



## TUGAS PERTEMUAN: 9

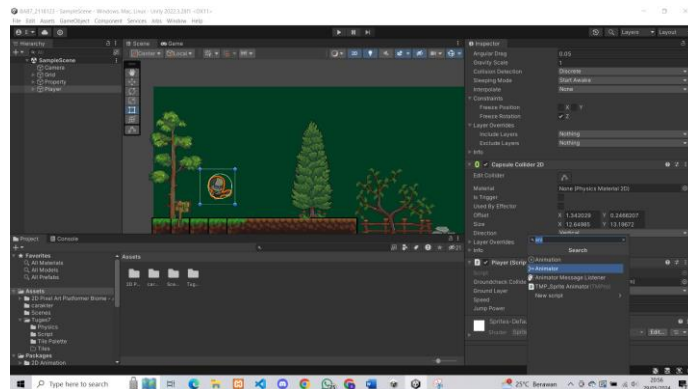
### GAME ANIMATION

NIM	:	2118123
Nama	:	M. Eri Kusyairi
Kelas	:	D
Asisten Lab	:	BERCHMANS BAYU BIN JAYA (2218034)

#### 1.1 Tugas 9 : Membuat Game Animation

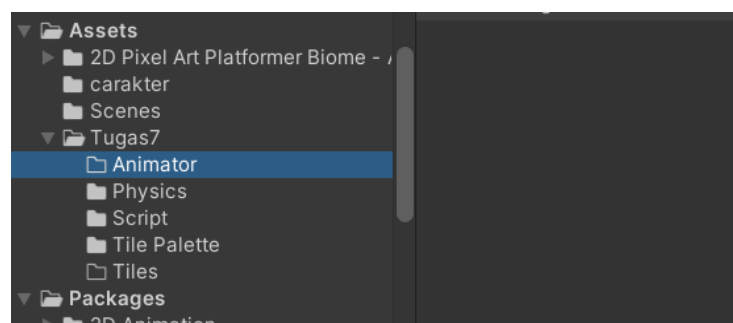
##### A. Character Animation

1. Masuk pada inspector player , lalu add Component Animator seperti berikut .



Gambar 9.1 add Component Animator

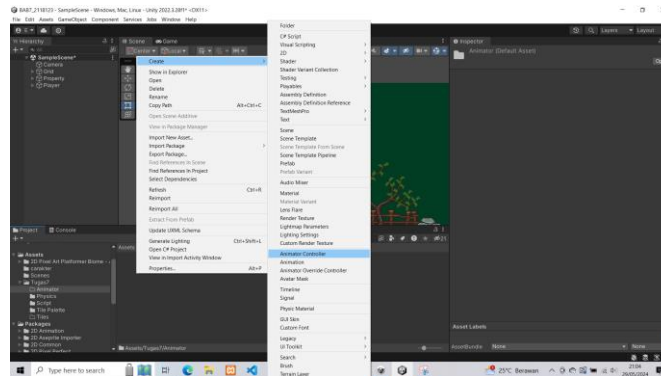
2. Create new Folder pada Folder Tugas 7 , rename menjadi “Animator”.



