



TUGAS PERTEMUAN: 9

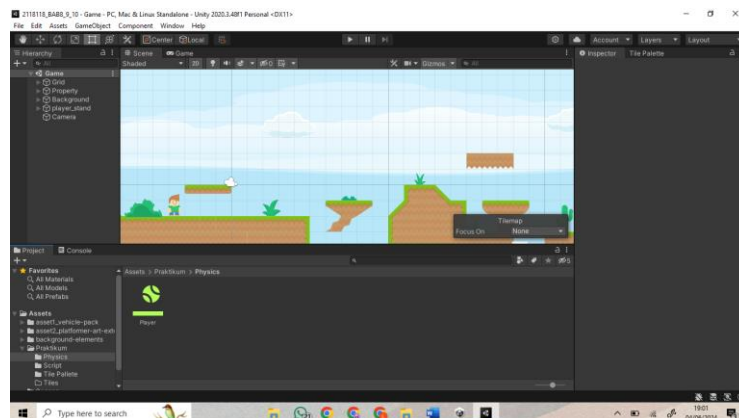
GAME ANIMATION

NIM	:	2118118
Nama	:	Legming Dwi Anggraini
Kelas	:	C
Asisten Lab	:	Nayaka Apta Nayottama (2218102)

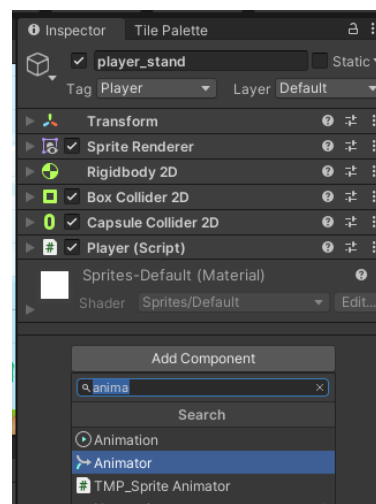
1.1 Tugas 1 : Membuat Game Animation

A. Membuat Game Animation

1. Buka project sebelumnya pada Unity

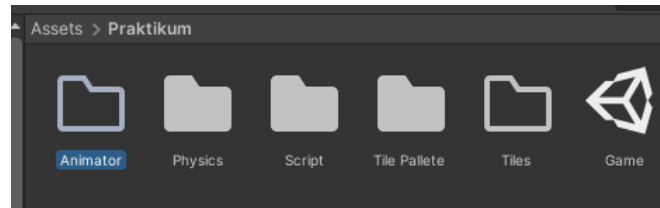


2. Tambahkan komponen animator pada player_stand

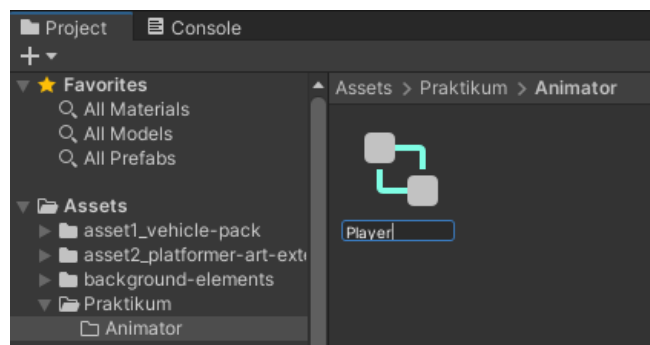




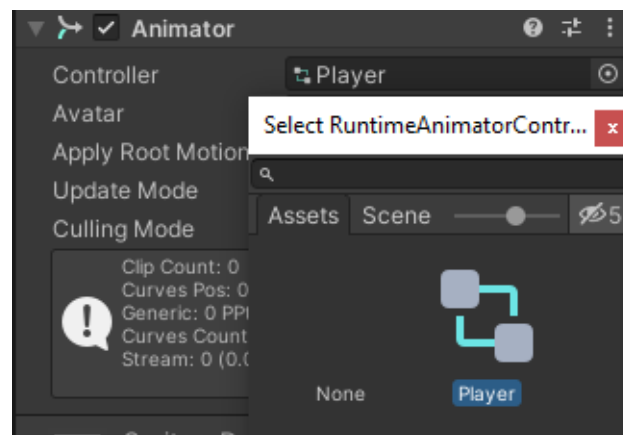
3. Membuat folder baru dengan nama Animator pada folder praktikum



