



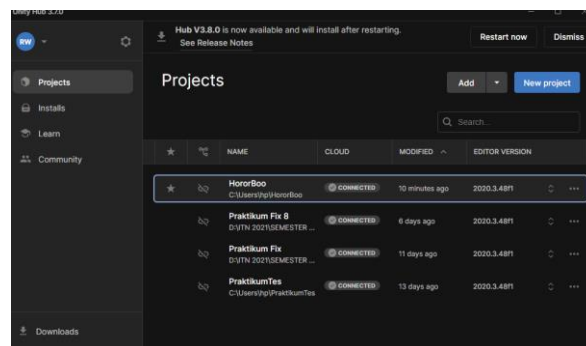
TUGAS PERTEMUAN: 9

Character Animation

NIM	:	2118055
Nama	:	Ridho Arif Wicaksono
Kelas	:	B
Asisten Lab	:	Difa Fisabililah (2118052)

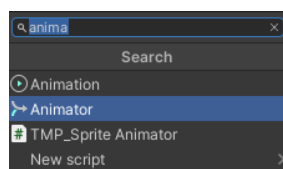
1.1 Tugas : Character Animation

1. Buka unity hub lalu klik pada project sebelumnya



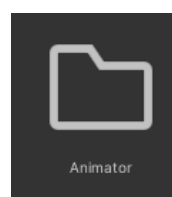
Gambar 9. 1 Project Baru

2. Setelah itu klik pada karakter lalu masuk kedalam abgian inspector, dan klik add component tulis “Animator”



Gambar 9. 2 Menambah Komponen Animator

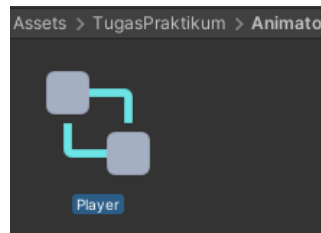
3. Pada folder Tugas Praktikum buat folder baru bernama “Animator” untuk menyimpan file animasi



Gambar 9. 3 Folder Animator

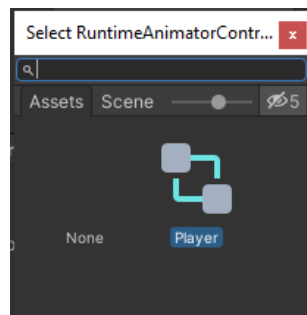


4. Membuat file animator controller dengan cara klik kanan create->Animator Controller buat animator bernama “Player” di dalam folder Animator



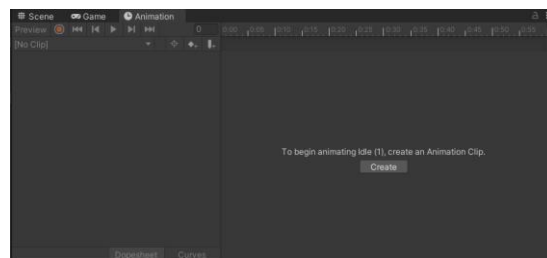
Gambar 9. 4 File Animator Controller

5. Setelah itu klik pada “Idle 1” lalu masuk kedalam inspector lalu ke animator pada bagian Animator Controller ubah Controller menjadi Player



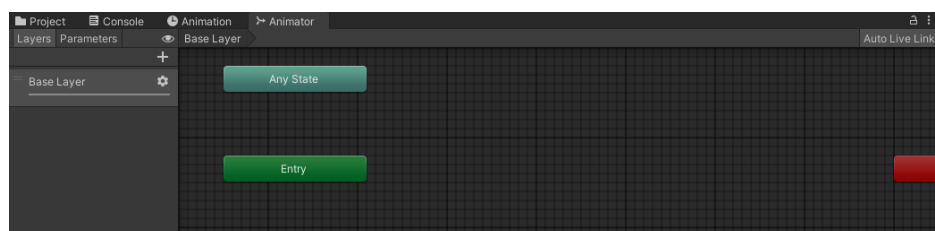
Gambar 9. 5 Setting Controller Animator

6. Setelah itu tambah jendela bagian animasi, dengan cara klik Window->animation->Animation, setelah itu letakan di bawah



