



## TUGAS PERTEMUAN: 9

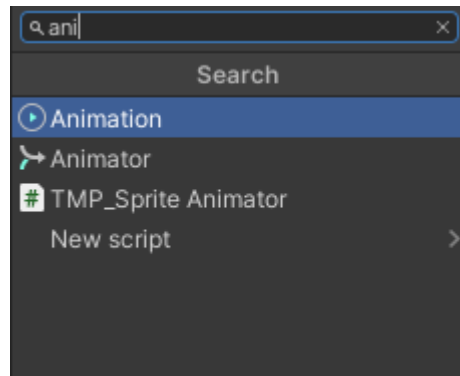
### GAME ANIMATION

NIM	:	2118075
Nama	:	Abdillah Arif Rohman
Kelas	:	B
Asisten Lab	:	MARIA AVRILIANA SURATLELAONA (2218108)

#### 9.1 Tugas 9 : Membuat Karakter Beranimation

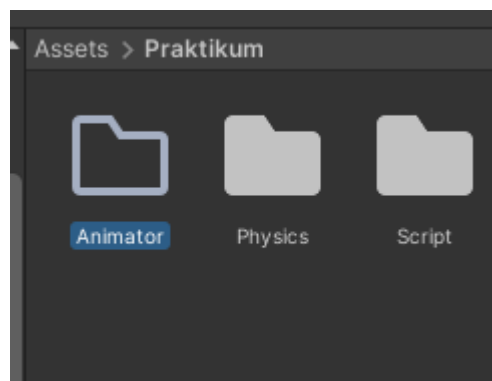
##### A. Character Animation

1. Pertama, Pencet karakter klik *inspector* kemudian pilih *Add Component Animator*



Gambar 9.1 Componen Animator

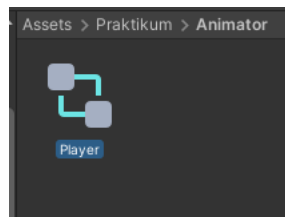
2. Lalu Buat Folder dengan Nama Folder Di Dalam Praktikum Buat Folder Animator



Gambar 9.2 Create Folder Animator

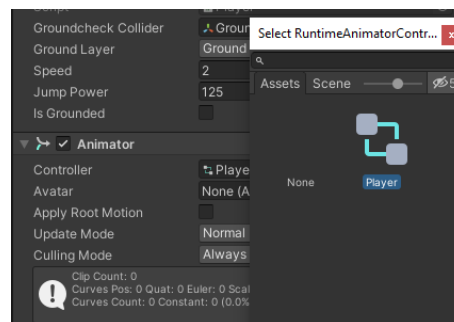


3. Kemudian Kita Membuat File *Animator Controller* pada folder *Animator*, ubah namanya menjadi *Player*



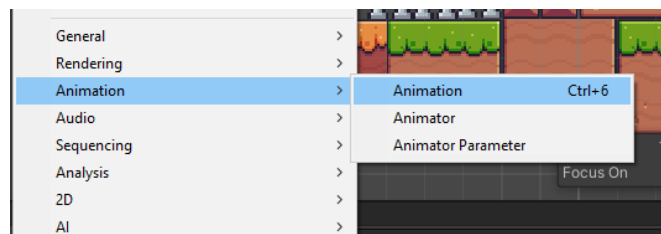
Gambar 9.3 File Controller Player

4. Pencet Karakter Di *Hierarchy*, kemudian cari **Component Animator**, pada *setting Controller* ubah menjadi *Player*.



