



## TUGAS PERTEMUAN: 9

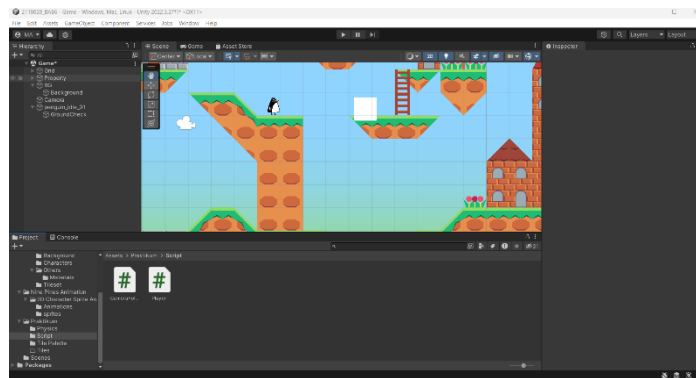
### GAME ANIMATION

|             |   |                                |
|-------------|---|--------------------------------|
| NIM         | : | 2118029                        |
| Nama        | : | Mohammad Arvin Manda Aradhana  |
| Kelas       | : | A                              |
| Asisten Lab | : | APRILLIA DWI DYAH S. (2118143) |

#### 1.1 Tugas 1 : Membuat Game Animation

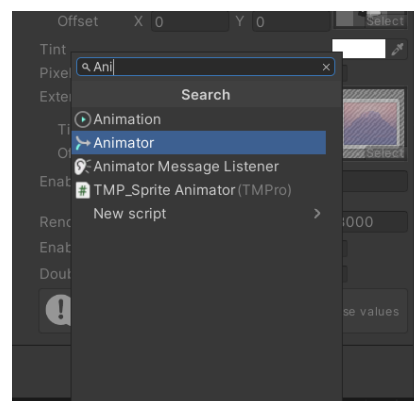
##### A. Mempersiapkan Project

1. Buka proyek Unity yang sudah memiliki kamera dan gerakan karakter yang ditambahkan sebelumnya.



Gambar 9.1 Tampilan project sebelumnya

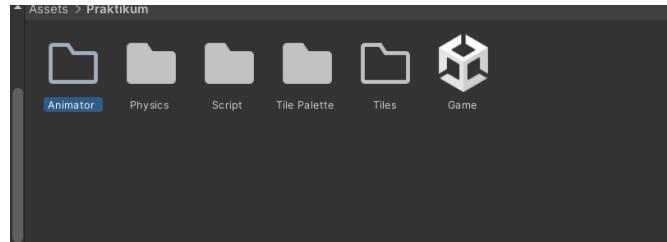
2. Di tab Inspector pada hierarki karakter, tambahkan komponen Animator.