Gambar 9.2 Folder Animator

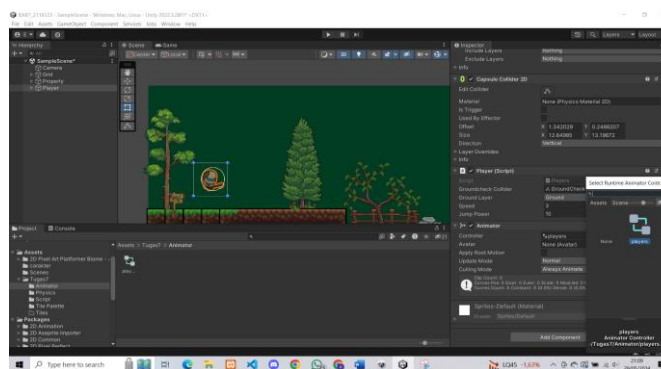


3. Buat file Animator Controller pada folder Animator , lalu rename menjadi “players”.



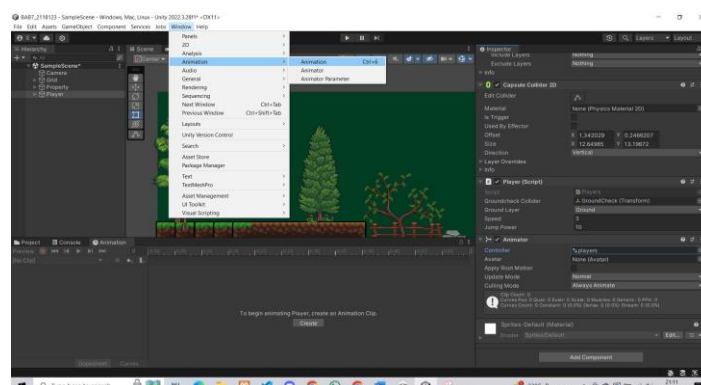
Gambar 9.3 Script Players

4. Setting Inspector pada player , masuk di Animator lalu setting Controller rubah menjadi “Players”.



Gambar 9.4 Controller Players

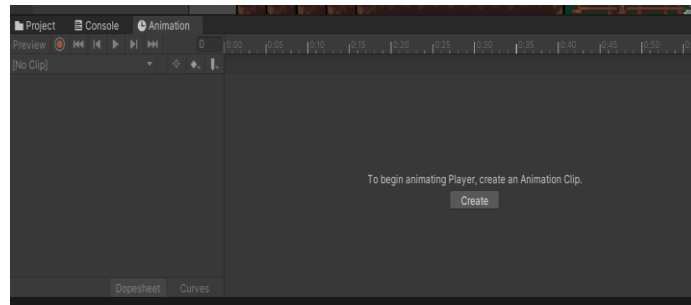
5. Selanjutnya masuk ke menu window lalu Pilih Animation - Animation .



Gambar 9.5 Animation

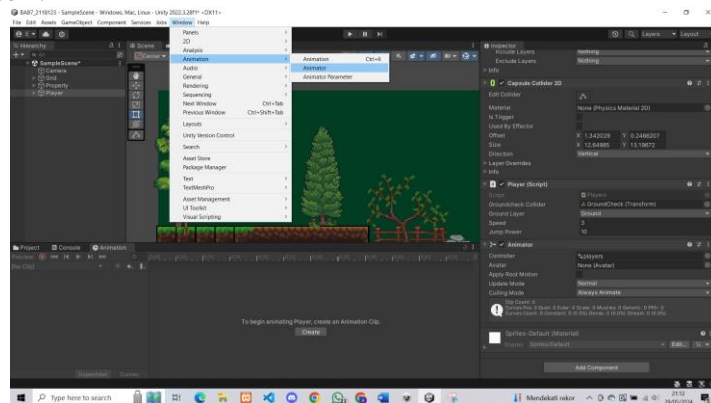


6. Jadikan 1 dengan project dan console pada menu bar di bawah ,seperti berikut ini.



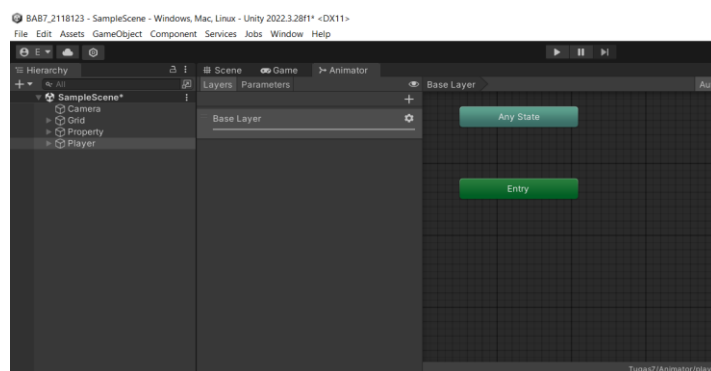
Gambar 9.6 Panel Animation.