Gambar 9.4 Ubah Jadi Player

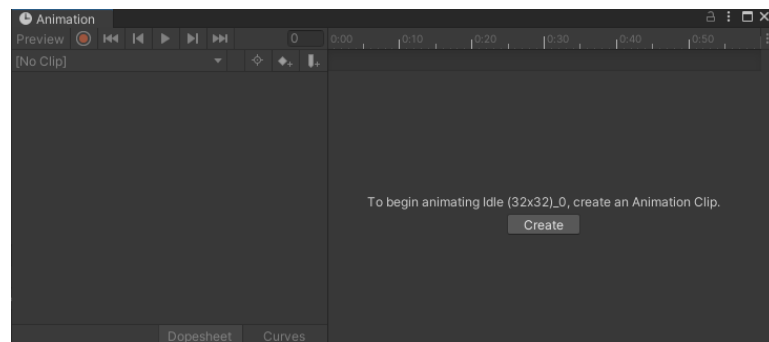
5. Jika Sudah, Ke menu panel *Animation* di menu Window, pilih Animation Lalu Animation.



Gambar 9.5 Membuat Animation



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



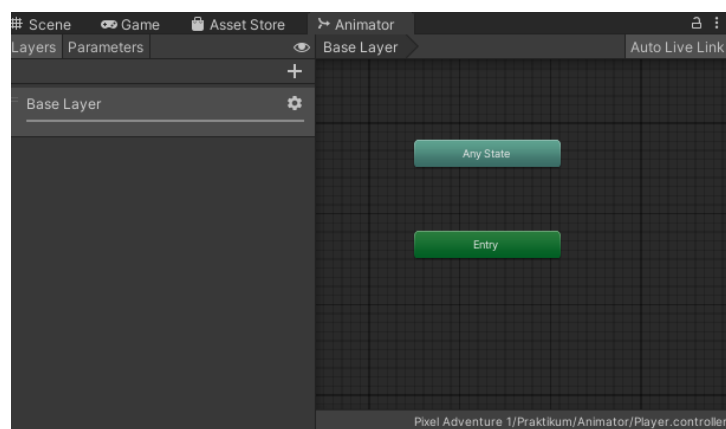
Gambar 9.6 Menu Animation

7. Ke menu panel *Animator*



Gambar 9.7 Panel Animator

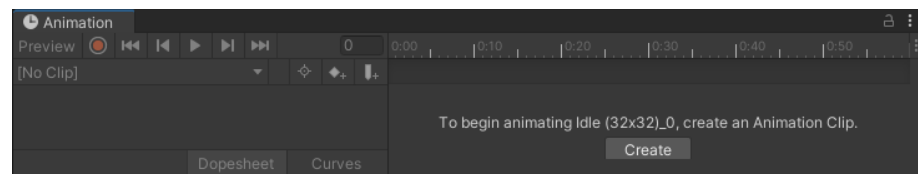
8. Panel tersebut sesuai dengan gambar.



Gambar 9.8 Base Layer

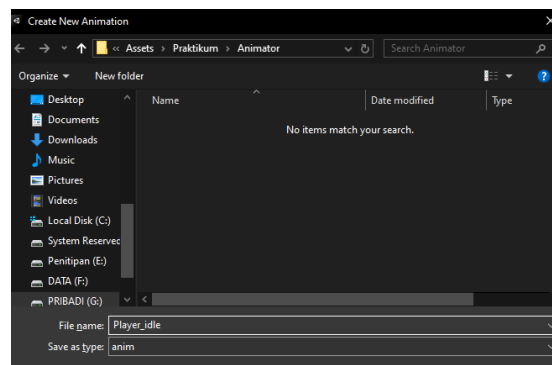


9. Kita Akan membuat animasi klik player Fall pada *Hierarchy*, kemudian ke menu panel *Animation*, Ke *Create*.



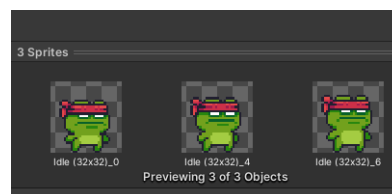
Gambar 9.9 Create Player Animation

10. Kita Save pada folder Animator dan beri nama Player\_idle



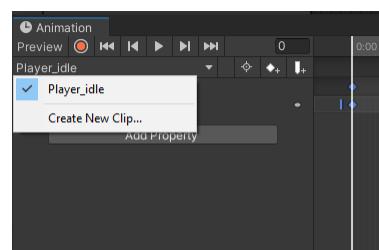
Gambar 9.10 Menyimpan File Player Animation