Gambar 9.2 Menambah Komponen Animator

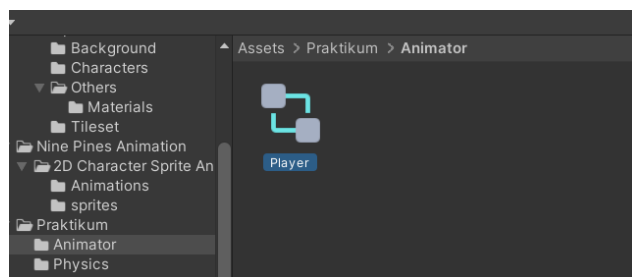


3. Selanjutnya, di dalam folder Praktikum, buatlah folder baru dengan nama Animator.



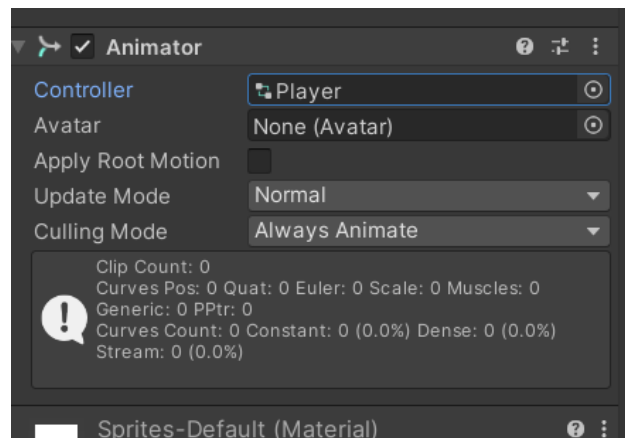
Gambar 9.3 Membuat Folder Animator

4. Setelah itu, buat file Animator Controller di dalam folder Animator, lalu ubah namanya menjadi Player.



Gambar 9.4 Membuat File Animator Controller

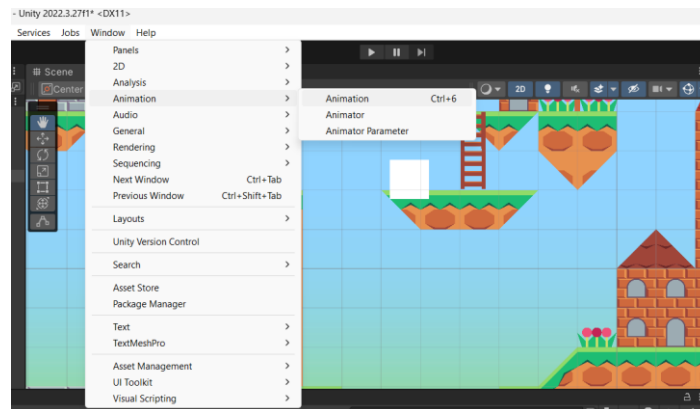
5. Kemudian, ubah pengaturan Controller di komponen Animator pada hierarki player menjadi Player.



Gambar 9.5 Mengubah Controller menjadi player

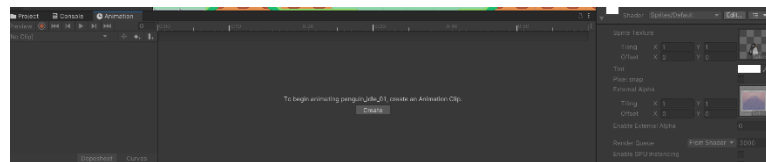


6. Tambahkan panel Animation melalui menu Window dengan memilih Animation, lalu pilih Animation atau tekan Ctrl+6.



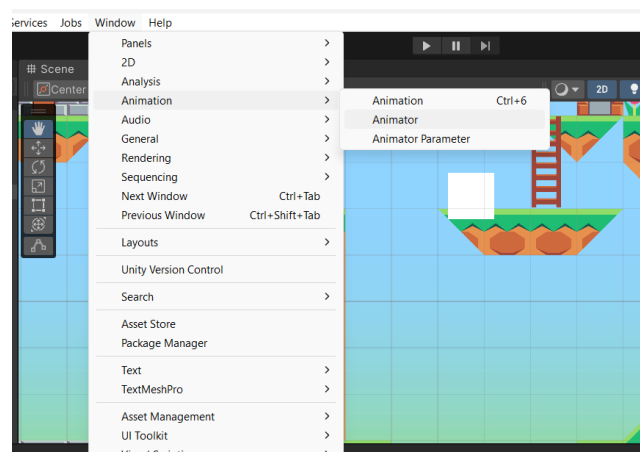
Gambar 9.6 Menambahkan Panel Animation

7. Setelah itu, pindahkan panel yang baru muncul ke bagian bawah, dan pisahkan panel Project dengan panel Animation.



Gambar 9.7 Masuk ke bagian *asset store*

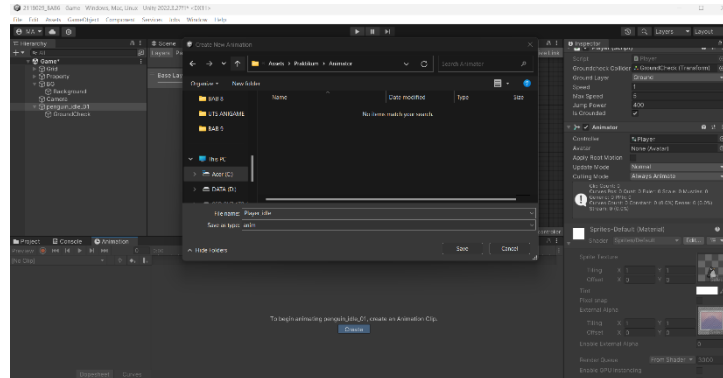
8. Tambahkan panel Animator.



