



TUGAS PERTEMUAN: 9

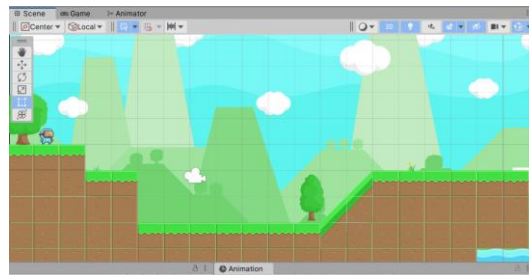
GAME ANIMATION

NIM	:	2118115
Nama	:	Ellok Ananda Madya Pratiwi
Kelas	:	C
Asisten Lab	:	Nayaka Apta Nayottama (2218102)

1.1 Tugas 1 : Game Animation

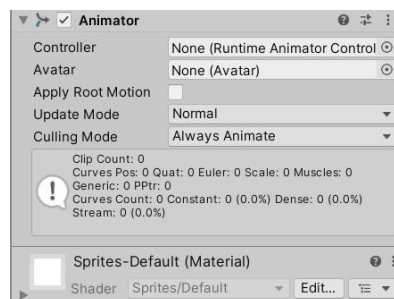
A. Membuat Animasi pada karakter

1. Buka file project unity sebelumnya



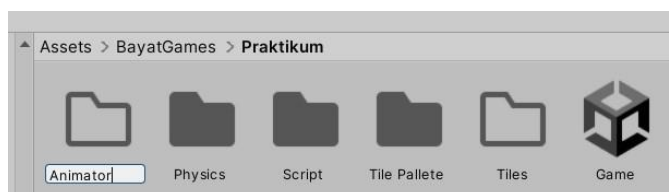
Gambar 9.1 Membuka Project

2. Klik player idle pada hirarki lalu tambahkan komponen animator pada inspector



Gambar 9.2 Menambah component animator

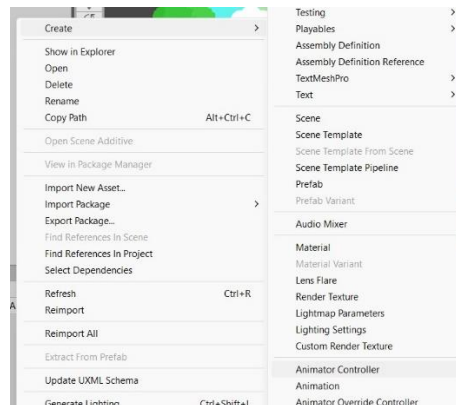
3. Buat folder baru dengan nama Animator didalam folder Praktikum



Gambar 9.3 Membuat folder Animator



4. Didalam folder Animator buat File Animator Controller



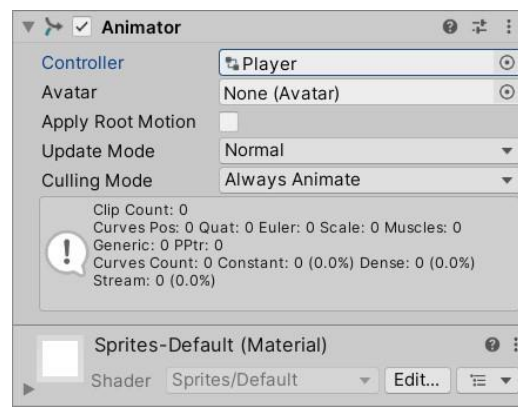
Gambar 9.4 Membuat Animator Controller

5. Beri nama Player



Gambar 9.5 Mengubah Nama

6. Pada inspector player idle ubah Controller menjadi Player



Gambar 9.6 Mengubah Controller

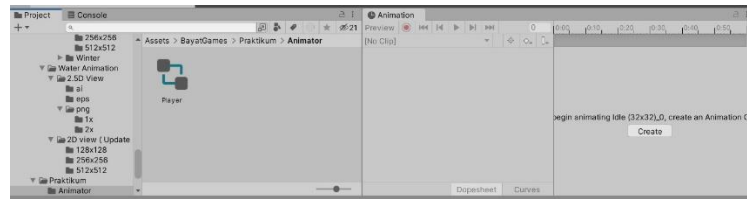
7. Tambahkan panel animation dengan pilih menu Window – Animation