Gambar 9. 6 Jendela Bagian Animasi

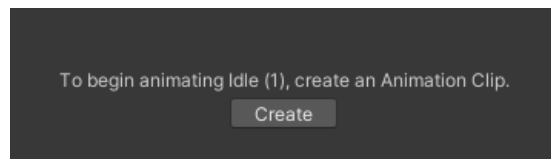
7. Setelah itu masuk kedalam menu Window->Animation->Animator setelah itu geser jendela Animator ke bawah sebelah menu Animation



Gambar 9. 7 Halaman Animator

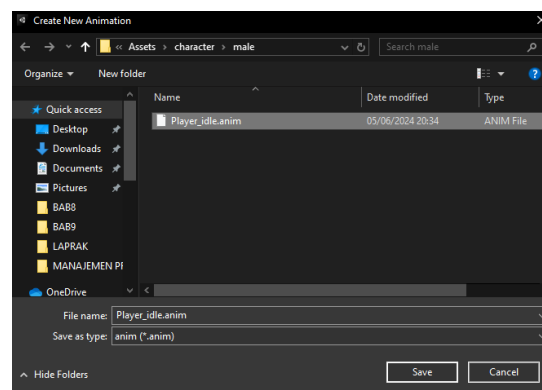


8. Setelah itu buat animasi baru dengan cara klik Idle 1 lalu masuk kedalam jendela Animation lalu klik Create



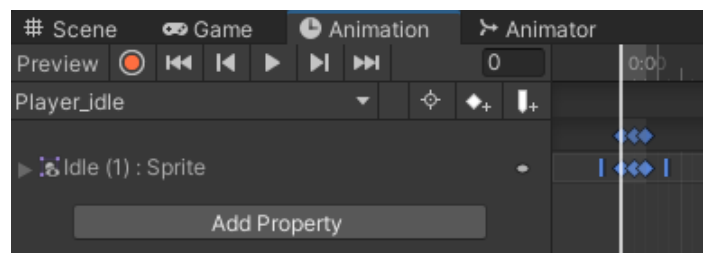
Gambar 9. 8 Membuat Animasi Baru

9. Setelah itu klik Create beri nama Player_idle dengan format anim, file ini digunakan untuk menyimpan animasi dari kondisi karakter



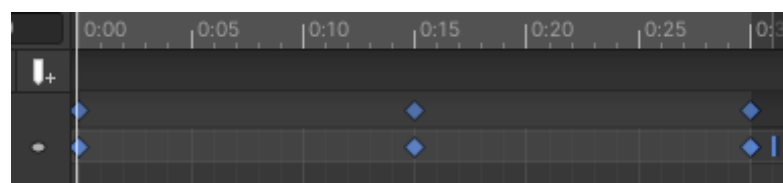
Gambar 9. 9 Membuat Animasi Idle

10. Setelah itu klik Save masuk kedalam folder Tugas Praktikum cari file bernama Idle 1, Idle 5, dan Idle 7 drag dan drop kedalam Animation



Gambar 9. 10 Animasi Idle

11. Setelah itu agar membuat animasi bertahan pada detik lama tidak berjalan terlalu cepat, masuk kedalam jendela Animation lalu geser kotak kotak kecil sampai frame 35



Gambar 9. 11 Mengatur Detik Animasi

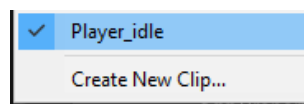


12. Setelah coba jalankan game nya untuk mencoba apakah animasi telah berhasil dibuat



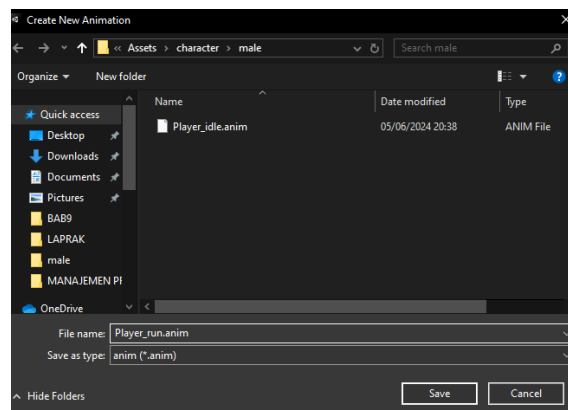
Gambar 9. 12 Menjalankan Game

13. Setelah itu buat animasi baru lagi bernama Player Run tetap di hierarki Idle 1 (Karakter), klik Player_idle lalu klik New Clip