Gambar 9.8 Menambah Panel Animator

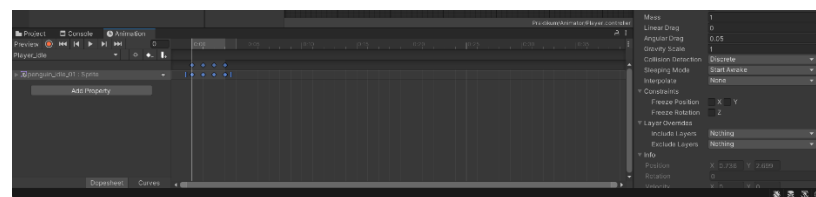


9. Pilih penguinidle\_1 di hierarki, lalu di panel Animation, klik Create dan simpan di dalam folder Animator dengan nama 'Player\_idle' untuk membuat animasi.



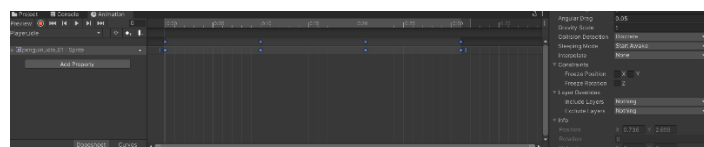
Gambar 9.9 Membuat Animasi

10. Di folder Project, buka folder Main Character, kemudian pilih folder 2 dan pilih gambar ' penguinidle \_0' hingga 'penguinidle\_10'. Setelah itu, seret gambar-gambar tersebut ke tab Animation.



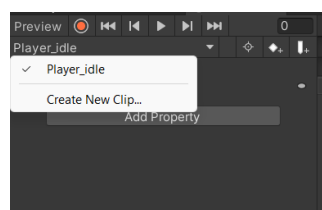
Gambar 9.10 Menambahkan Animasi Idle

11. Tekan Ctrl+A di panel Animation, lalu seret kotak kecil pada timeline hingga mencapai frame 0:33.



Gambar 9.11 Mengubah Timeline

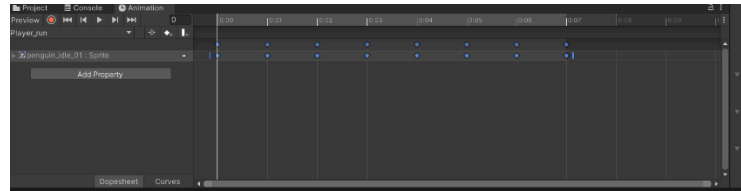
12. Buat animasi baru dengan mengklik Player\_idle, kemudian pilih Create New Clip dan beri nama Player\_run, lalu simpan di dalam folder Animator.



Gambar 9.12 Membuat Animasi Baru

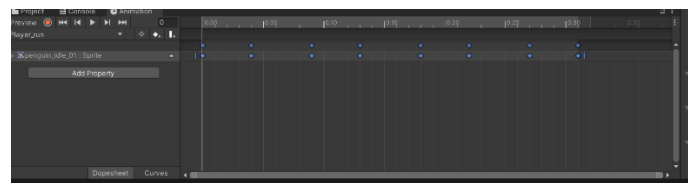


13. Di folder Project, buka folder Main Character, kemudian pilih folder 2 dan pilih gambar 'Run\_0' hingga 'Run\_10'. Setelah itu, seret gambar-gambar tersebut ke tab Animation.



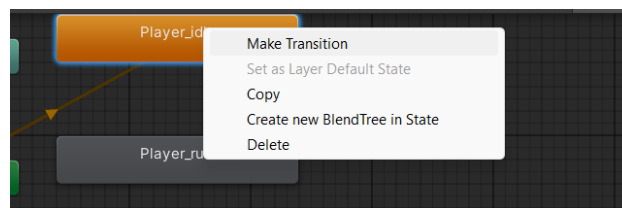
Gambar 9.13 Menambahkan Animasi Run

14. Tekan Ctrl+A di panel Animation, lalu seret kotak kecil pada timeline hingga mencapai frame 0:30.



Gambar 9.14 Mengubah Timeline

15. Buat transisi antara Player\_idle dan Player\_run pada tab Animator dengan mengklik kanan Player\_idle, lalu pilih Make Transition, dan seret ke Player\_run.