Gambar 9.7 Menambahkan Panel Animation



8. Posisikan panel project split dengan panel animation



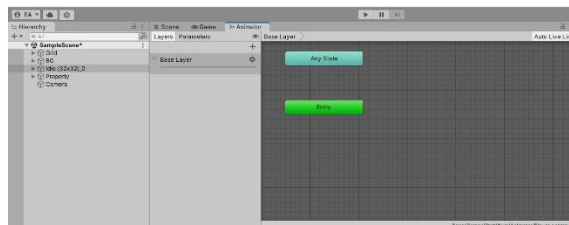
Gambar 9.8 Mengatur Posisi Panel Animation

9. Tambahkan menu Animator dengan pilih menu Window – Animator



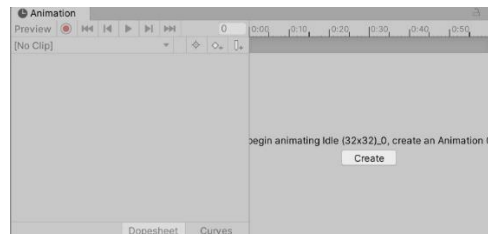
Gambar 9.9 Menambahkan Menu Animator

10. Maka akan tampil seperti gambar



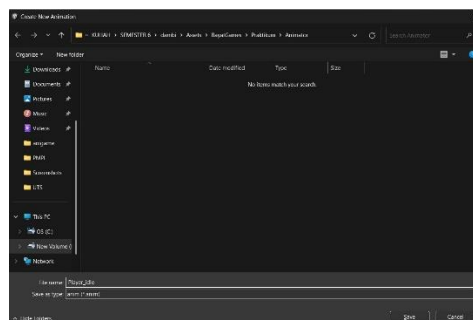
Gambar 9.10 Tampilan Menu Animator

11. Klik player idle pada hirarki lalu tekan create pada panel animation



Gambar 9.11 Membuat Clip

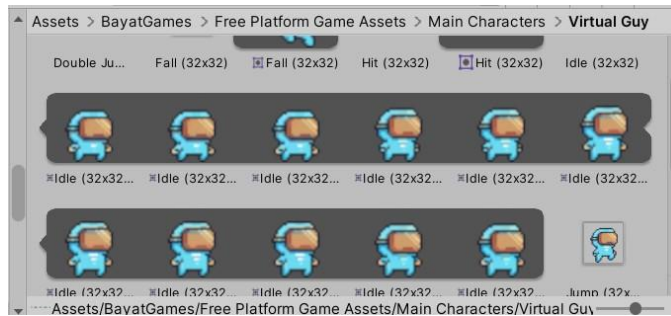
12. Arahkan ke direktori folder Animator dan beri nama Player_idle