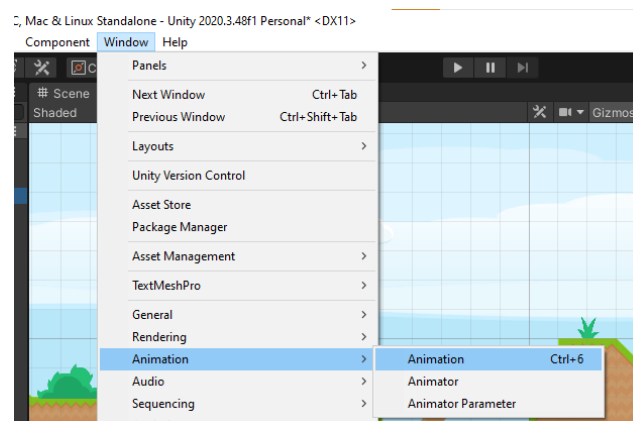
4. Membuat file animator controller dengan cara klik kanan dalam folder animator, lalu pilih create dan animator controller. Beri nama Player



5. Pada komponen animator, di bagian controller pilih Player

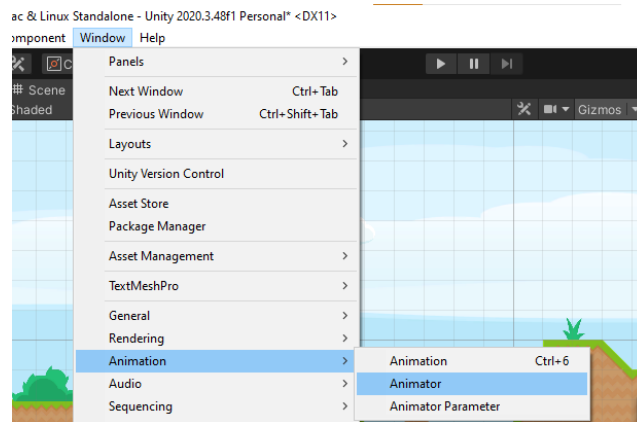


6. Tampilkan window Animation dengan cara klik tab window pilih animation lalu animation seperti berikut.

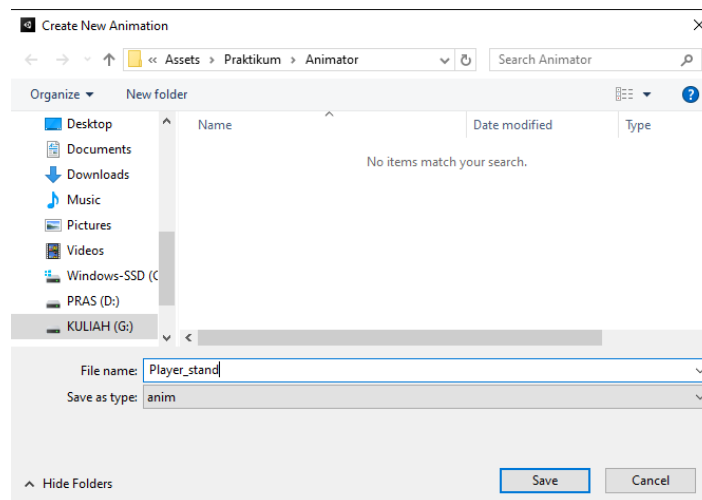




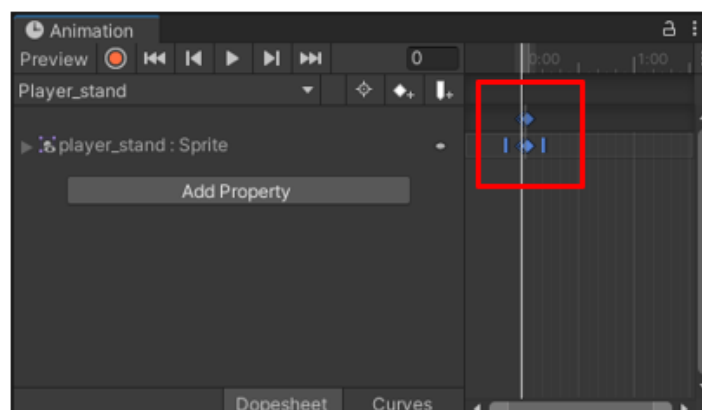