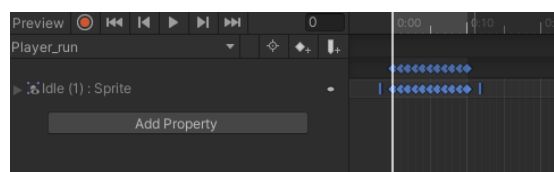
Gambar 9. 13 Create New Clip

14. Simpan dengan nama Player_run.anim



Gambar 9. 14 Membuat Animasi Berlari

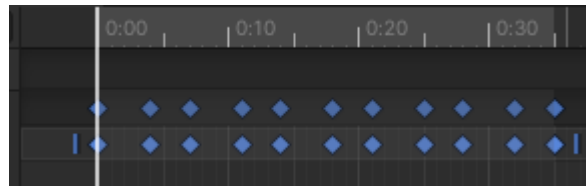
15. Setelah itu masuk kedalam animasi Player_run lalu drag drop file yang bernama Walk(1)-Walk(8)



Gambar 9. 15 Animasi Run

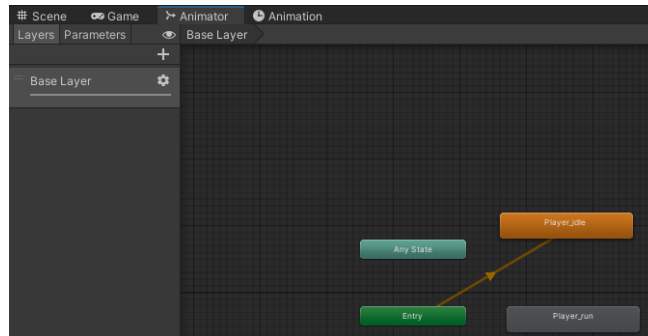


16. Setelah itu geser animasi sampai dengan detik 35



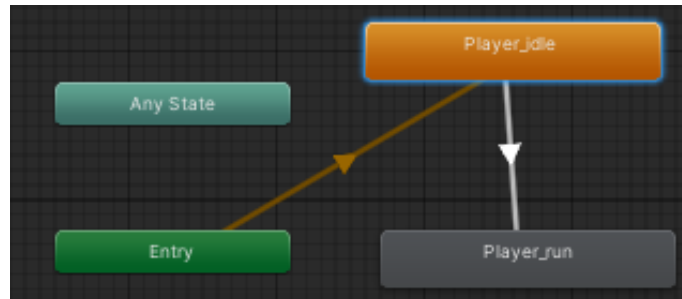
Gambar 9. 16 Memperpanjang Animasi

17. Setelah itu masuk kedalam jendela Animator



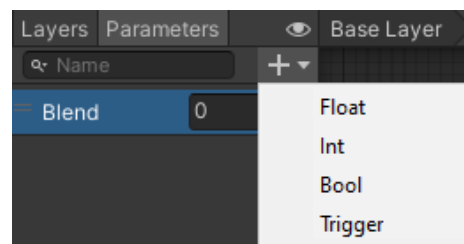
Gambar 9. 17 Masuk Kedalam Animator

18. Setelah itu klik kiri pada Player_idle lalu klik Make Transition arahkan kedalam Player_run



Gambar 9. 18 Transition Player_idle

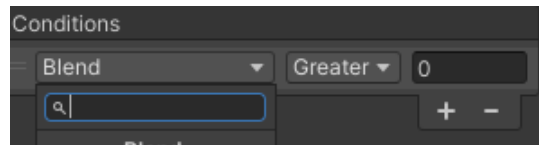
19. Setelah itu masuk kedalam tab parameter pada jendela Animator, lalu klik + dengan tipe data float, ubah Namanya menjadi Blend