Gambar 9.12 Membari Nama Clip Idle

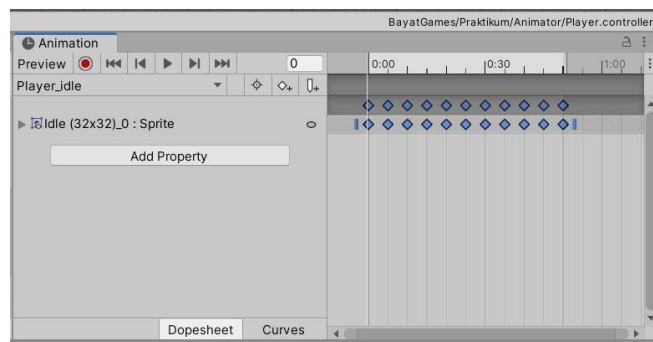


13. Cari folder player idle



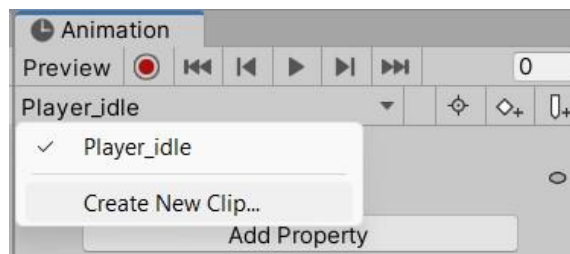
Gambar 9.13 Folder Idle

14. Drag ke panel animation dan Tarik kotak biru kecil hingga ke detik 50



Gambar 9.14 Mengatur Jarak

15. Buat clip baru dengan klik Player_idle beri nama Player_run



Gambar 9.15 Menambah Clip

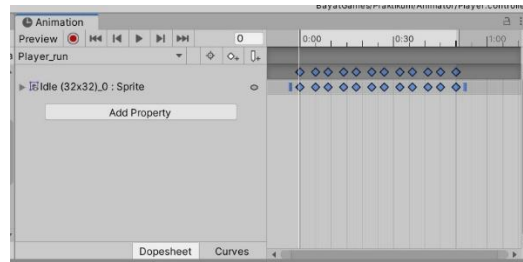
16. Cari folder player run



Gambar 9.16 Folder Run

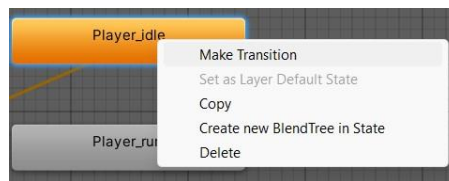


17. Drag ke panel animation dan Tarik kotak biru kecil hingga ke detik 50



Gambar 9.17 Mengatur Jarak

18. Pada menu Animator buat transition dari player_idle ke player_run dengan klik kanan player_idle – Make Transition



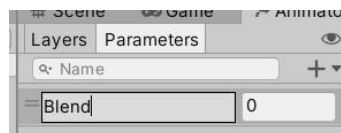
Gambar 9.18 Membuat Transition

19. Arahkan panah ke player_run



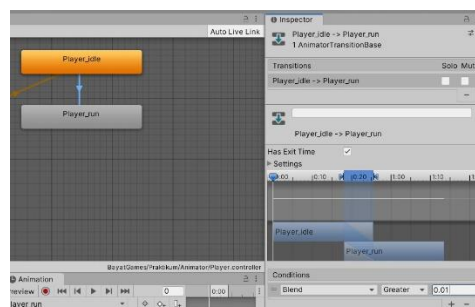
Gambar 9.19 Mengarahkan Panah

20. Pada parameter tambah tipe data float beri nama Blend



Gambar 9.20 Membuat Parameter

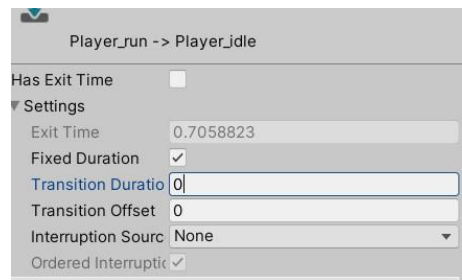
21. Klik panah player_idle ke player_run lalu tambah conditions Blend dan ubah nilainya menjadi 0.01



Gambar 9.21 Menambahkan Conditions



22. Pada settings ubah transition Duration menjadi 0 dan unchecklist pada Has exit time



Gambar 9.22 Mengatur Duration

23. Tambahkan source code berikut pada Player.cs

```
public class Player : MonoBehaviour
{
    public Animator animator;
    Rigidbody2D rb;
    [SerializeField] Transform groundcheckCollider;
    [SerializeField] LayerMask groundLayer;
}
```

Gambar 9.23 Mengubah Source Code

24. Ubah method Awake menjadi seperti berikut

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

Gambar 9.24 Mengubah Method Awake

25. Ubah method FixedUpdate menjadi seperti berikut

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

Gambar 9.25 Mengubah Method FixedUpdate

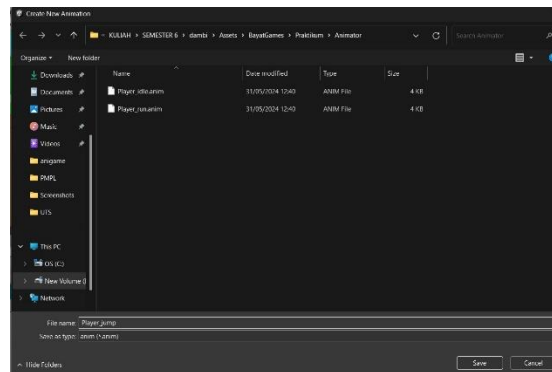
26. Jalankan maka player dapat memiliki animasi Ketika berhenti dan berjalan