7. Tampilkan window animator dengan cara klik tab window pilih animation lalu animator.



8. Pada jendela animation klik button create and beri nama Player_stand dan disimpan pada folder animator.

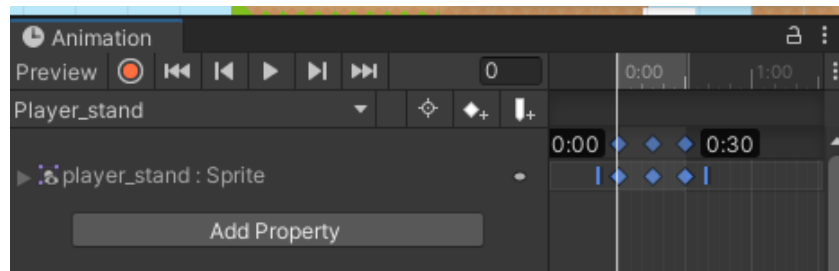


9. Buka asset player, lalu pilih asset dengan nama player stand dan player_idle. Kemudian, drag and drop ke bagian frame seperti berikut.

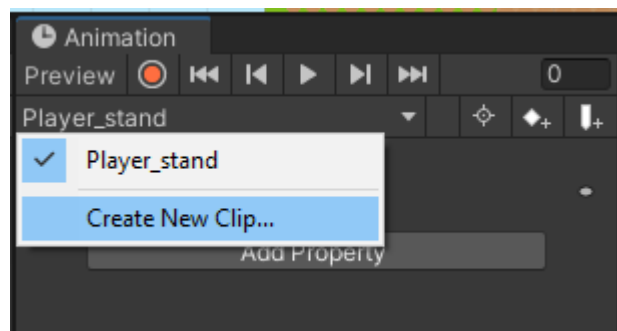




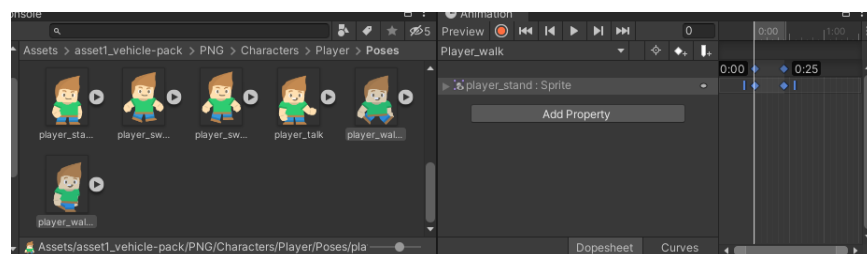
10. Tarik frame tersebut hingga frame ke-30 agar perubahan gerak tidak terlalu cepat.