Gambar 9. 19 Membuat Tipe Data Baru

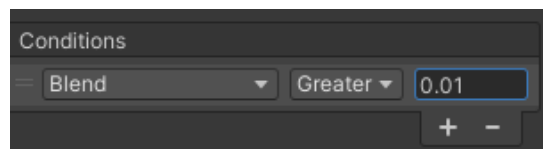


20. Setelah itu masuk kedalam Layers lalu klik pada panah warna putih, pada bagian conditions



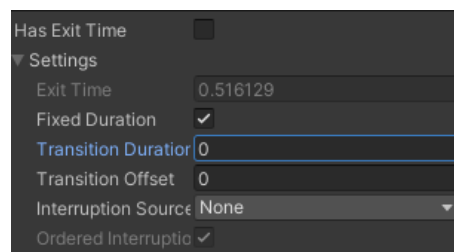
Gambar 9. 20 Bagian Conditions

21. Ubah greater menjadi 0.01



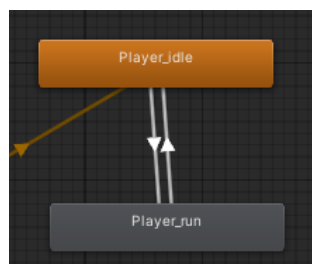
Gambar 9. 21 Kondisi 0.01

22. Setelah itu masuk kedalam setting inspector centang pada bagian Has Exit Time
Time atur bagian Transition duration menjadi 0



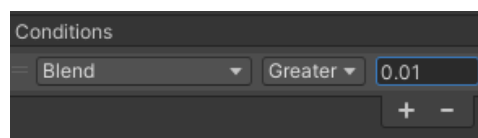
Gambar 9. 22 Setting Inspector

23. Buat transisi baru dari player run_ke player idle



Gambar 9. 23 Transisi baru Player_run

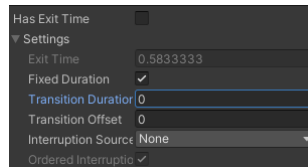
24. Setelah itu dengan cara yang sama klik bagian panah lalu atur bagian conditions, klik + lalu atur ke 0.01 greater



Gambar 9. 24 Mengatur Kondisi



25. Centang pada bagian Has Exit Time dan Transition Duration menjadi 0



Gambar 9. 25 Has Exit Time

26. Setelah itu masuk kedalam source code Player.cs, ubah pada bagian bawah class Player tambahkan kode program berikut

```
public Animator animator;
```

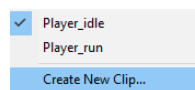
27. Setelah itu tambahkan kode berikut pada method Awake()

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

28. Tambahkan kode berikut pada method fixedUpdate()

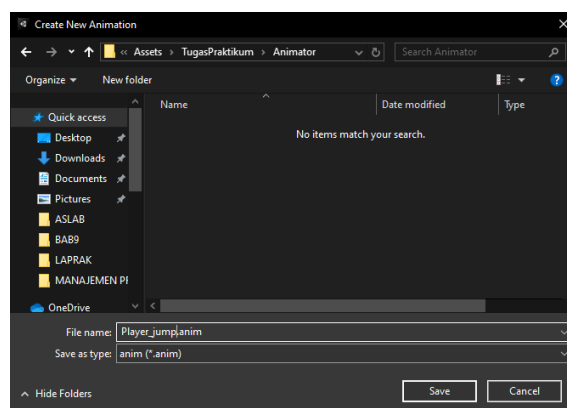
```
void FixedUpdate()  
{  
    GroundCheck();  
    Move(horizontalValue, jump);  
    animator.SetFloat("Blend",  
    Mathf.Abs(rb.velocity.x));  
}
```