Gambar 9.15 Membuat Transisi

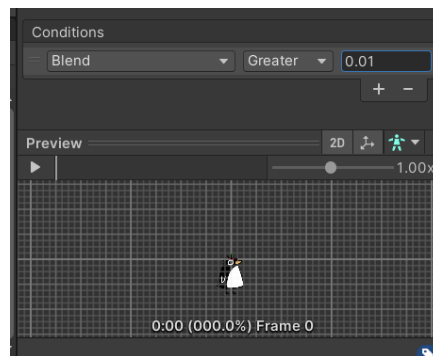
16. Beralih ke tab parameter, kemudian tambahkan tipe data float dengan menekan ikon tambah, dan ubah namanya menjadi Blend.



Gambar 9.16 Membuat tipe data baru

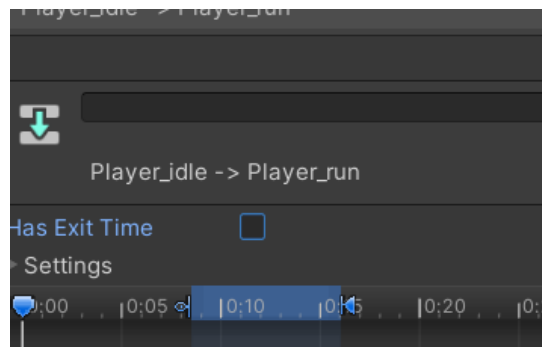


17. Klik panah putih tersebut, lalu pada bagian Condition tambahkan tipe data Blend dengan nilai kondisi 0.01.



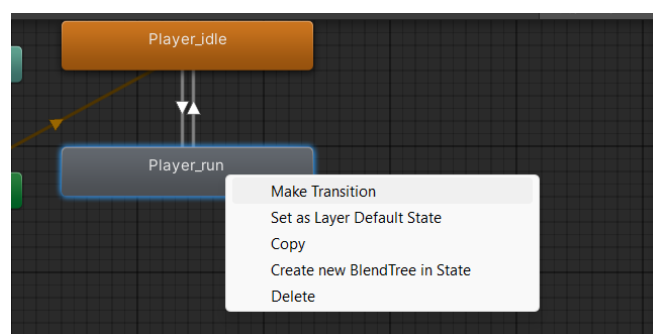
Gambar 9.17 Memberikan nilai kondisi

18. Nonaktifkan centang pada pengaturan Has Exit Time, dan atur nilai Transition Duration menjadi 0 pada bagian Settings.



Gambar 9.18 Mengubah Pengaturan Transisi

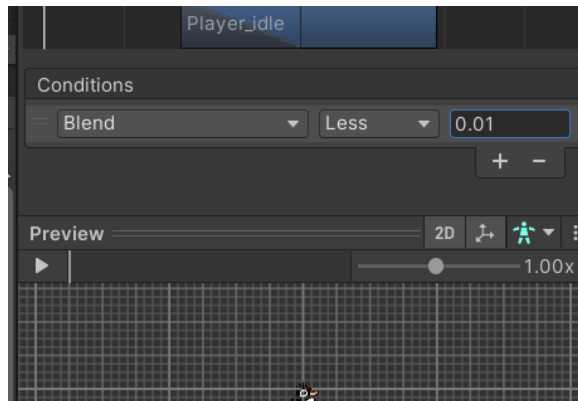
19. Tambahkan juga transisi dari Player\_run ke Player\_idle dengan mengklik kanan pada Player\_run, kemudian pilih Make Transition.



Gambar 9.19 Menambahkan Transisi

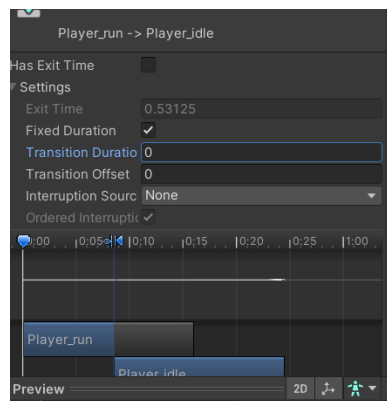


20. Lalu, pada transisi yang baru saja dibuat, tambahkan tipe data Blend pada bagian Condition dengan nilai Less dari 0.01.



