



TUGAS PERTEMUAN: 9

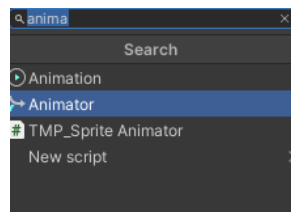
GAME ANIMATION

NIM	:	2118054
Nama	:	M. Farish Ramadhan Ferdinansyah
Kelas	:	B
Asisten Lab	:	Difa Fisabilillah (2118052)

9.1 Tugas 1 : Membuat Character Animation

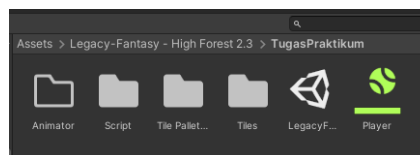
A. Character Animation

1. Pada karakter klik *Inspector* kemudian pilih *Add Component Animator*.



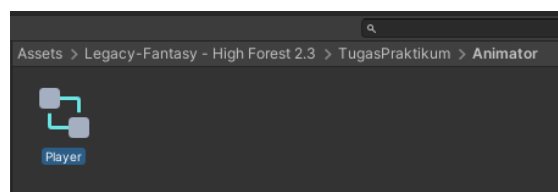
Gambar 9.1 Menambahkan Komponen Animator

2. Pada folder TugasPraktikum buat folder baru bernama “Animator”.



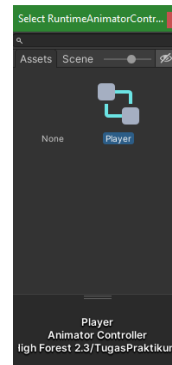
Gambar 9.2 Membuat Folder Animator

3. Buat file *Animator Controller* pada folder Animator , ubah namanya menjadi Player.



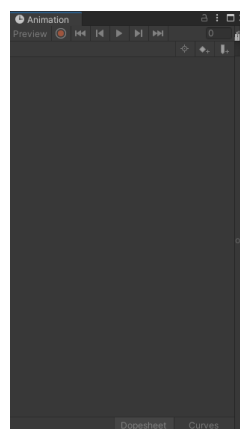
Gambar 9.3 Membuat *Animator Controller*

4. Pada *Hierarchy* klik Player, kemudian cari *Component Animator* di *Inspector*, pada *setting Controller* ubah menjadi Player.



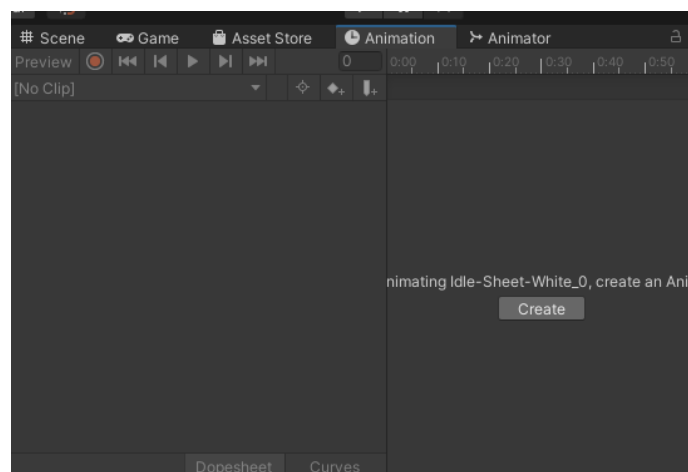
Gambar 9.4 *Setting Controller Player*

5. Tambahkan menu panel *Animation* di menu *Window*, pilih *Animation > Animation* atau tekan Shortcut CTRL + 6.



Gambar 9.5 Menambahkan Panel *Animation*

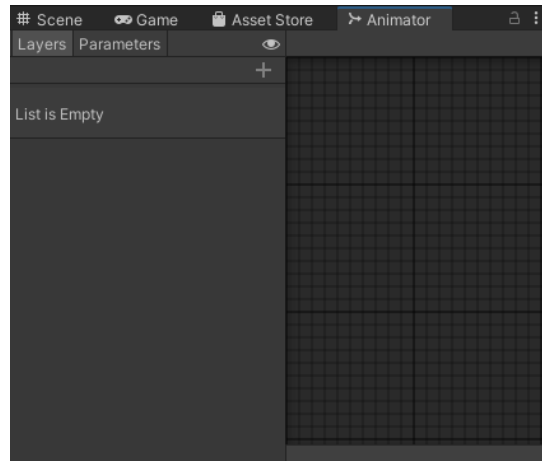
6. Akan muncul menu panel baru, geser panel tersebut dibawah sendiri seperti gambar dibawah ini, *Split Panel Project* dan *Animation*.