7. Selanjutnya masuk ke window lagi , lalu tambahkan Animation - Animator .



Gambar 9.7 Panel Animator

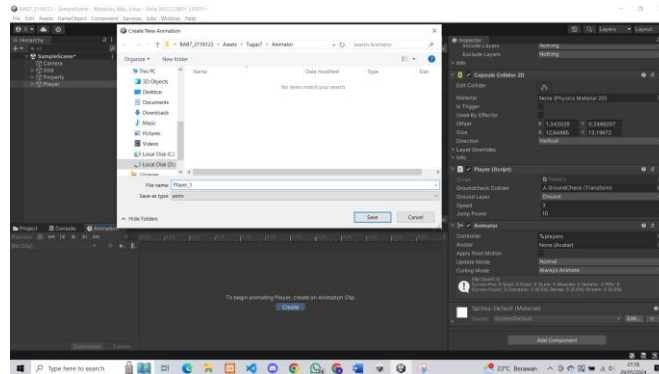
8. Tepatkan Animator disamping Scene dan game ,seperti Tampilan akan berikut ini.



Gambar 9.8 Tampilan Animator

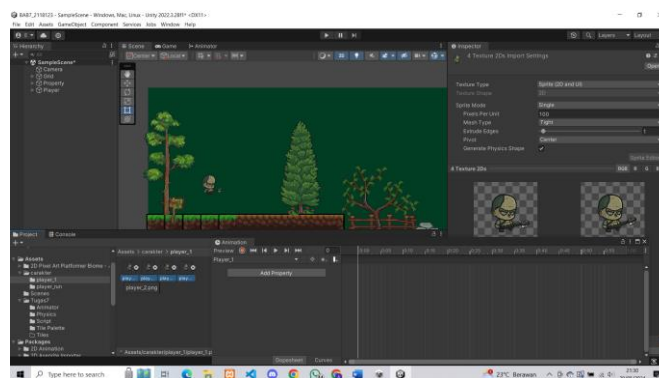


9. Masuk ke hierarchy player , lalu create pada panel Animation ,lalu simpan dengan nama Player\_1.



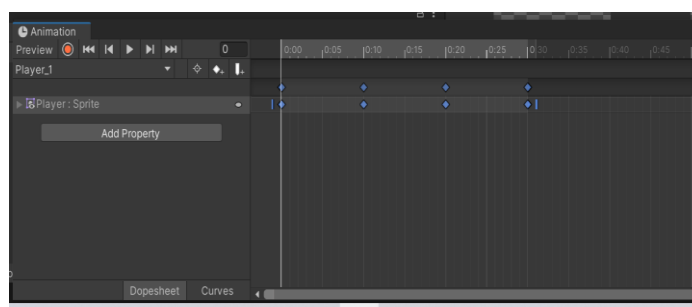
Gambar 9.9 Create Player\_1

10. Masuk dalam karakter , pada folder player\_1 kita seleksi semua karakternya lalu kita drag & drop ke dalam panel Animation.



Gambar 9.10 Karakter Player\_1

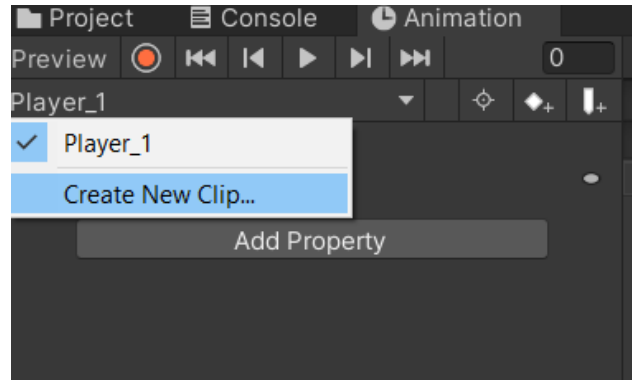
11. Ctrl + A pada player\_1 lalu kita klik Ctrl sambil kita Tarik ke frame 0.30 seperti berikut ini .



Gambar 9.11 Frame 0.30 Palyer\_1

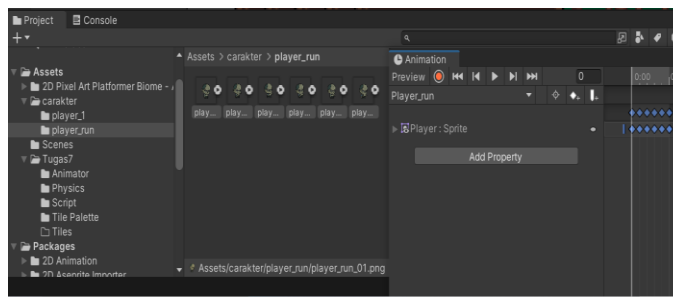


12. Pada Panel Animation kita klik player\_1 ,lalu create New Clip lalu rename menjadi Player\_run , simpan di folder animator.