Gambar 9.20 Memberikan Tipe Data

21. Nonaktifkan centang pada opsi Has Exit Time dan atur nilai Transition Duration menjadi 0 pada bagian Settings.



Gambar 9.21 Mengubah Pengaturan Transisi

22. Untuk menyesuaikan animasi, buka script Player, kemudian tambahkan baris kode berikut di awal kelas Player.

```
public Animator animator;
```

23. Ubah fungsi Awake untuk menambahkan script komponen Animator.

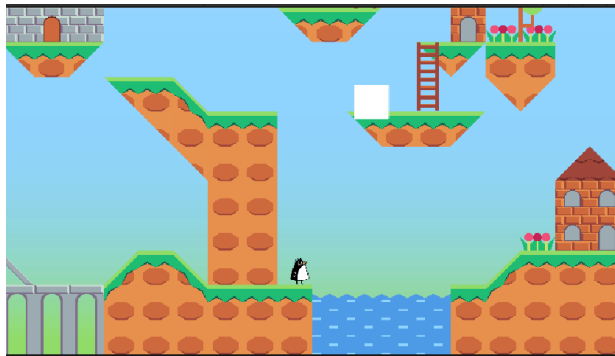
```
private void Awake() {  
    rb = GetComponent();  
    animator = GetComponent();  
}
```

24. Ubah fungsi FixedUpdate seperti berikut untuk memicu transisi berjalan.

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

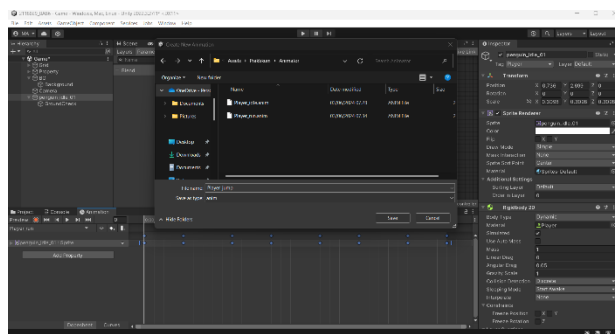


25. Setelah itu, coba jalankan, dan player akan memiliki animasi saat berhenti maupun berjalan.



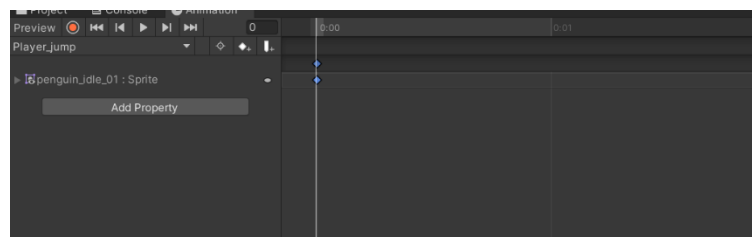
Gambar 9.22 Menjalankan Game

26. Buat animasi baru dengan mengklik Player\_run, lalu pilih Create New Clip, dan beri nama Player\_jump, kemudian simpan di dalam folder Animator.



Gambar 9.23 Membuat Animasi Baru

27. Di folder Project, buka folder character, lalu pilih folder 2 dan pilih gambar 'Jump', kemudian seret ke tab Animation.

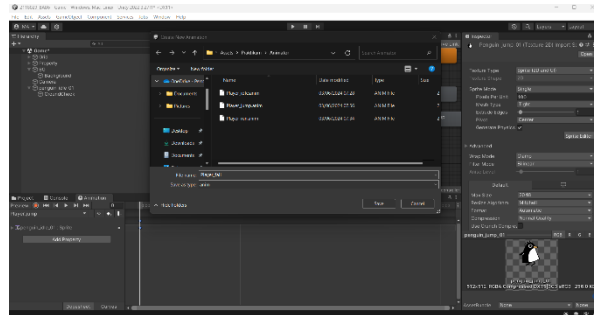


Gambar 9.24 Menambahkan Animasi Jump



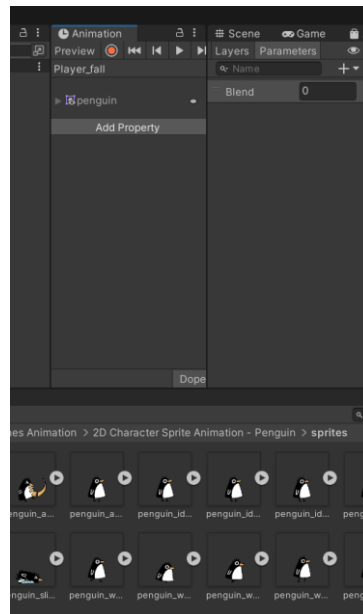


28. Buat animasi baru dengan mengklik Player\_jump, kemudian pilih Create New Clip, dan beri nama Player\_fall, kemudian simpan di dalam folder Animator.