11. Lalu menu Project buka folder player lalu pilih Idle dan pilih gambar idle 1, idle 2 Sampai idle Akhir, kemudian drag ke tab *Animation*



Gambar 9.11 Drag And Drop Player idle

12. Kita Membuat animasi baru, Klik pada “Player\_idle” kemudian pilih *Create New Clip*, dan beri nama “Player\_run”, Simpan pada Folder *Animator*



Gambar 9.12 Create New Clip

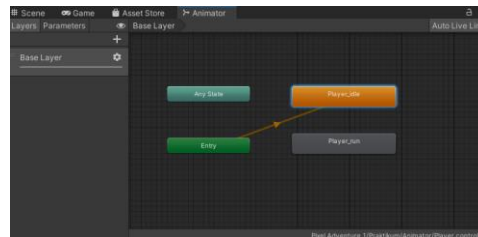


13. Pada menu Project kemudian cari folder Player Pilih run, Pilih Fall run sampai Fall Akhir, drag and drop pada menu *Animation*



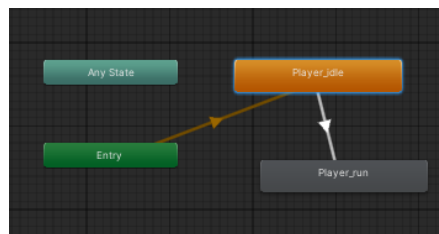
Gambar 9.13 Drag And Drop Player Run

14. ke menu *Animator* yang telah dibuka sebelumnya dan akan tampil seperti berikut.



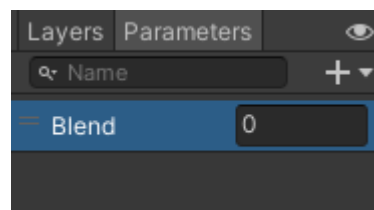
Gambar 9.14 Base Layer Animation

15. Setelah Itu, 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 9.15 Sambungkan Transisi

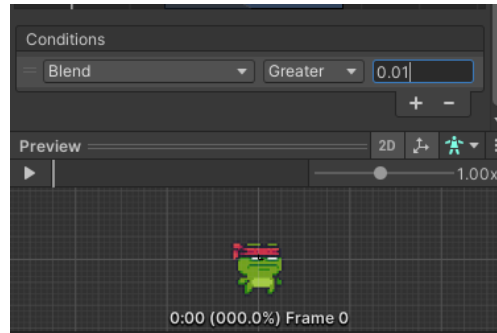
16. Lalu ke tab parameter, tambahkan tipe data Tekan icon tambah dan ubah namanya menjadi “Blend”



Gambar 9.16 Parameter

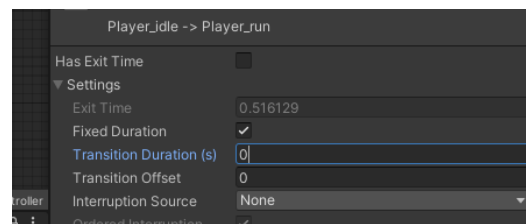


17. Lalu Pencet panah putih tersebut, pada bagian *conditions* klik icon tambah kemudian atur menjadi *Blend* Dan Atur nilai *conditions blend* Jadi 0.01.



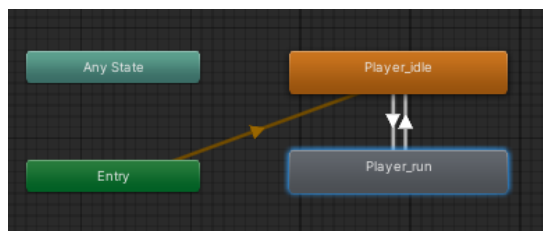
Gambar 9.17 Mengatur Kondisi Animasi

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



Gambar 9.18 Seting Has Exit Time

19. Buat transisi juga dari *player\_run* ke *player\_idle* dengan cara klik kanan pada *player\_run* dan pilih *Make Transition*.