29. Setelah itu masuk kedalam jendela animation pada Idle (1), lalu buat variabel baru yuntuk enangani animasi loncat



Gambar 9. 26 Membuat Animasi Baru

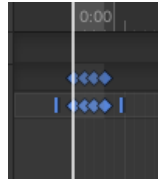
30. Beri nama Player_jump.anim



Gambar 9. 27 Membuat Animasi Baru

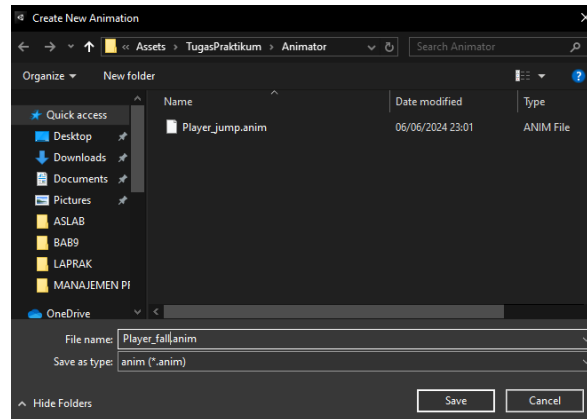


31. Setelah itu drag drop gambar Dead(2) kedalam animasi



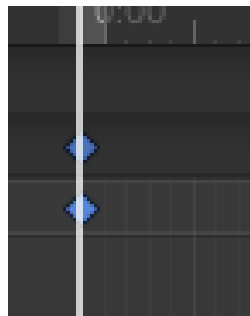
Gambar 9. 28 Membuat Animasi Jump

32. Setelah itu buat animasi baru dengan nama Player_fall



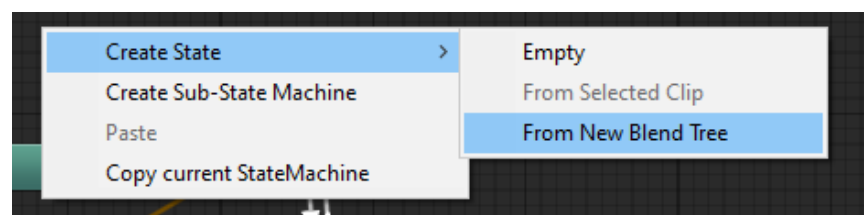
Gambar 9. 29 Animasi Player Fall

33. Lalu drag and drop gambar Dead(3) kedalam player fall



Gambar 9. 30 Membuat Animasi Player Fall

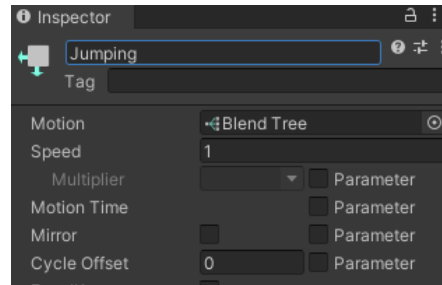
34. Masuk kedalam tab Animator setelah itu klik kanan Create State->From New Blend Tree



Gambar 9. 31 Membuat State Baru

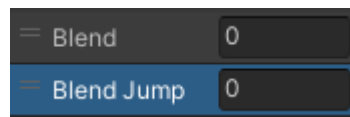


35. Pada animator blend tree yang baru ubah nama menjadi Jumping



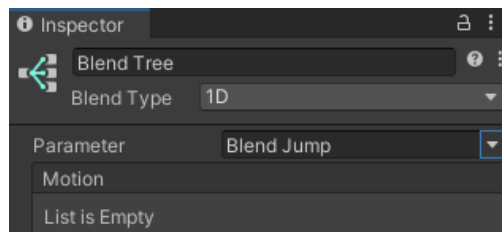
Gambar 9. 32 Membuat Blend Tree

36. Tambahkan parameter baru dengan nama Blend Jump dengan tipe data Float



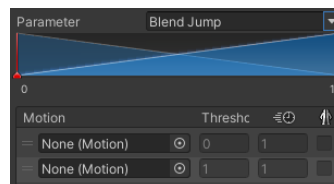
Gambar 9. 33 Parameter Blend Jump

37. Klik dua kali pada Tree Jumping lalu atur Blend Tree, setelah itu pada inspector ubah parameter Blend Tree menjadi Blend Jump



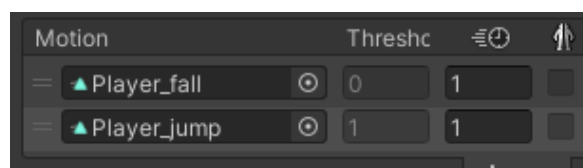
Gambar 9. 34 Mengubah Parameter Blend Tree