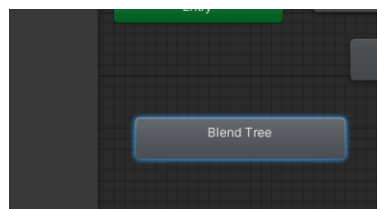
Gambar 9.25 Membuat Animasi Baru

29. Di folder Project, buka folder Character, lalu pilih folder 2 dan pilih gambar 'Fall', kemudian seret ke tab Animation.



Gambar 9.26 Menambahkan Animasi Fall

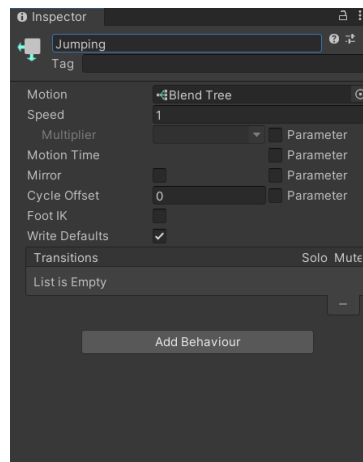
30. Selanjutnya, klik kanan di area kosong pada menu Animator, pilih Create State, dan kemudian From New Blend Tree untuk menambahkan animasi ketika lompat.



Gambar 9.27 Membuat Blend Tree Baru

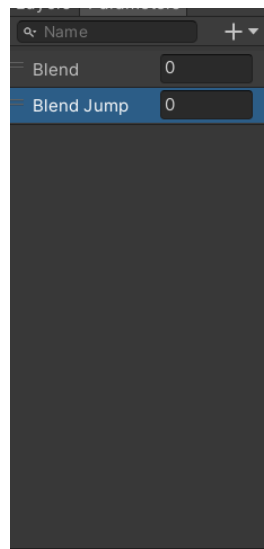


31. Buka menu Inspector pada Blend Tree, dan ganti namanya menjadi Jumping.



Gambar 9.28 Mengubah Nama Blend Tree

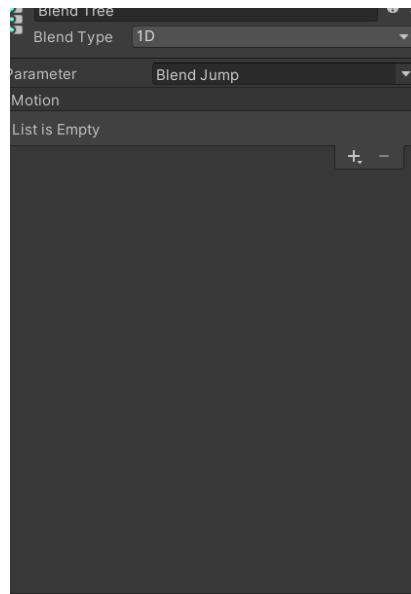
32. Di tab parameter, tambahkan tipe data float dengan menekan ikon tambah, dan ubah namanya menjadi Blend Jump.



Gambar 9.29 Membuat Tipe Data Baru

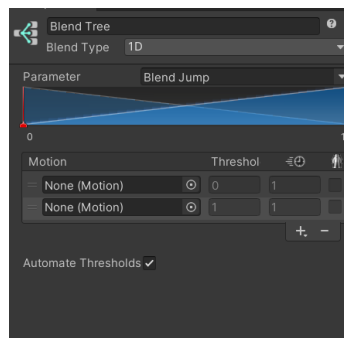


33. Klik dua kali pada Blend Tree Jumping, lalu klik Blend Tree untuk mengubah parameter menjadi Blend Jump di menu Inspector.