Gambar 9.6 *Split Panel Project* dan *Animation*

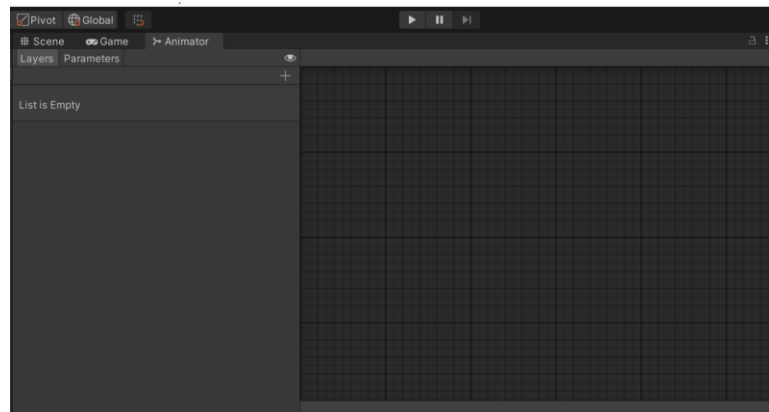


7. Tambahkan menu panel *Animator*.



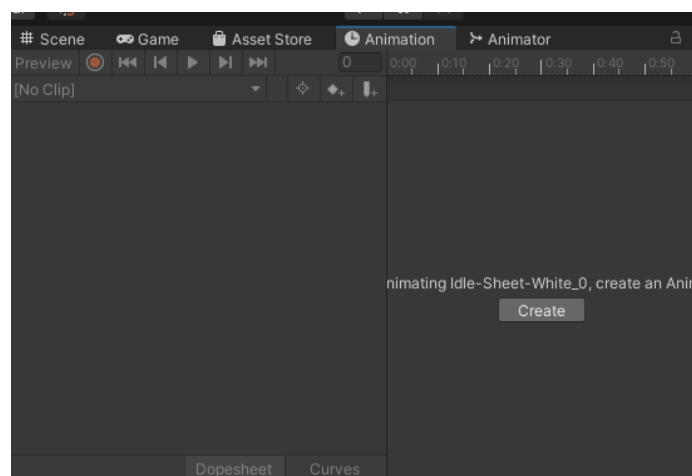
Gambar 9.7 Menambahkan Panel *Animator*

8. Geser panel sesuai gambar berikut.



Gambar 9.8 Menyesuaikan Panel *Animator*

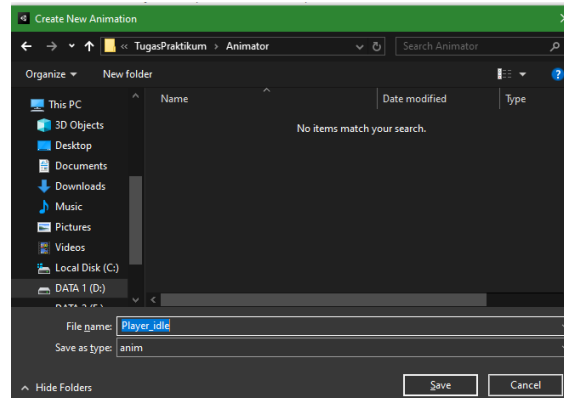
9. Untuk membuat animasi klik Idle-Sheet_0 pada *Hierarchy*, kemudian ke menu panel *Animation*, pilih *Create*.