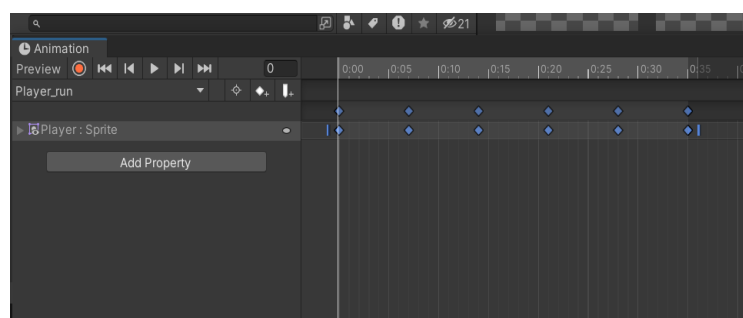
Gambar 9.12 Cretae New Clip

13. Masuk kedalam project ,lalu ke karakter masuk di folder player\_run kita drag & drop ke dalam menu Animation Player\_run.



Gambar 9.13 Player\_run

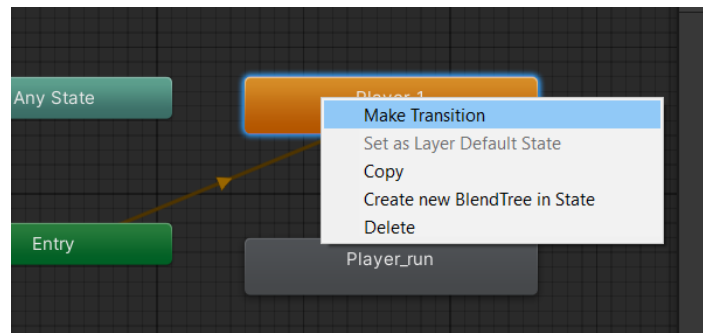
14. Ctrl + A pada Panel Animation , klik Ctrl lalu Tarik ke frame 0.35.