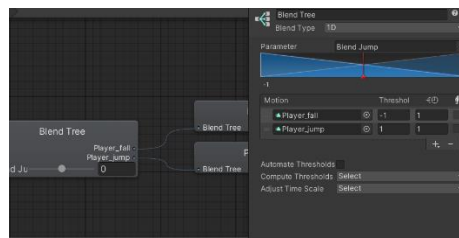
Gambar 9.30 Megubah Parameter

34. Tambahkan dua Motion Field dengan menekan ikon +, lalu pilih Add Motion Field.



Gambar 9.31 Membuat Motion Field

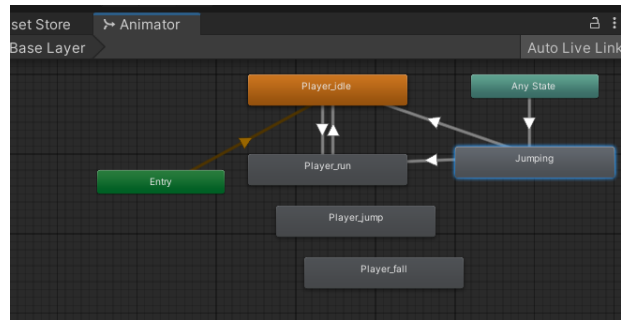
35. Klik icon pada bagian None (Motion), lalu sesuaikan pengaturannya seperti gambar berikut.



Gambar 9.32 Mengatur Motion Field

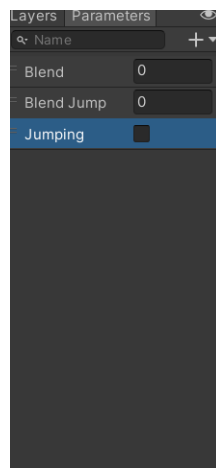


36. Kembali ke Base Layer, lalu buat transisi seperti pada gambar berikut.



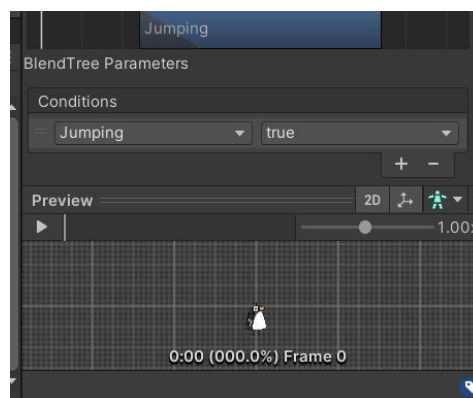
Gambar 9.33 Membuat Transisi

37. Masuk ke tab parameter, lalu tambahkan tipe data bool dengan cara menekan icon tambah dan ubah nama menjadi Jumping.



Gambar 9.34 Membuat Tipe Data Baru

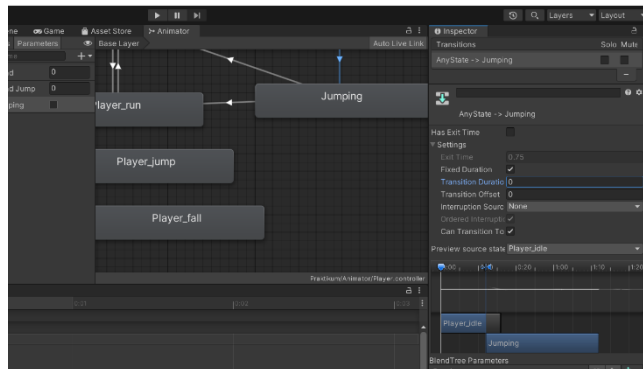
38. Klik panah yang mengarah ke Jumping, lalu tambahkan kondisi Jumping dan ubah nilainya menjadi true di menu Inspector.



Gambar 9.35 Memberikan Tipe Data

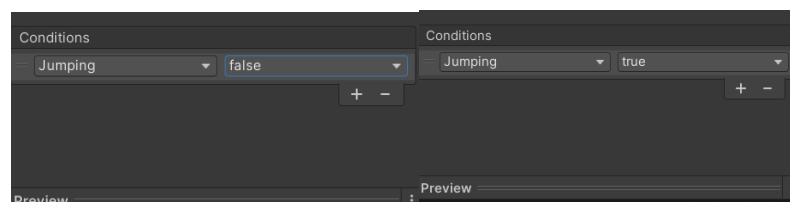


39. Nonaktifkan centang pada opsi Has Exit Time, dan atur nilai Transition Duration menjadi 0 di bagian Settings.



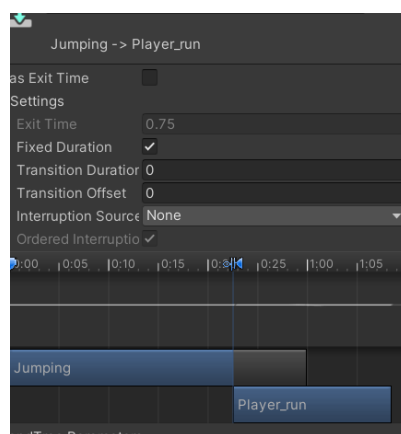
Gambar 9.36 Mengubah Pengaturan Transisi