Gambar 9.9 Membuat Animasi

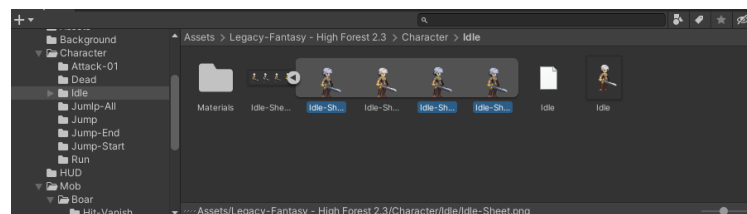


10. Simpan pada folder *Animator* dan beri nama “Player_idle”.



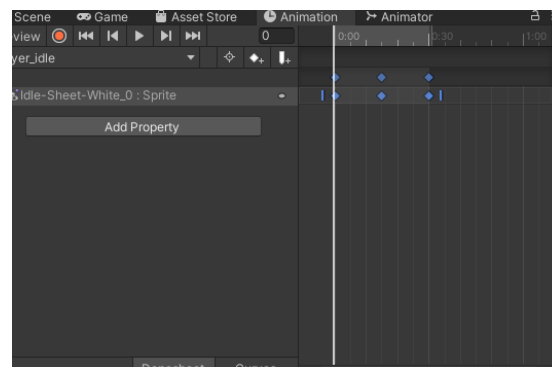
Gambar 9.10 Menyimpan File Player_idle

11. Pada menu Project buka folder Warrior lalu pilih Blue dan pilih gambar idle-Sheet_1 sampai idle-Sheet_4, kemudian *drag* ke tab *Animation*.



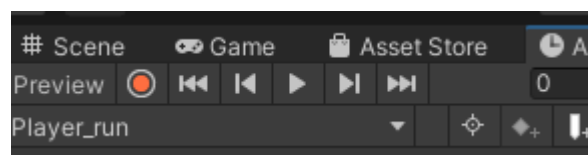
Gambar 9.11 Drag Gambar Player Idle

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



Gambar 9.12 Geser Keyframe Player_idle

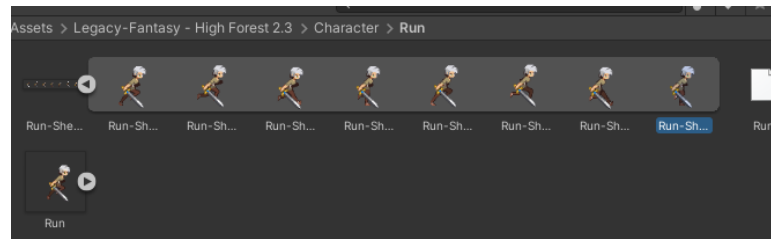
13. Buat animasi baru, klik pada “Player_idle” kemudian pilih *Create New Clip*, dan beri nama “Player_run”, Simpan pada folder *Animator*.



Gambar 9.13 Membuat Animasi Player_run

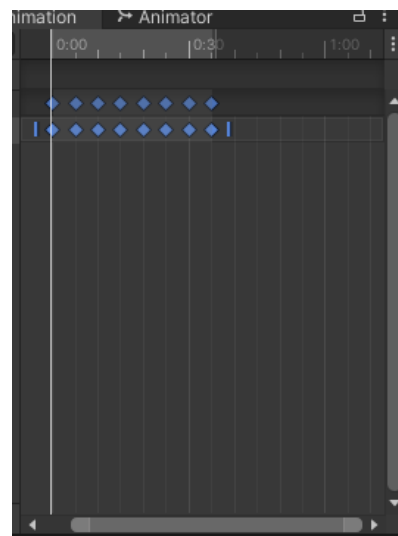


14. Buka menu Project kemudian cari folder Character > Run, Pilih Run-Sheet_0 sampai Run-Sheet_9. *drag and drop* pada menu *Animation*.



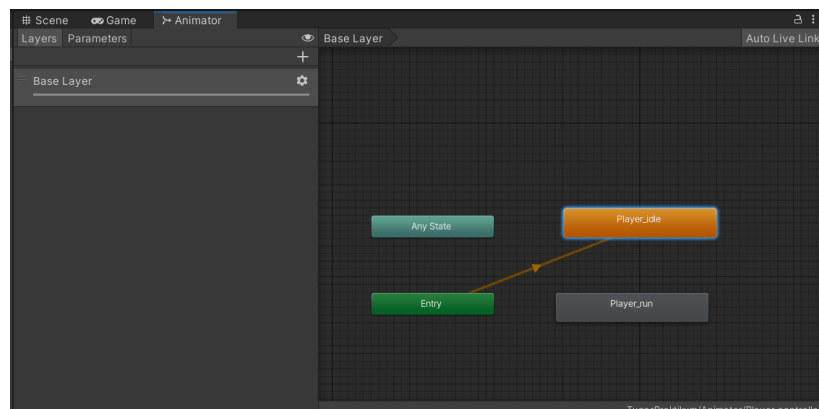
Gambar 9.14 Drag Gambar *Player Run*

15. Pada panel *timeline* tekan Ctrl+A di keyboard, klik bagian kotak kecil disamping *keyframe* terakhir dan geser sampai waktu 0:35.