Gambar 8.19 Layer Transisi Player idle Dan Run

20. Jika Mau Animasi dapat sesuai ketika berjalan, buka script *Player* dan tambahkan *source code* berikut pada *class Player*.

```
public class Player : MonoBehaviour
{
    Rigidbody2D rb;
    public Animator animator;
```



21. Pada Script Komponen *Animator*

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

22. Kemudian pada fungsi *FixedUpdate* tambahkan source code berikut.

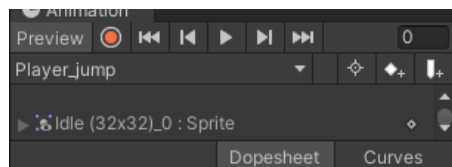
```
animator.SetFloat("Blend", Mathf.Abs(rb.velocity.x));
```

23. dijalankan maka Karakter dapat memiliki animasi ketika berhenti ataupun ketika berjalan



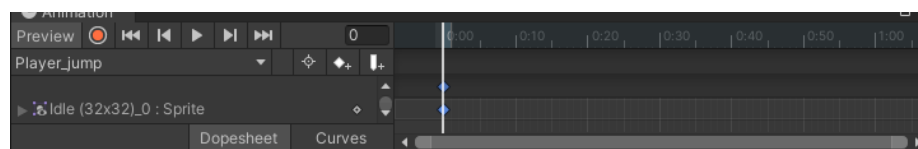
Gambar 9.20 Hasil Animasi Berjalan

24. Lalu, buat animasi baru tekan tulisan “Player\_run” kemudian pilih *Create New Clip*, dan beri nama “Player\_jump”



Gambar 9.21 Membuat Animasi Jump

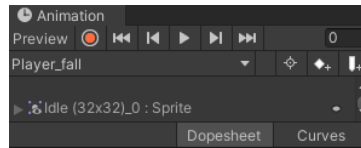
25. folder player buka jump lalu pilih gambar player-jump-1, kemudian drag ke tab Animation.



Gambar 9.22 Drag And Drop Player Jump

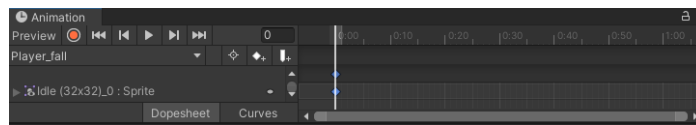


26. Kita Buat animasi baru dengan cara tekan tulisan “Player\_jump” kemudian pilih *Create New Clip*, dan beri nama “Player\_fall”



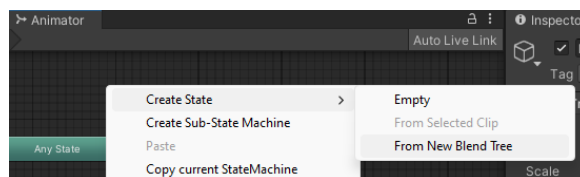
Gambar 8.23 Membuat Animasi Fall

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



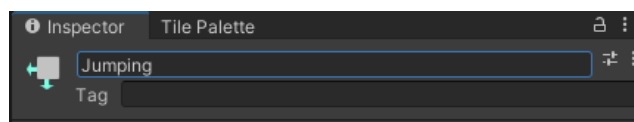
Gambar 8.24 Drag And Drop Player Fall

28. Setelah Itu untuk menambahkan animasi ketika melompat. Klik kanan pada menu Animator, di area kosong , pilih *Create State* Pilih *From New Blend Tree*.