38. Klik icon + dibawah List Empty lalu buat dua motion baru



Gambar 9. 35 Membuat Motion Field

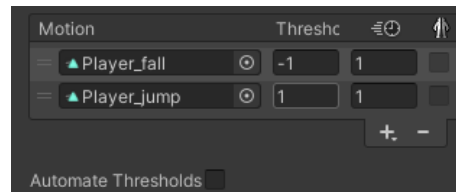
39. Ubah bagian Motion menjadi Motion Fall dan Motion Jump



Gambar 9. 36 Motion Fall dan Motion Jump

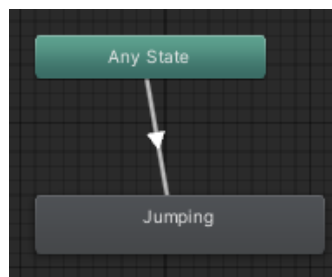


40. Atur threshold seperti gambar 9.40 jangan lupa untuk hapus centang Automate Threshold



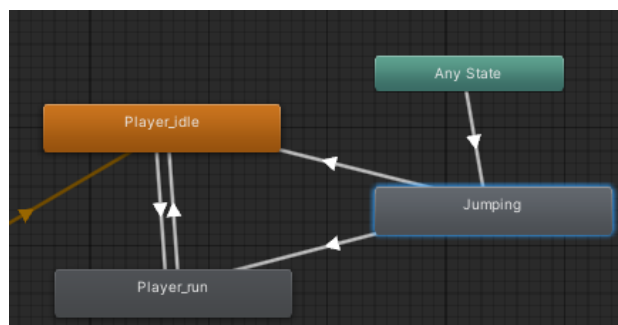
Gambar 9. 37 Mengatur Thresold

41. Masuk kedalam Base Layer pada Tree Any State klik kanan->make transition lalu arahkan panah tersebut kedalam Tree Jumping



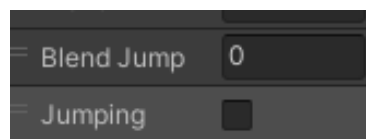
Gambar 9. 38 Mengatur State Jumping

42. Kembali klik kanan pada Tree Jumping->make transition lalu arahkan kedalam Player Idle dan Player Run



Gambar 9. 39 State Jumping

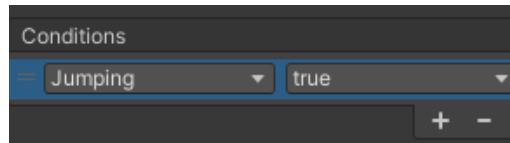
43. Tambah parameter bernama Jumping dengan tipe data Boolean ubah nilainya menjadi true



Gambar 9. 40 Parameter Jumping

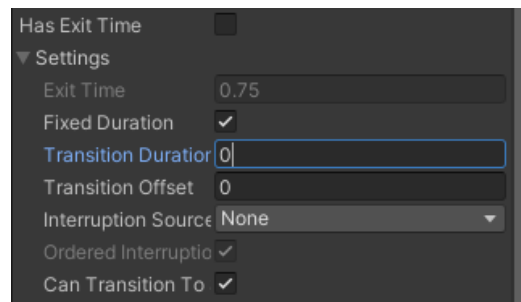


44. Untuk melakukan kondisi pada posisi jumping klik kana panah yang mengarah pada Tree Jumping, tambahkan kondisi Jumping dan set menjadi true