Gambar 9.15 Geser *Keyframe* *Player_run*

16. Pergi ke menu *Animator* yang telah dibuat sebelumnya dan akan tampil seperti berikut.



Gambar 9.16 Tampilan Menu Animator

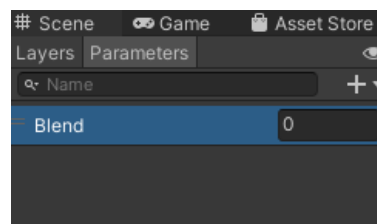


17. Kemudian buat transisi antara `Player_idle` dan `Player_run` dengan cara klik kanan pada `Player_idle` dan pilih *Make Transition* dan tarik ke `Player_run`.



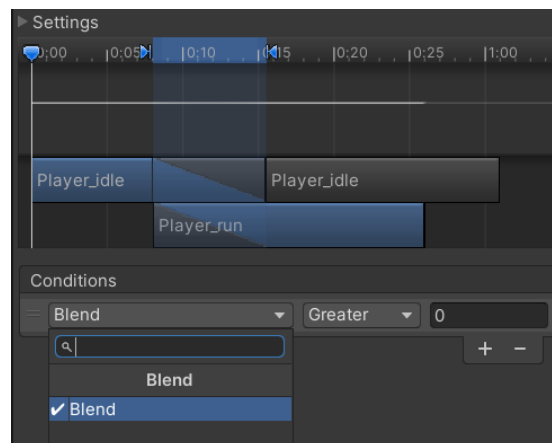
Gambar 9.17 Membuat Transisi

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



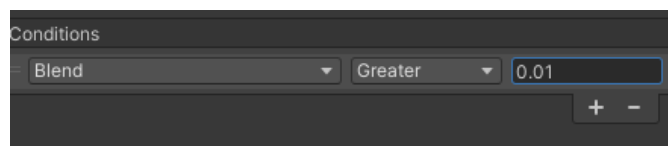
Gambar 9.18 Menambahkan Tipe Data Float

19. Klik panah putih tersebut, pada bagian *Conditions* klik icon tambah kemudian atur menjadi “*Blend*”.



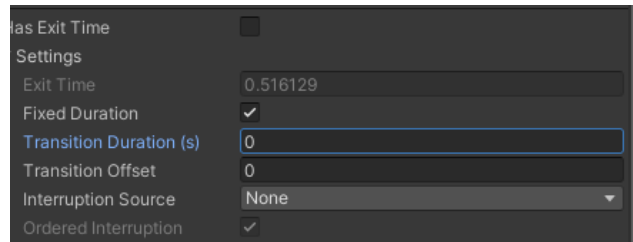
Gambar 9.19 Tambah Parameter Blend

20. Atur nilai *Conditions* blend tersebut menjadi 0.01.



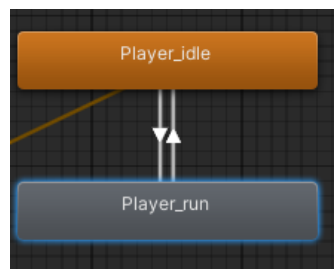
Gambar 9.20 Mengatur Nilai *Conditions* Blend

21. Pada bagian *Settings*, hilangkan centang pada *Has Exit Time* dan atur nilai *Transition Duration* menjadi 0.



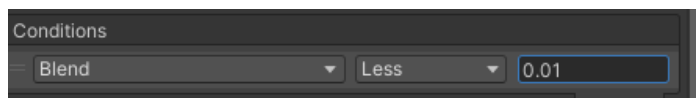
Gambar 9.21 Tampilan *Setting* Animasi

22. Buat transisi juga dari player_run ke player_idle dengan cara klik kanan pada Player_run dan pilih *Make Transition*.



