Pada Function Update

```
animator.SetBool("Jumping", true);
```

30. Pada Fungsi FixedUpdate tambahkan seperti berikut

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

    animator.SetFloat("Blend", Mathf.Abs(rb.velocity.x));
    animator.SetFloat("Blend Jump", rb.velocity.y);
}
```

Gambar 9.30 Menambahkan Script Pada Function FixedUpdate

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



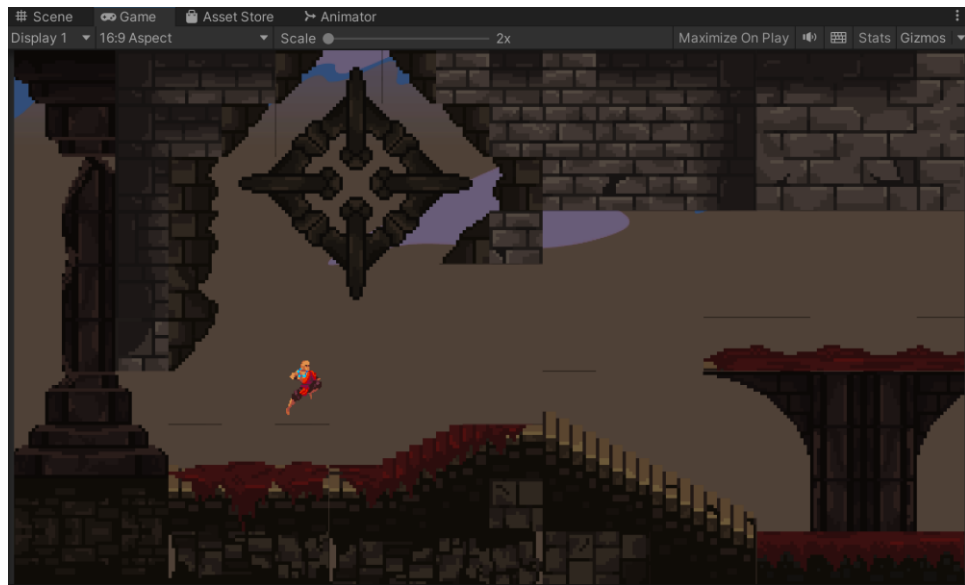
31. Tambahkan baris kode seperti dibawah ini dalam method GroundCheck

```
void GroundCheck()
{
    isGrounded = false;
    Collider2D[] colliders = Physics2D.OverlapCircleAll(groundcheckCollider.position, groundCheckRadius, groundLayer);
    if (colliders.Length > 0)
        isGrounded = true;
    animator.SetBool("Jumping", !isGrounded);
}
```

Gambar 9.31 Menambahkan Script Pada Function GroundCheck

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

32. Jika di play maka karakter sudah bisa bergerak dengan animasi



Gambar 9.32 Tampilan Hasil Running

## 9.2 Kuis CameraFollow

Lengkapi Source code dibawah ini :

(Berikan Tanda Merah yang menyebabkan Source code Error)

```
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);
}
}
```

### Analisa

Pada potongan kode tersebut terdapat beberapa kesalahan. Pada fungsi `HandleJumpInput`, terdapat dua panggilan `animator.SetBool("isJumping", );` yang tidak memiliki nilai boolean yang diatur, ini akan menyebabkan kesalahan sintaksis. Harusnya menetapkan nilai boolean seperti `true` atau `false`. Dalam `HandleMovementInput`, kondisi `if (move != 1)` salah karena seharusnya memeriksa `if (move == 0)` untuk menetapkan animator ke "isIdle". Selain itu, transformasi skala tidak konsisten; saat `move` tidak sama dengan 0, seharusnya mengatur skala berdasarkan arah gerakan horizontal (`move > 0` untuk skala kanan dan `move < 0` untuk skala kiri). Penempatan pengaturan animator juga salah dalam kondisi ini. Semua kesalahan ini bersama-sama mengakibatkan kode tidak berfungsi sesuai harapan.