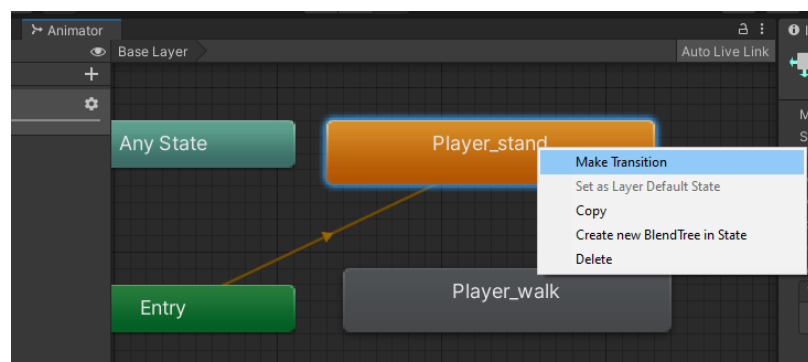
11. Buat clip baru dengan cara sebagai berikut, beri nama Player_walk



12. Kembali buka asset player, pilih player_walk1 dan player_walk2 kemudian drag and drop ke bagian frame animation. Ubah durasinya menjadi 25

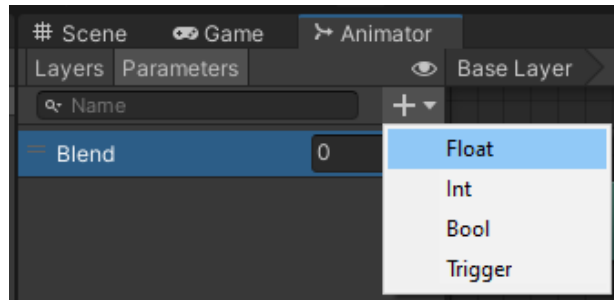


13. Buka window animator, klik kanan pada player_stand lalu make transition arahkan ke player_walk.

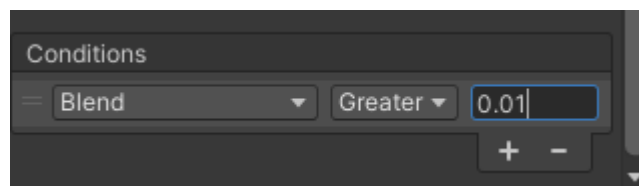




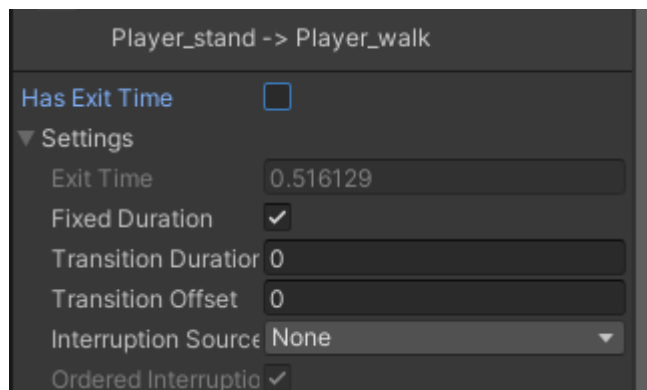
14. Buat parameter baru dengan tipe data float dan beri nama Blend



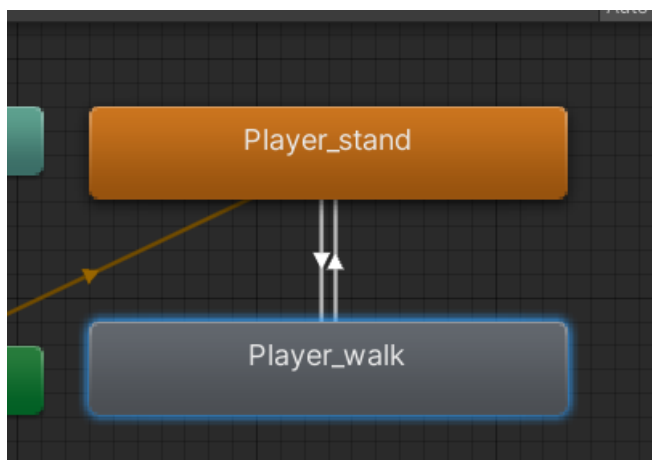
15. Klik arah panah player_stand ke player_walk, lalu pada bagian conditions klik *icon plus* ubah nilainya seperti berikut.