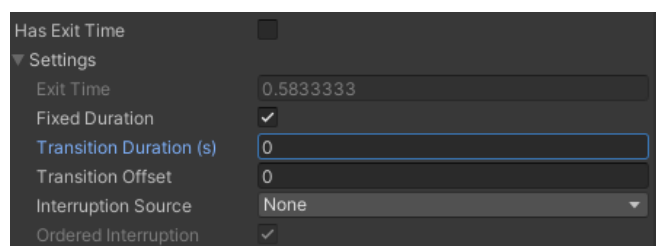
Gambar 9.22 Membuat Transisi

23. Klik panah yang mengarah dari Player_run ke Player_idle, lalu pada bagian *Conditions*, tambahkan parameter Blend, ubah operator dari *Great* menjadi *Less* dan ubah nilai menjadi 0.01



Gambar 9.23 Menambahkan Parameter Transisi

24. Pada bagian *Settings*, hilangkan centang pada *Has Exit Time* dan atur nilai *Transition Duration* menjadi 0.



Gambar 9.24 Tampilan *Setting* Animasi

25. Agar animasi dapat sesuai ketika berjalan, buka *script* PlayerMovement dan tambahkan *source code* berikut pada *class* PlayerMovement.

```
public Animator animator;
```

```
public class Player : MonoBehaviour  
{  
    public Animator animator;
```



Gambar 9.25 Menambahkan Kode pada *Class* PlayerMovement

26. Tambahkan *script* komponen animator pada void Awake seperti dibawah.

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

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

Gambar 9.26 Menambahkan Kode pada Fungsi Awake

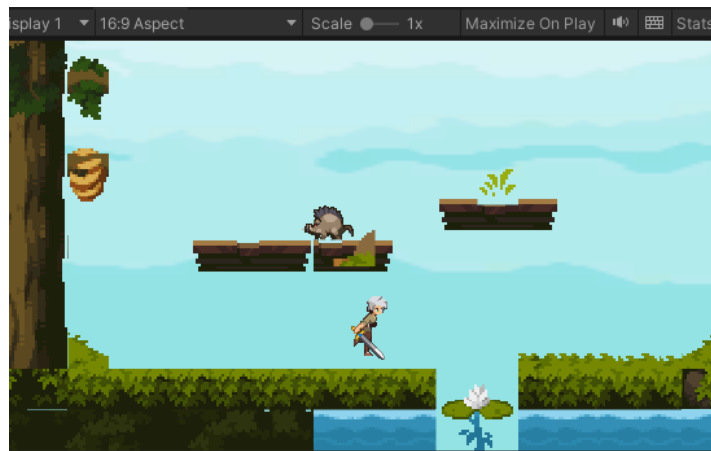
27. Dan pada fungsi FixedUpdate tambahkan *source code* berikut.

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

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

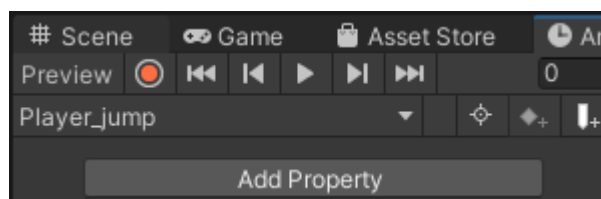
Gambar 9.27 Menambahkan Kode pada Fungsi FixedUpdate

28. Jika dijalankan maka player dapat memiliki animasi ketika berhenti ataupun ketika berjalan.



Gambar 9.28 Tampilan Hasil Animasi

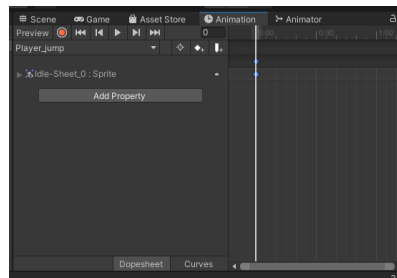
29. Buat animasi baru, klik pada “Player_idle” kemudian pilih *Create New Clip*, dan beri nama “Player_jump”, Simpan pada folder *Animator*.