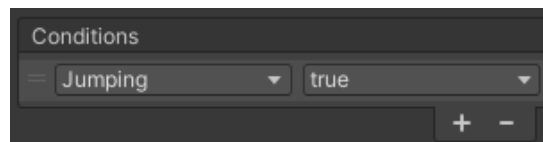
Gambar 9. 41 Kondisi Jumping

45. Pada transition duration ubah menjadi 0



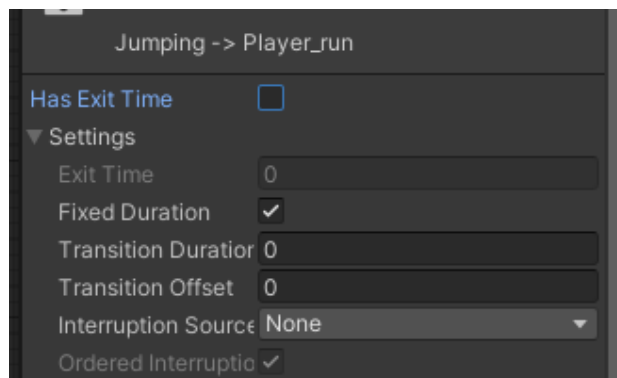
Gambar 9. 42 Mengatur Transition Duration

46. Klik panah yang mengarah pada Player_run ubah tambahkan kondisi dan ubah menjadi Jumping dan set nilainya menjadi true



Gambar 9. 43 Kondisi Jumping Player_run

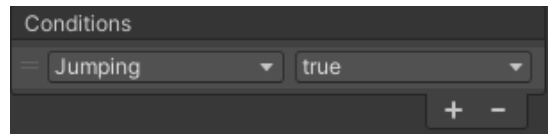
47. Setelah itu masuk pada menu setting ubah pada Transition Duration ubah menjadi 0 dan Has Exit Time jangan dicentang



Gambar 9. 44 Setting Jumping Player_run

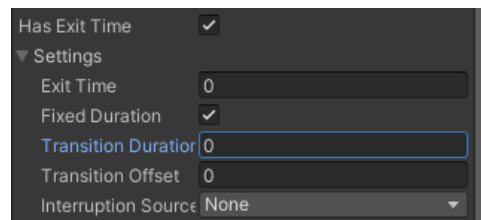


48. Klik pada panah yang mengarah pada Player_idle lalu buat kondisi baru, ubah kondisinya menjadi Jumping dan set nilainya menjadi true



Gambar 9. 45 Kondisi Jumping Player_idle

49. Pada bagian Has Exit Time hilangkan centang dan ubah pada transition duration menjadi 0



Gambar 9. 46 Has Exit Time

50. Buka kode program Player.cs pada method Update() ubah dan tambahkan kode berikut

```
if (Input.GetButtonDown("Jump"))
    animator.SetBool("Jumping", true);
    jump = true;
else if (Input.GetButtonUp("Jump"))
    jump = false;
```

51. Pada method FixedUpdate() ubah dan tambahkan kode berikut

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

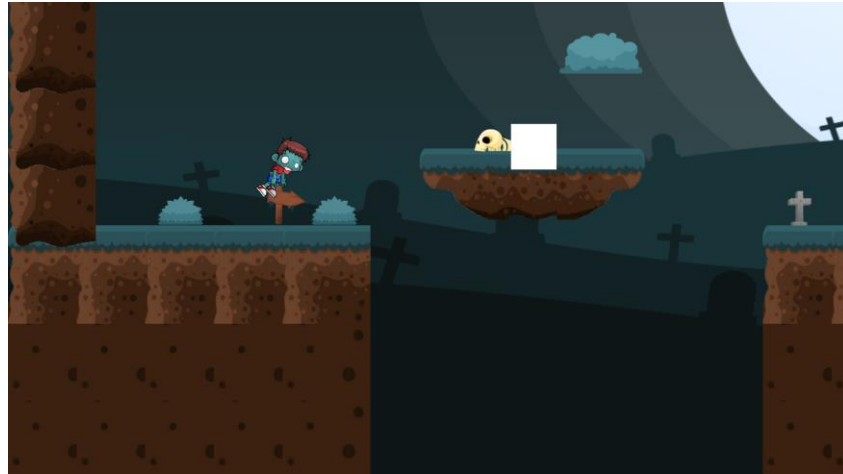
52. Pada method GroundCheck() ubah dan tambahkan kode berikut

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

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



53. Hasil Akhir



Gambar 9. 47 Hasil Akhir

Link Github Pengumpulan

Link : https://github.com/arifmalabar/2118055_PRAK_ANIGAME.git