40. Klik panah yang mengarah ke Player\_idle dan Player\_run, lalu tambahkan kondisi Jumping. Pada panah ke Player\_idle, ubah nilainya menjadi false, dan pada panah ke Player\_run, ubah nilainya menjadi true.



Gambar 9.37 Memberikan Tipe Data

41. Nonaktifkan centang pada opsi Has Exit Time, dan atur nilai Transition Duration menjadi 0 di bagian Settings pada kedua arah panah tersebut.



Gambar 9.38 Mengubah Pengaturan Transisi



42. Buka script Player, 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;
}
```

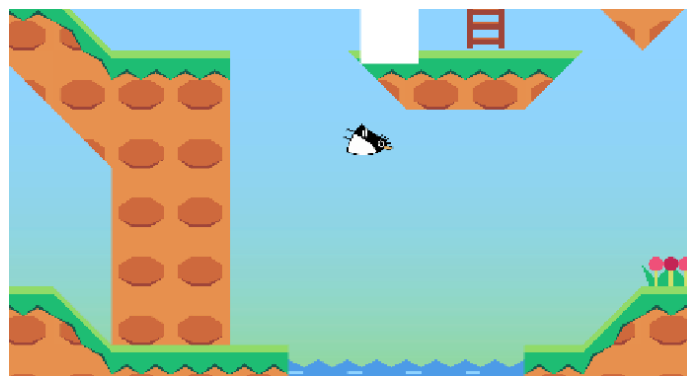
43. Ubah fungsi FixedUpdate seperti berikut.

```
void FixedUpdate()
{
    GroundCheck();
    Move(horizontalValue, jump);
    animator.SetFloat("Blend",
    Mathf.Abs(rb.velocity.x));
    animator.SetFloat("Blend Jump", rb.velocity.y);
    jump = false; // Reset flag jump setelah
    FixedUpdate
}
```

44. Ubah fungsi GroundCheck seperti berikut.

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

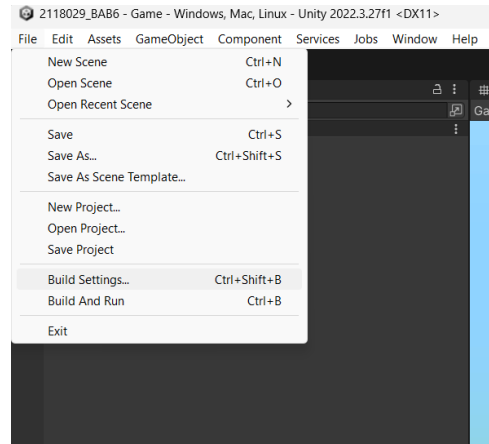
45. Kemudian jika di-play maka karakter sudah menggunakan animasi.



Gambar 9.39 Menjalankan Game



46. Lalu untuk merender game, pilih Build Settings pada menu File.



Gambar 9.40 Build settings

## B. Kuis : Memperbaiki 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);
    }
}
```



Penjelasan:

Pada fungsi `HandleJumpInput()`, diperlukan perbaikan saat menetapkan nilai `isJumping` ke `true` ketika tombol spasi ditekan menggunakan `Input.GetKeyDown`. Selain itu, perbaikan diperlukan saat mengatur `isJumping` menjadi `false` ketika tombol spasi dilepas menggunakan `Input.GetKeyUp`. Dalam fungsi `HandleMovementInput()`, terdapat kebutuhan untuk memperbaiki logika saat nilai `move` sama dengan 0. Ketika `move` adalah 0, karakter harus dalam keadaan diam (`isIdle` true dan `isWalking` false), sementara jika `move` tidak sama dengan 0, karakter harus dalam keadaan berjalan (`isIdle` false dan `isWalking` true). Selain itu, skala karakter perlu diatur agar ukurannya tetap sama saat bergerak ke kiri atau ke kanan, untuk menjaga proporsionalitas.