16. Kemudian, pada bagian settings *unchecked* bagian *has exit time* dan ubah nilai *transition duration* menjadi 0.

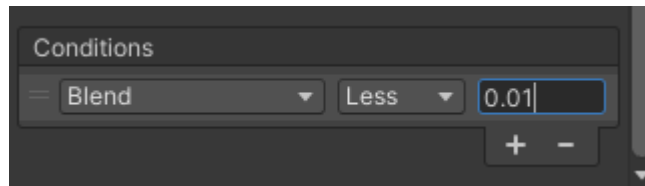


17. Buat transition dari arah player_walk ke player_stand





18. Lakukan langkah ke-15 dan 16, hanya saja perbedaan terletak pada bagian greater diganti dengan less.



19. Tambahkan source code berikut untuk mendefinisikan class animator

```
public Animator animator;
```

20. Tambahkan source code berikut pada void Awake().

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

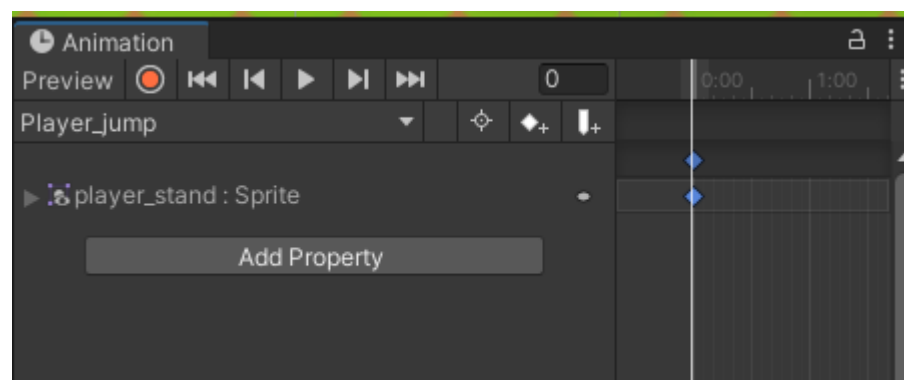
21. Tambahkan source code berikut pada void FixedUpdate()

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

22. Coba jalankan, maka tampilan player saat berjalan akan nampak bergerak melangkah.

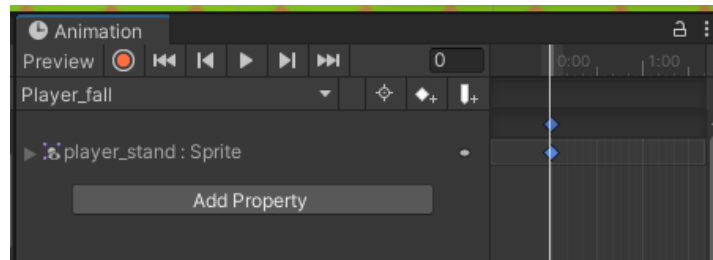


23. Buat clip baru dengan nama player_jump dan drag and drop asset player_jump

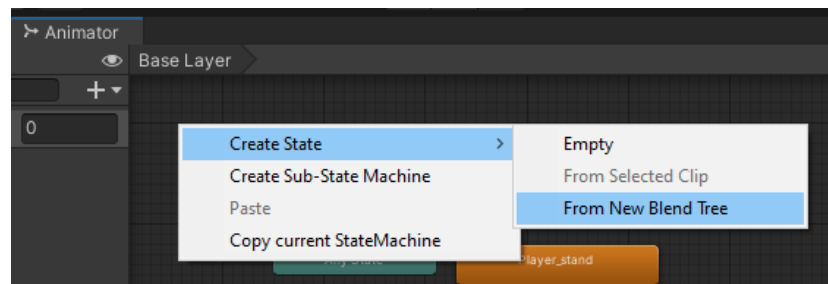




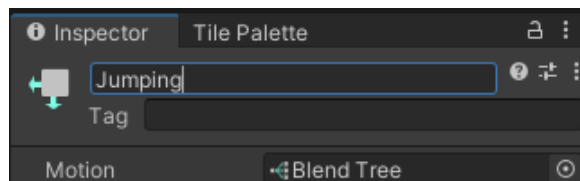
24. Buat juga clip player_fall, lalu drag and drop asset player_fall