Gambar 9.26 Menjalankan Karakter

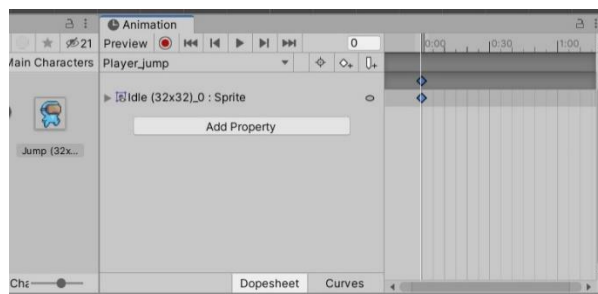


27. Tambah clip baru dengan nama Player_jump



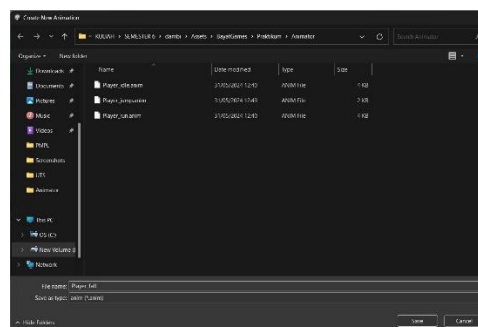
Gambar 9.27 Menambahkan Clip

28. Cari folder player jump dan drag ke animation



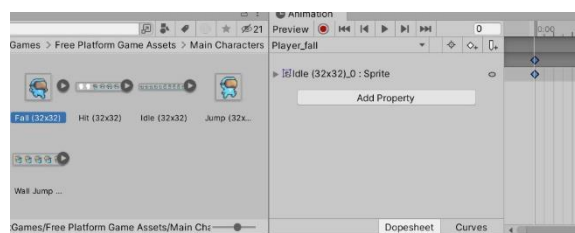
Gambar 9.28 Folder Jump

29. Tambah clip baru dengan nama Player_fall



Gambar 9.29 Menambah Clip

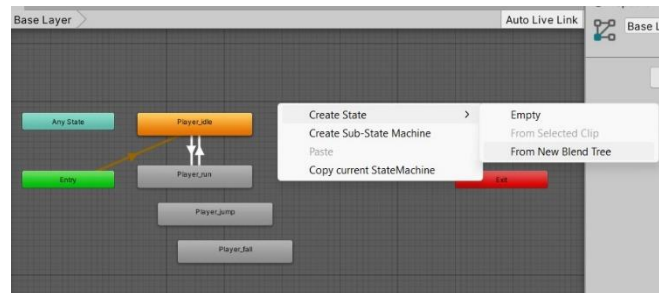
30. Cari folder player fall dan drag ke animation