Gambar 9.29 Membuat Animasi Player_jump

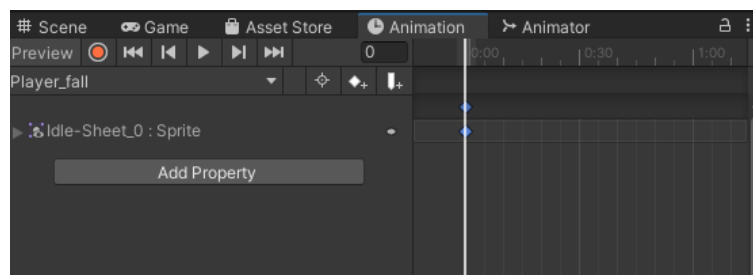


30. Pada folder player buka jump lalu pilih gambar player-jump-1, kemudian drag ke tab Animation.



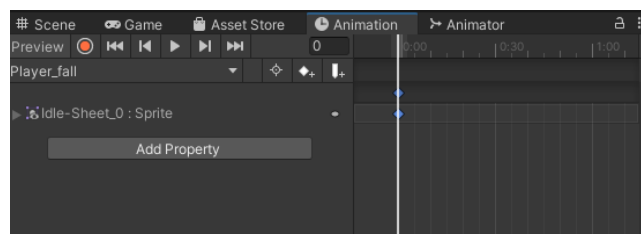
Gambar 9.30 Drag Gambar Player Jump

31. Buat animasi baru dengan cara tekan tulisan “Player_jump” kemudian pilih Create New Clip, dan beri nama “Player_fall”



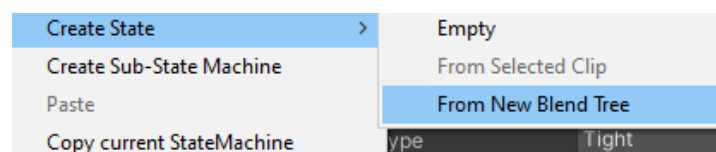
Gambar 9.31 Membuat Animasi Player_fall

32. Pada tab Project buka folder karakter lalu pilih Idle dan pilih gambar player-fall, kemudian drag ke tab Animation..



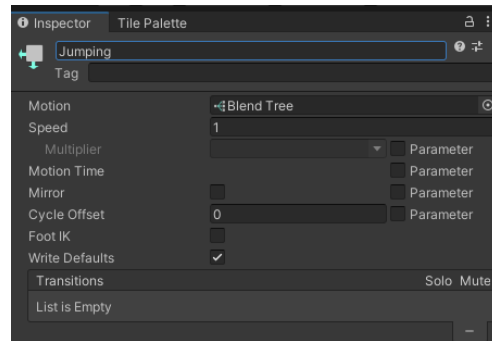
Gambar 9.32 Drag Gambar Player Fall

33. Kemudian untuk menambahkan animasi ketika melompat. Klik kanan pada menu Animator, di area kosong , pilih Create State>From New Blend Tree.