Gambar 9.14 Frame 0.35 Player\_run

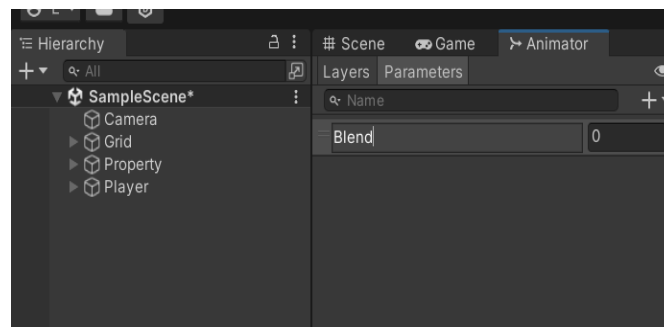


15. Kemudian tambahkan transisi antara player\_1 dan Player\_run , klik kanan pada player\_1 lalu make transition.



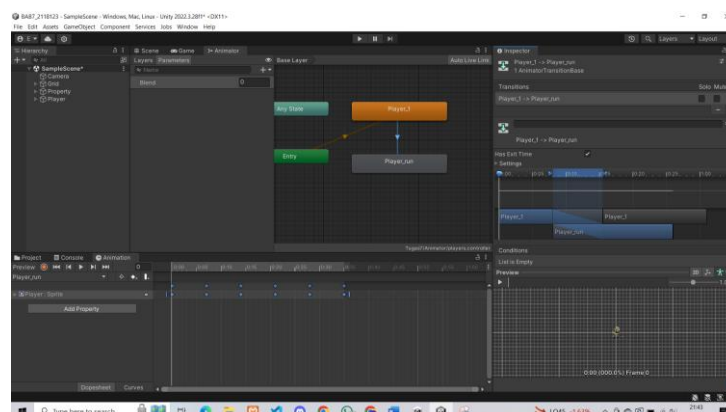
Gambar 9.15 Make Transition

16. Selanjutnya masuk kedalam tab parameter , tambahkan sebuah tipe data Float dan rename menjadi “Blend” .



Gambar 9.16 New Blend

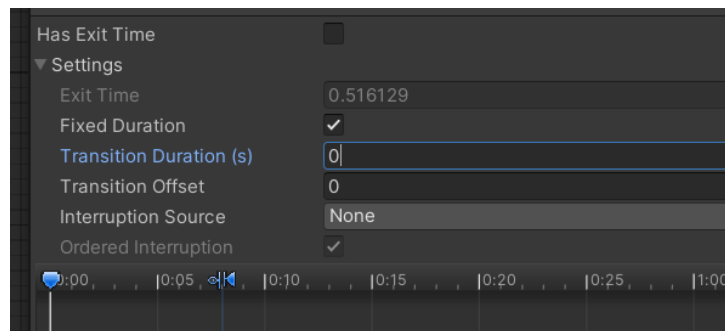
17. Klik 2X kali pada arah panah player\_run , lalu muncul sebuah inspector lalu masuk ke dalam condition lalu tambah kemudian atur menjadi Blend.



Gambar 9.17 Condition Blend

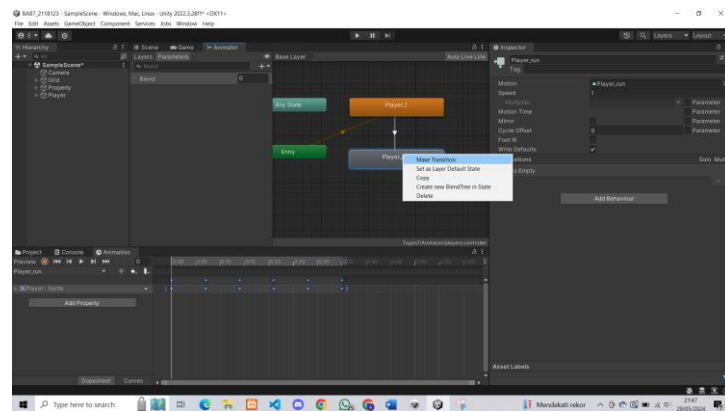


18. Pada bagian setting , kita hilangkan centangnya pada Has Exit Time ,lalu Transition Duration ganti menjadi 0 , seperti gambar berikut.



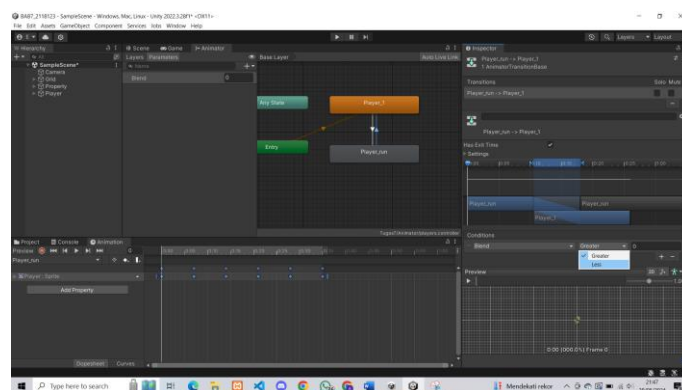
Gambar 9.18 Transition Duration

19. Selanjutnya kita buat transisi sebaliknya yaitu player\_run ke player\_1.



Gambar 9.19 Transisi Player\_run

20. Klik 2X panah arah player\_1 , lalu masuk tambahkan conditions Blend dan ubah operator menjadi Less, seperti berikut.



Gambar 9.20 Inspector Player\_1



21. Tambahkan script seperti gambar berikut di clas player.

```
0 references
public class Player : MonoBehaviour
{
    2 references
    public Animator animator;
    5 references
```

Gambar 9.21 Script Clas Player

22. Tambahkan juga script berikut ke dalam Komponen Animator .

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

Gambar 9.22 Script Animator

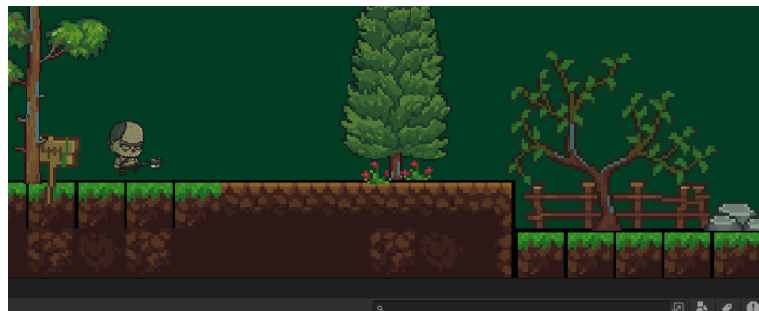
23. Selanjutnya pada fuction fixedUpdate tambahkan juga script berikut.