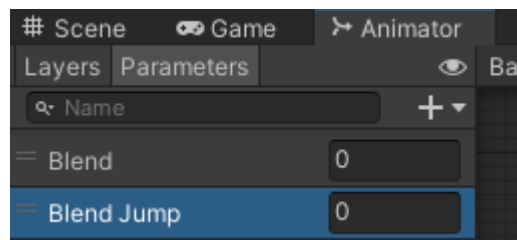
25. Buka jendela animator, klik pada bagian yang kosong kemudian create state dan from new blend tree.



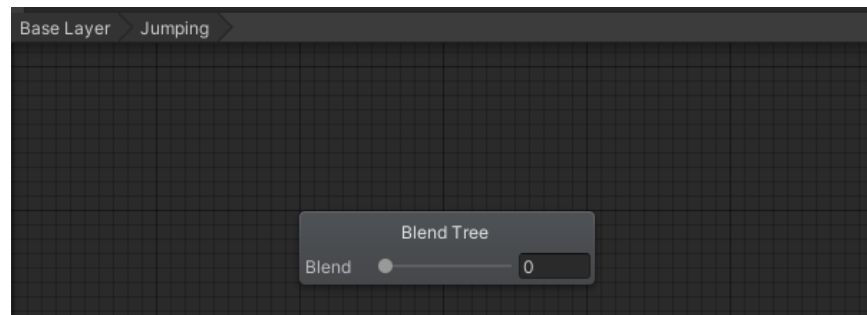
26. Setelah terbentuk blend tree baru, ubah namanya menjadi jumping pada bagian inspector.



27. Membuat parameter baru dengan tipe data float bernama Blend Jump.

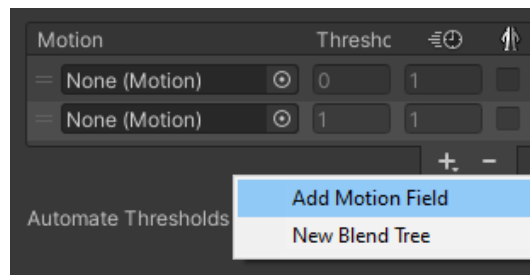


28. Kemudian, klik 2 kali blend tree yang telah dibuat, hingga masuk ke layer jumping.

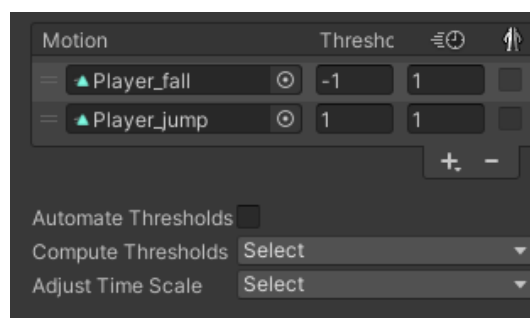




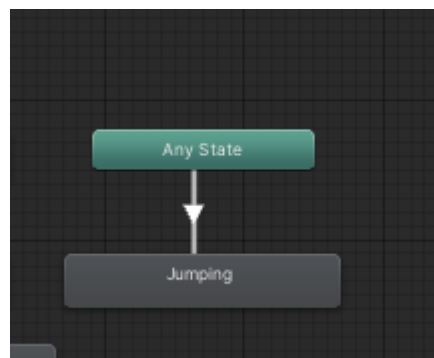
29. Klik bagian blend tree yang ada di no.28. Tambahkan dua motion field seperti berikut