Gambar 9.30 Folder Fall

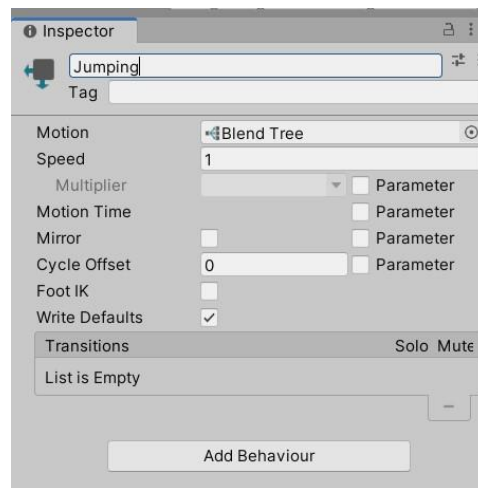


31. Pada animator klik kanan pada area kosong lalu Create State – From New Blend Tree



Gambar 9.31 Menambahkan Blend Tree

32. Klik blend Tree lalu pada inspector ubah Namanya menjadi Jumping



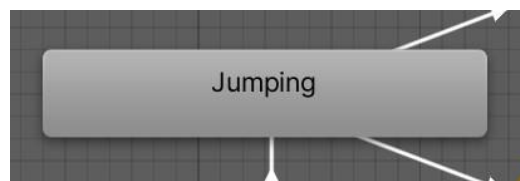
Gambar 9.32 Mengubah Nama

33. Tambah tipe data float pada parameter dengan nama Blend Jump



Gambar 9.33 Menambah Parameter

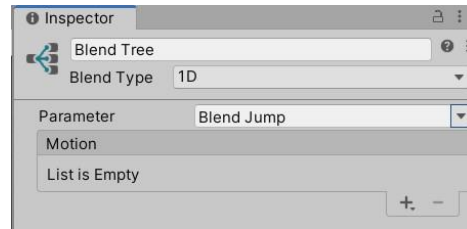
34. Pada animator, klik 2 kali pada Blend Tree Jumping



Gambar 9.34 Blend Tree

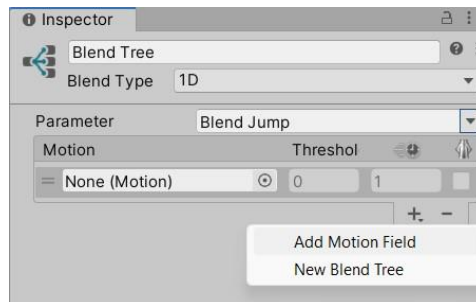


35. Pada inspector, ubah parameter menjadi Blend Jump



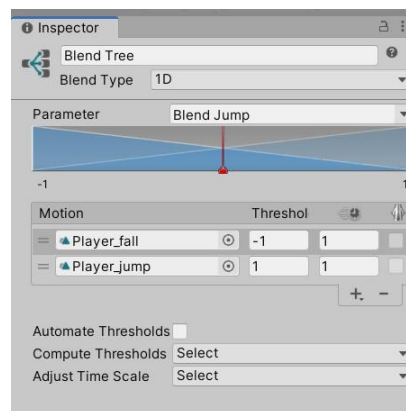
Gambar 9.35 Mengubah Parameter

36. Klik tanda panah lalu pilih Add Motion Field sebanyak 2 kali



Gambar 9.36 Menambahkan Motion

37. Pilih Player_fall dan Player_jump pada Motion. Lalu unchecklist Automate Thresholds dan sesuaikan thresholds dengan gambar



Gambar 9.37 Mengatur Inspector Blend Tree

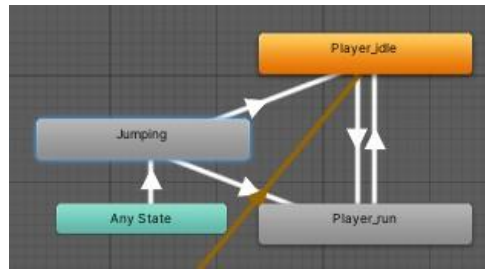
38. Buat transition dari Any State ke Jumping