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

    animator.SetFloat("Blend", Mathf.Abs(rb.velocity.x));
    jump = false; // Reset jump flag after each FixedUpdate
}
```

Gambar 9.23 Script FixedUpdate

24. Coba jalankan gamenya , maka karakter kita akan ada animationnya.

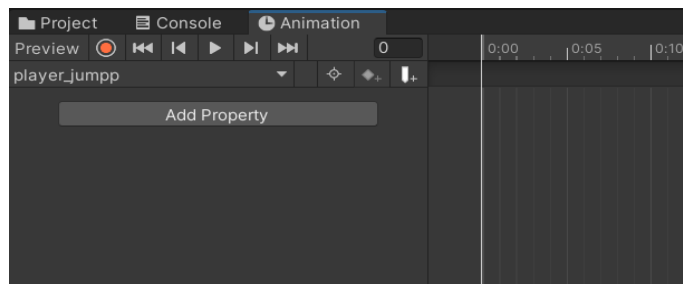


Gambar 9.24 Run Game



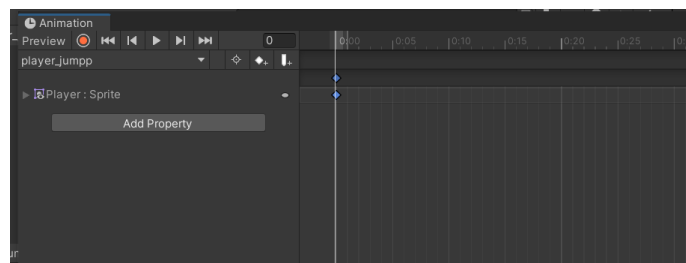


25. Berikutnya kita akan membuat animasi melompatnya , klik pada panel animation lalu create new clip , rename menjadi Player\_Jump.



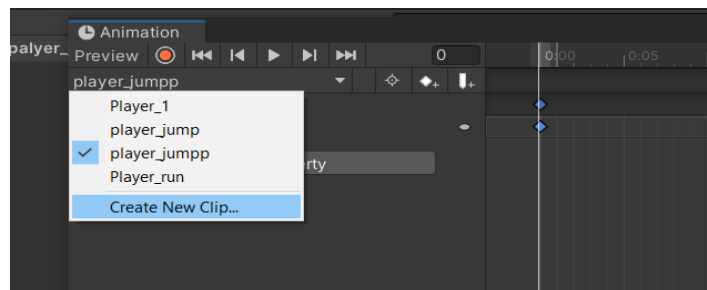
Gambar 9.25 Player\_jumpp

26. Masuk kedalam karakter , lalu cari player jump kita drag and drop kedalam panel animation seperti berikut.



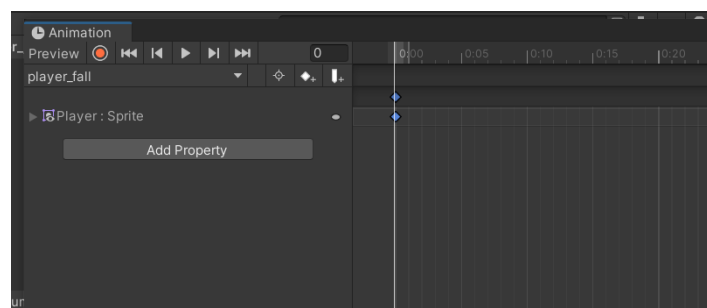
Gambar 9.26 Frame Panel Animation

27. Create new clip lagi lalu beri nama player fall.



Gambar 9.27 Player\_fall

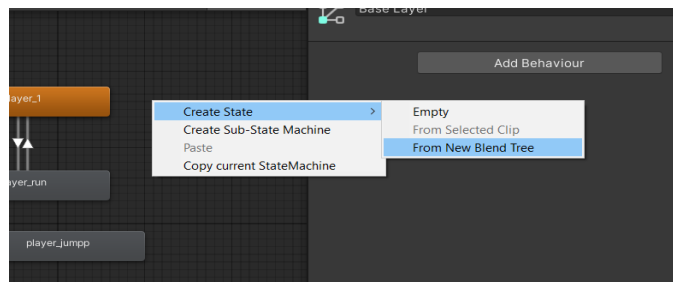
28. Masukkan karakter player fall ke dalam panel animation .



Gambar 9.28 Frame Player Fall

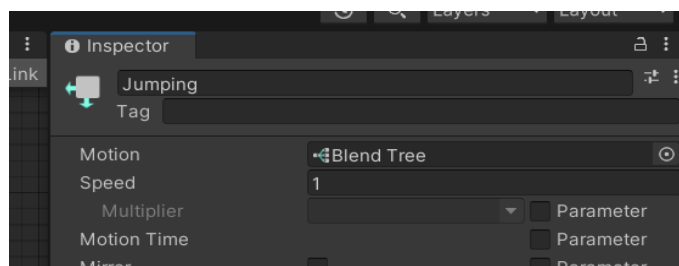


29. Selanjutnya masuk kedalam Animator , klik kanan pilih Create State - From New Blend Tree.



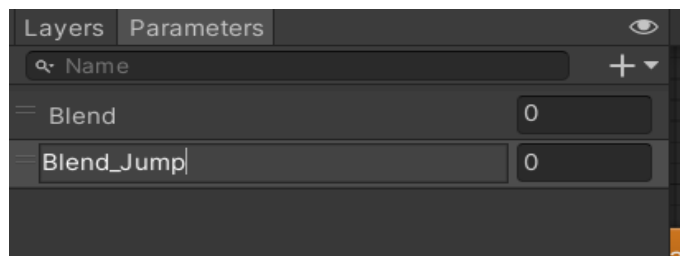
Gambar 9.29 From New Blend tree