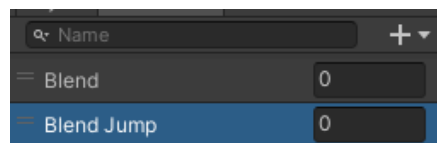
Gambar 9.33 Membuat New Blend Tree

34. Pada Animator klik Blend Tree, di menu Inspector, ubah namanya menjadi Jumping”.



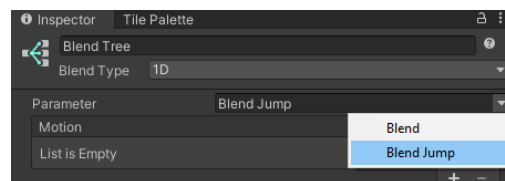
Gambar 9.34 Menambahkan Tipe Data Float

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



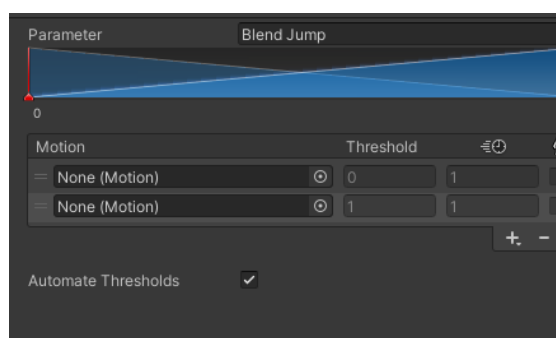
Gambar 9.35 Tambah Parameter Blend Jump

36. Klik 2X Blend Tree “Jumping”, pada inspector ubah parameter menjadi “Blend Jump”.



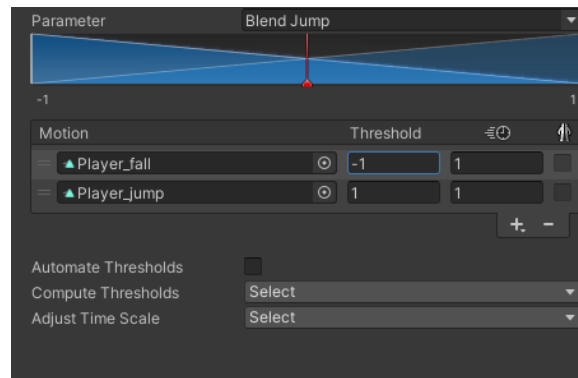
Gambar 9.36 Mengatur Nilai *Conditions* Blend Jump

37. Buka menu Inspector, tekan icon + dan pilih Add Motion Field. Tambahkan dua Motion Field.



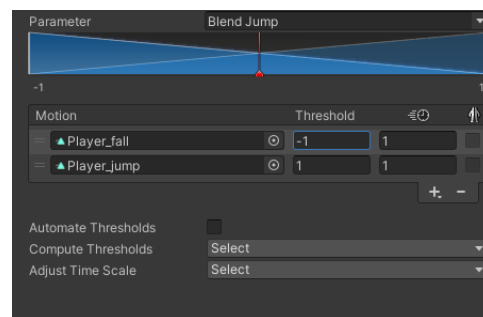
Gambar 9.37 Tampilan *Setting* Animasi

38. Klik bagian icon None (Motion), maka akan muncul Windows Motion, Tambahkan Sesuai dengan urutan.



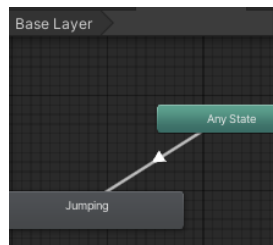
Gambar 9.38 Membuat *Motion*

39. Hilangkan centang “Automate Thresholds” dan atur nilai Threshold seperti berikut.



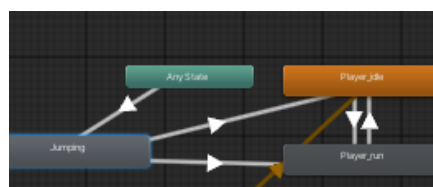
Gambar 9.39 Mengatur nilai *Threshold*

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



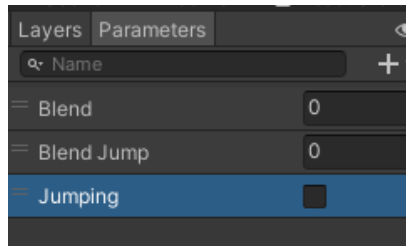
Gambar 9.40 Mengatur *Any State*

41. Klik kanan Jumping, pilih Make Transition dan arahkan panahnya ke Player_idle dan Player_run.



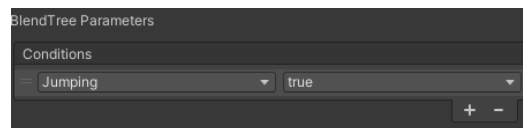
Gambar 9.41 Tampilan *Make Transition*

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



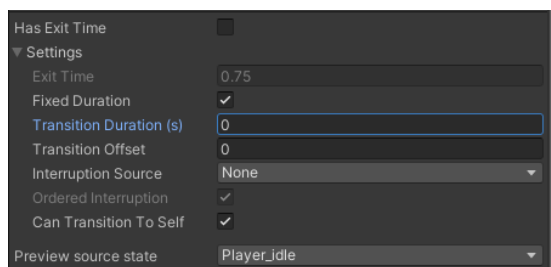
Gambar 9.42 Membuat Parameters Jumping

43. Klik panah yang mengarah ke Jumping, pada inspector tambahkan condition, pilih condition Jumping dan ubah nilainya menjadi true.