Gambar 9.38 Membuat Transition

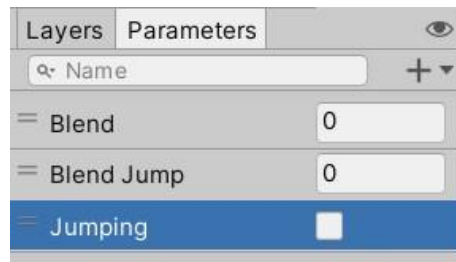


39. Dari Jumping, buat Transition ke Player_idle dan Player_run



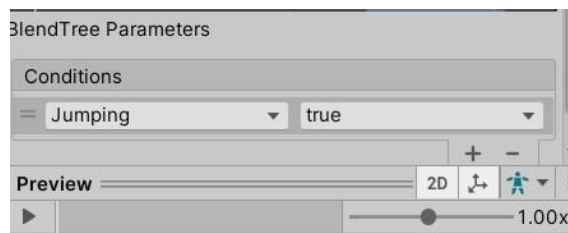
Gambar 9.39 Membuat Transition

40. Tambah tipe data bool pada parameter dan beri nama Jumping



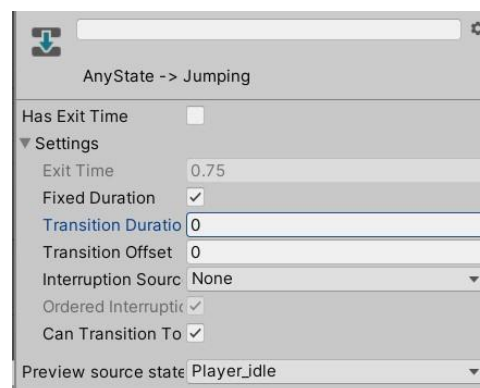
Gambar 9.40 Menambahkan Parameter

41. Klik panah dari Any State ke Jumping lalu tambahkan conditions Jumping dengan nilai true



Gambar 9.41 Menambahkan Conditions

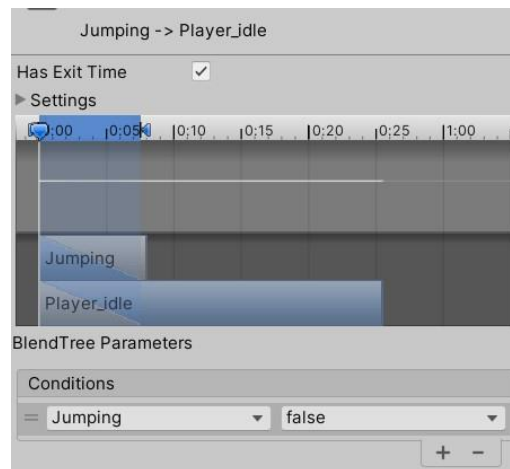
42. Unchecklist has exit time dan beri nilai 0 pada transition duration



Gambar 9.42 Mengatur Inspector Jumping

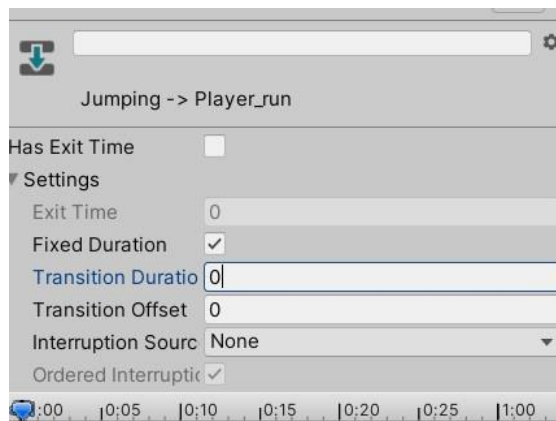


43. Klik panah yang mengarah ke Player_idle dari Jumping lalu tambahkan conditions Jumping dengan nilai false



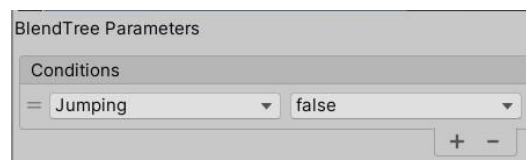
Gambar 9.43 Menambahkan Conditions

44. Klik panah yang mengarah ke Player_run dari Jumping lalu unchecklist has exit time dan ubah transition duration menjadi 0



Gambar 9.44 Mengatur Inspector Player_run

45. Tambah conditions Jumping dengan nilai false



Gambar 9.45 Menambahkan Conditions



46. Tambahkan code berikut pada method Update, FixedUpdate, dan GroundCheck

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

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

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

void GroundCheck()
{
    isGrounded = false;
    Collider2D[] colliders = Physics2D.OverlapCircleAll(groundcheckCollider, 1f);
    if (colliders.Length > 0){
        isGrounded = true;
    }
    animator.SetBool("Jumping", !isGrounded);
}
```

Gambar 9.46 Mengubah Code

47. Hasil render



Gambar 9.47 Render



Kuis :

```
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);
        }
    }
}
```

Analisa :

pada method HandleJumpInput() terdapat variable animatpr yang diberi nilai true bertipe data Boolean. Kondisi if terpenuhi jika keyboard space ditekan nilai menjadi true. jika space tidak ditekan maka nilai false. Pada method HandleMovement Input() pembetulannya pada agar isIdle diatur menjadi true hanya Ketika nilai dari variable move harus 0. Ketika nilai dari variable move tidak sama dengan 0 makan menjalankan statement 'animator.SeBool("isWalking", true). Menyesuaikan



'transform.localScale' agar karakter menghadap ke arah yang benar berdasarkan nilai 'move.