30. Masuk ke inspector Blend Tree , ubah Namanya menjadi Jumping



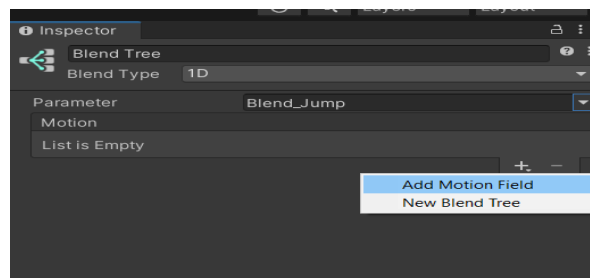
Gambar 9.30 Rename Jumping

31. Masuk di menu parameters tambahkan parameter tipe data float dan rename menjadi “blend Jump”.



Gambar 9.31 Blend\_Jump

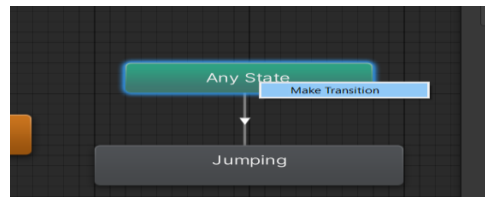
32. Klik 2X pada Blend tree,Tekan pada Blend tree untuk masuk kedalam inspectornya , lalu Add Motion Field ,seperti berikut .



Gambar 9.32 Add Motion Field

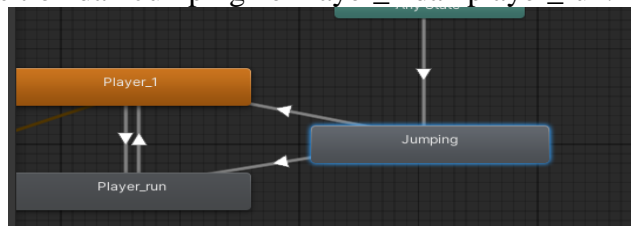


33. Apabila sudah masuk ke dalam base layer lagi , make transition dari Any State ke Jumping.



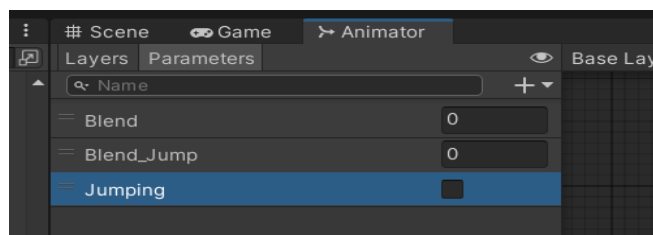
Gambar 9.33 Make Transition Any State

34. Make transition dari Jumping ke Player\_1 dan player\_run.



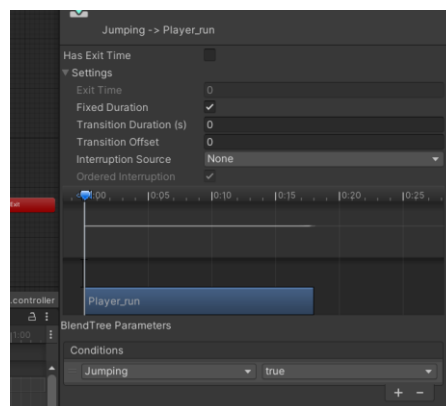
Gambar 9.34 Make Transition Jumping

35. Buat transisi baru di parameter dengan tipe data bool lalu rename menjadi jumping



Gambar 9.35 Animator Jumping

36. Klik arah panah yang mengarah ke player\_1 dan player\_run , lalu setting inspector dan tambahkan conditionsnya jumping dan player\_1 menjadi false dan player\_run menjadi true.



Gambar 9.36 Conditions Jumping



37. Tambahkan script pada void update, seperti berikut.