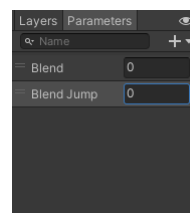
Gambar 9.25 Create State

29. Jika Sudah *Animator* klik *Blend Tree*, di menu *Inspector*, ubah namanya menjadi *Jumping*.



Gambar 9.26 Mengubah Nama Inspector

30. Menu *Parameters* tambahkan parameter tipe data *Float* dan ubah namanya menjadi *Blend Jump*

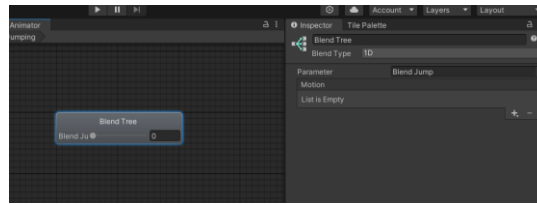


Gambar 9.27 Menambahkan Parameter



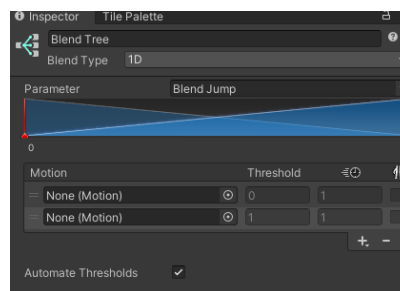


31. Kita Ke menu Animator, Klik dua kali pada *Blend Tree* Jumping, Tekan pada *Blend Tree*



Gambar 9.28 Menu Inspector Blend Tree

32. Pencet 2x *Blend Tree* Jumping, pada *inspector* ubah parameter menjadi *Blend Jump*.



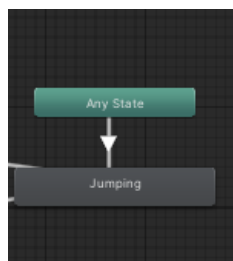
Gambar 9.29 Inspector Parameter

33. Ke bagian icon *None Motion*, maka akan muncul *Windows Motion*, Tambahkan Sesuai dengan urutan.



Gambar 9.30 Motion Windows

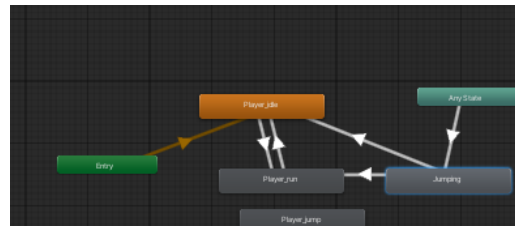
34. Kembali ke *Base Layer*, klik kanan *Any State*, pilih *Make Transition* dan arahkan panahnya ke *Jumping*



Gambar 9.31 Pilih Project Tugas

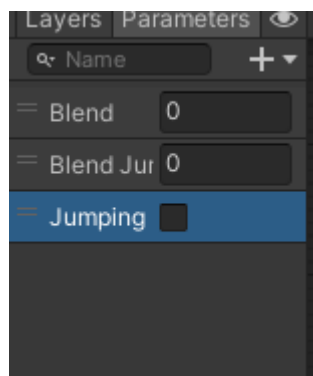


35. Kemudian Klik kanan Jumping, pilih *Make Transition* dan arahkan panahnya ke Player\_idle dan Player\_run.



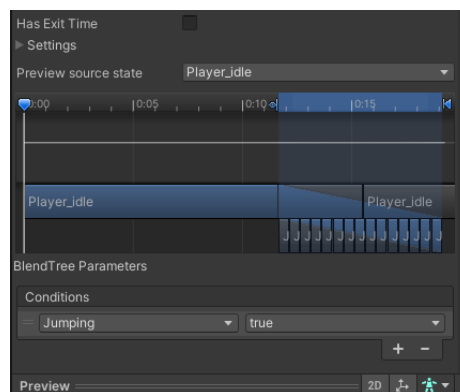
Gambar 9.32 Transisi Jumping Ke Player Idle dan Run

36. Setelah Itu, parameter transisi dengan tipe data *Bool* tdan ubah namanya menjadi Jumping



Gambar 9.33 Tambah Parameter

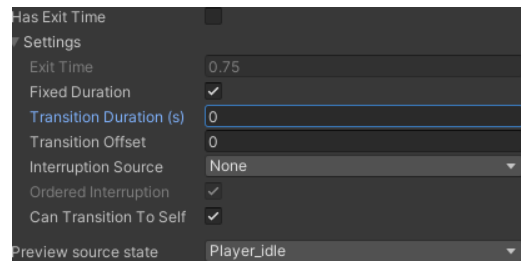
37. Lalu Pilih panah yang mengarah ke Jumping, pada *inspector* tambahkan *condition*, pilih *condition* Jumping dan ubah nilainya menjadi *true*.