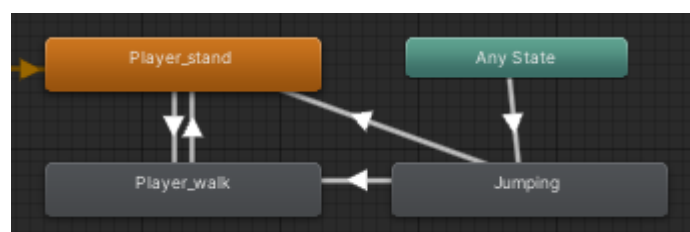
30. Atur motion seperti berikut, untuk mengubah nilai threshold unchecked dahulu menu automate threshold. Lalu sesuaikan nilainya.



31. Buat transition dari any state ke jumping.

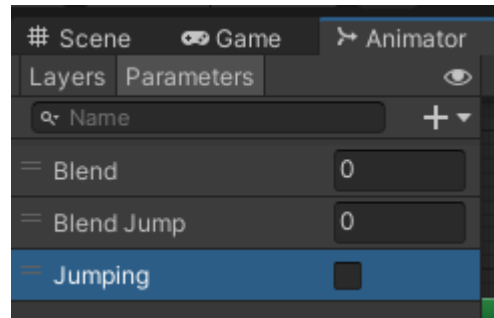


32. Buat juga transition dari jumping ke player_stand dan player_walk

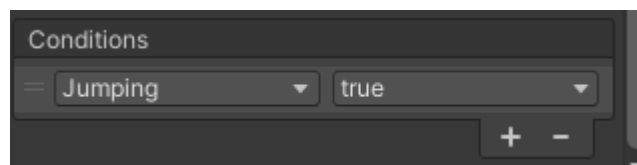




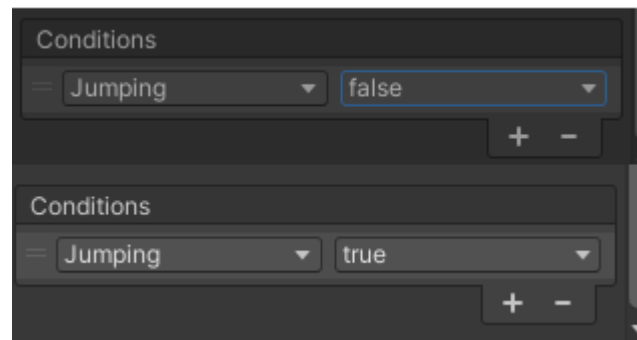
33. Tambahkan parameter dengan tipe data Boolean dan beri nama Jumping.



34. Klik arah panah any state ke jumping, lalu ubah bagian conditions seperti berikut. Kemudian, atur bagian settings seperti langkah ke-16.



35. Klik arah panah jumping ke player_stand dan player_walks, lalu ubah bagian conditions seperti berikut. Kemudian, atur bagian settings seperti langkah ke-16.



36. Tambahkan source code berikut untuk void Update().

```
if (Input.GetButtonDown("Jump")) {  
    animator.SetBool("Jumping", true);  
    jump = true;  
}
```

37. Tambahkan source code berikut pada void FixedUpdate().

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

38. Tambahkan source code berikut pada void GroundCheck()

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



39. Tampilkan hasilnya seperti berikut.



B. Kuis Analisis Error Source Code

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

void HandleMovementInput()
{
    float move = Input.GetAxis("Horizontal");

    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);

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



Hasil Analisis :

Pada source code di atas, terlihat bahwasanya error code berada pada void HandleJumpInput() di bagian pengkondisian isJumping = false. Kondisi tersebut akan menyetel isJumping ke false jika tombol Space masih ditekan, yang mungkin tidak sesuai dengan logika yang diinginkan. Umumnya, isJumping hanya diatur ke true saat melompat dan harus diatur ke false hanya ketika karakter menyentuh tanah (misalnya, menggunakan event dari collision).

C. Link Github Pengumpulan

https://github.com/Legming-DA/2118118_PRAK_ANIGAME.git