```
0 references
void Update()
{
    horizontalValue = Input.GetAxisRaw("Horizontal");
    if (Input.GetButtonDown("Jump") || Input.GetKeyDown(KeyCode.W))
    {
        animator.SetBool("Jumping", true);
        jump = true;
    }
}
```

Gambar 9.37 Script Void Update

38. Masukkan juga source code di FixedUpdate.

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

    animator.SetFloat("Blend", Mathf.Abs(rb.velocity.x));
    animator.SetFloat("Blend Jump", rb.velocity.y);
    jump = false; // Reset jump flag after each FixedUpdate
}
```

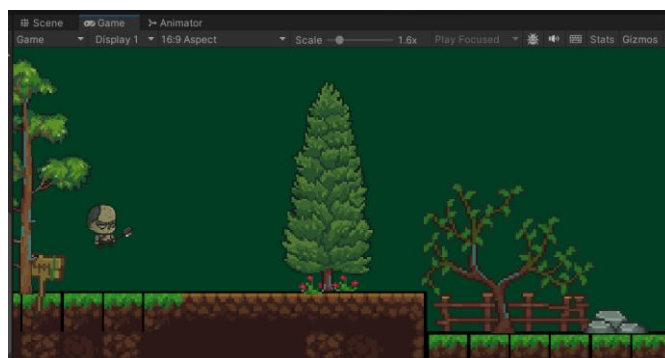
Gambar 9.38 Script FixedUpdate

39. Selanjutnya masuk di dalam method GroundCheck, masukan source code berikut.

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

Gambar 9.39 Script GroundCheck

40. Kita akan mencoba Run Gamenya lalu kita Render File Gamenya.



Gambar 9.40 Run Game Animator



## B. QUIZ :

### Soal Kuis Pertemuan 9

Lengkapi Source code dibawah ini :

```
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 source code diatas ,pertama yaitu metode SetBool dalam HandleJumpInput memerlukan dua parameter: nama parameter animator dan nilai bool (true atau false).kesalahan source code Kedua, dalam HandleMovementInput, logika yang digunakan untuk mengatur animasi dan pergerakan tidak konsisten. Kondisi if (move != 1) memeriksa nilai move yang bisa berkisar antara -1 dan 1, namun pergerakan seharusnya



dikalikan dengan nilai move untuk arah yang benar. Untuk kesalahan Ketiga, logika untuk membalik skala dalam HandleMovementInput memiliki kesalahan karena  $\text{move} \neq 0$  sudah mencakup semua nilai non-nol, sehingga kondisi `else if (move > 0)` tidak pernah terpenuhi. Selain itu, mengubah skala karakter seharusnya hanya membalik sumbu x tanpa mengubah nilai lain, misalnya `new Vector3(-1, 1, 1)` atau `new Vector3(1, 1, 1)`.

### **C. Link Github Pengumpulan**

[https://github.com/Eri-Kusyairi/2118123\\_PRAK\\_ANIGAME](https://github.com/Eri-Kusyairi/2118123_PRAK_ANIGAME)