Gambar 9.34 Mengganti Kondisi Animasi

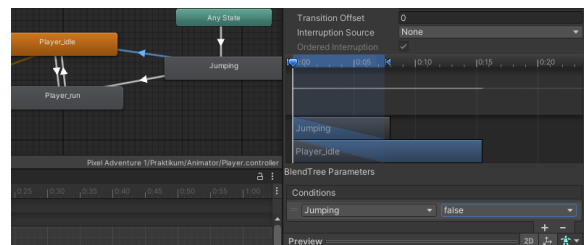


38. Pilih *Settings* dan ubah nilai *Transition Duration* menjadi 0 dan hilangkan centang *Has Exit Time*.



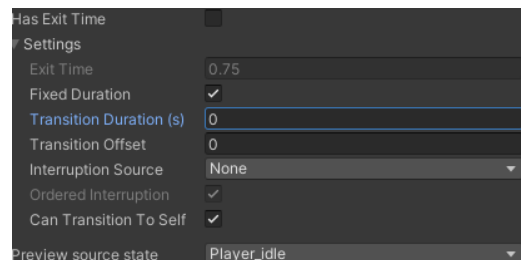
Gambar 9.35 Menghilangkan Ceklis

39. 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 9.36 Menggnati Kondisi True

40. Ke *Settings* dan ubah nilai *Transition Duration* menjadi 0 dan hilangkan centang *Has Exit Time*.



Gambar 9.37 Mengnnati Angka Jadi 0

41. Ke *Visual Stuio Code* Cari script *Player*, dan tambahkan *source code* berikut pada fungsi *update*.

```
animator.SetBool("Jumping", true);
```

42. Lalu Fungsi *FixedUpdate* tambahkan seperti berikut.

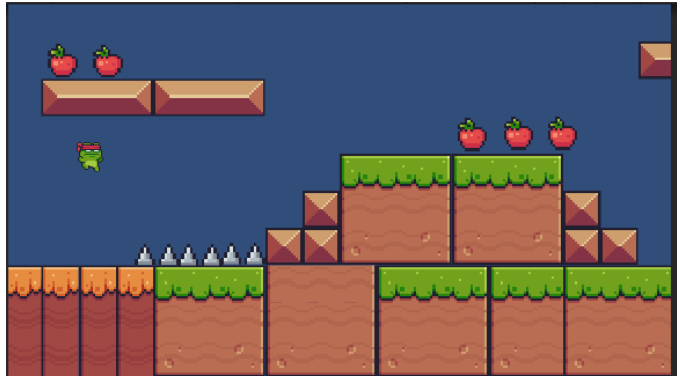
```
animator.SetFloat("Blend Jump", rb.velocity.y);
```

43. Tambahkan baris kode seperti dibawah ini dalam method *GroundCheck*

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



44. Untuk Hasil Akhirnya sudah bisa bergerak dengan animasi



Gambar 9.38 Hasil Akhir

## B. Kuis

```
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 :

Source Code yang dicantumkan berfungsi sebagai pemberi perintah untuk karakter melakukan lompatan pada karakter yang dilakukan dengan tombol spasi. Jika tombol spasi ditekan `Input.GetKeyDown(KeyCode.Space)` melakukan pengecekan dan melakukan lompatan dengan fungsi `isjumping` diatur menjadi `true` dan diberikan gaya lompatan dari fungsi `rb.AddForce`

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

[https://github.com/abdillahar/2118075\\_PrakAniGame](https://github.com/abdillahar/2118075_PrakAniGame)