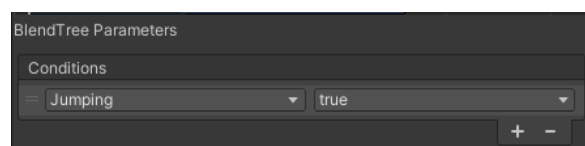
Gambar 9.43 Membuat *Condition*

44. Klik Settings dan ubah nilai Transition Duration menjadi 0 dan hilangkan centang Has Exit Time.



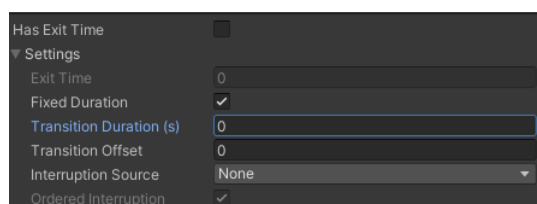
Gambar 9.44 Mengatur nilai *Transition Duration*

45. 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.



Gambar 9.45 Mengatur *Condition*

46. Klik Settings dan ubah nilai Transition Duration menjadi 0 dan hilangkan centang Has Exit Time.



Gambar 9.46 Tampilan *Settings*



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

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

Gambar 9.47 Menambahkan *script player*

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

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

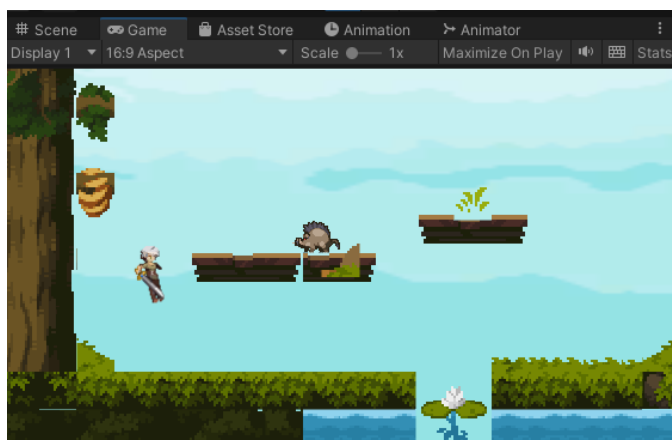
Gambar 9.48 Menambahkan *script player*

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

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

Gambar 9.49 Menambahkan *script player*

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



Gambar 9.50 Menambahkan *script player*



B. Link Github

https://github.com/rishramaa/2118054_PRAK_ANIGAME

9.2 Kuis

Menjelaskan 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 fungsi HandleJumpInput, baris kode animator.SetBool("isJumping, "); tidak ada nilai yang ditetapkan untuk parameter isJumping yang akan menyebabkan error. Sehingga diperlukan nilai boolean "true" atau "false" untuk mengatur nilai parameternya. Selanjutnya pada kondisi percabangan else if() seharusnya diubah ke GetKeyUp untuk deteksi akhir lompat, perbaikannya seperti dibawah.

```
void HandleJumpInput()
{
    if (Input.GetKeyDown(KeyCode.Space))
    {
```



```
        animator.SetBool("isJumping", true);
        rb.AddForce(Vector2.up * jumpForce, ForceMode2D.Impulse);
    }
    else if (Input.GetKeyUp(KeyCode.Space))
    {
        animator.SetBool("isJumping", false);
    }
}
```

Kemudian pada fungsi `HandleMovementInput`, kondisi percabangan pertama “`move != 1`” salah dan pernyataannya juga kurang. Kondisi ini seharusnya “`move == 0`” untuk mengecek apakah pemain sedang tidak bergerak, perbaikannya seperti dibawah.

```
if (move == 0)
{
    animator.SetBool("isIdle", true);
    animator.SetBool("isWalking", false);
}
else
{
    animator.SetBool("isIdle", false);
    animator.SetBool("isWalking", true);
    transform.Translate(Vector3.right * move * Time.deltaTime);
}
```

Pada kondisi percabangan kedua “`move != 1`” juga terdapat kesalahan, ini seharusnya “`move < 0`” untuk mencerminkan pemain menghadap kiri. Selain itu, nilai `Vector3` juga harus sama agar skala karakter tidak kacau, perbaikannya seperti dibawah.

```
if (move < 0)
{
    transform.localScale = new Vector3(-1, 1, 1);
}
else if (move > 0)
{
    transform.localScale = new Vector3(1, 1, 1);
}
```