# 10 Respawn and AI Enemy Attack

|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118145 |
| **Nama** | : | Esa Arya Mahardika |
| **Kelas** | : | D |
| **Asisten Lab** | : | 2118004 - Bagas Anardi Surya W. |
| **Baju Adat** | : | Banjar Babaju Kubaya Panjang |
| **Referensi** | : | https://berita.99.co/wp-content/uploads/2023/08/banjar-babaju-kubaya-panjang.jpg |

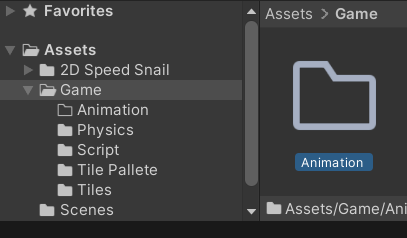
## Tugas 10: Menggerakkan Karakter dan Kamera

1. Buka proyek sebelumnya



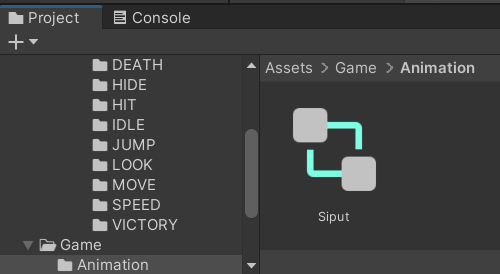
### 10.1 Proyek Sebelumnya

1. Buat folder baru dengan nama ‘Animation’



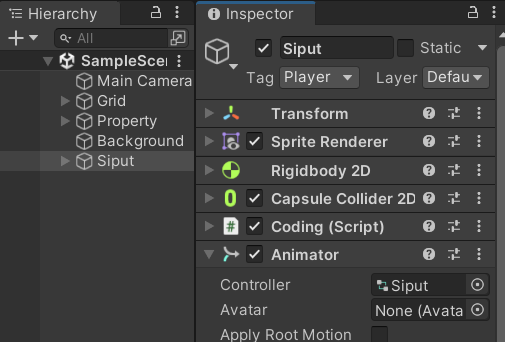
### 10.2 Komponen Karakter

1. Buat *Animation Controller* baru di dalam *folder* Animation



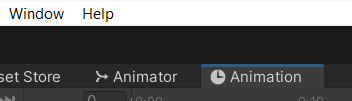
### 10.3 Membuat Animation Controller

1. Tambahkan komponen *Animator* pada Siput dan ubah *controller* menjadi Siput seperti yang sudah dibuat sebelumnya



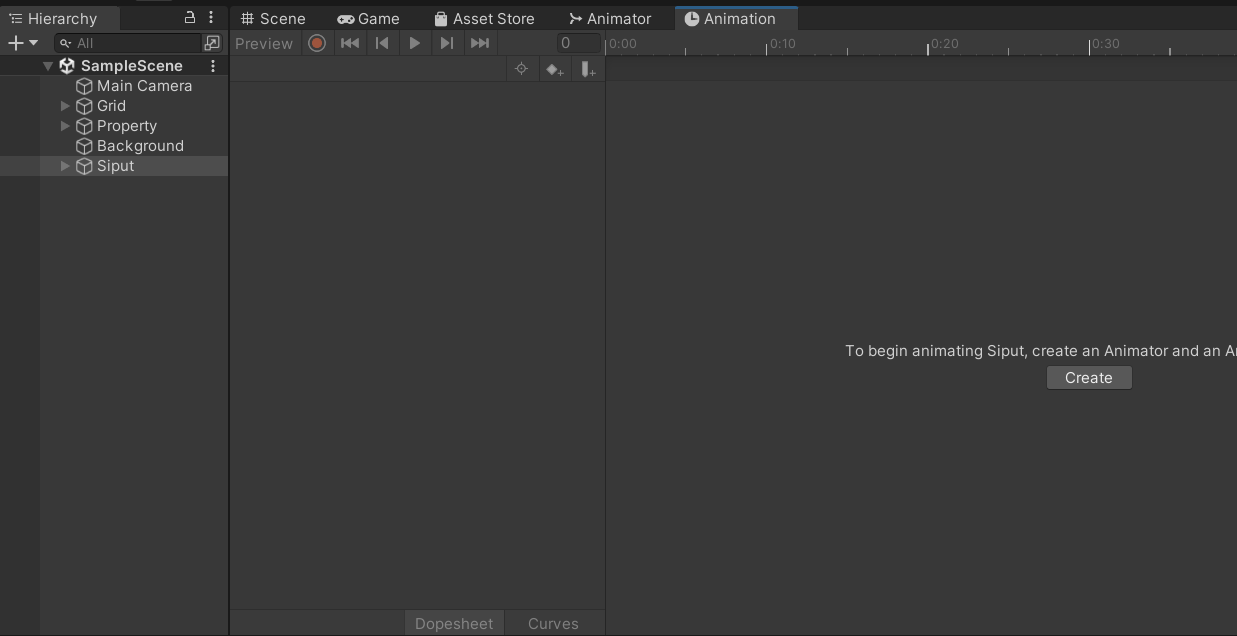
### 10.4 Menambahkan Komponen *Animator*

1. Siapkan halaman kerja *Animation* dan *Animator* di Menu *Window*



### 10.5 Menyiapkan Jendela *Animation* dan *Animator*

1. Pilih hierarki Siput dan buka halaman kerja *Animation*, tekan ‘*Create’*



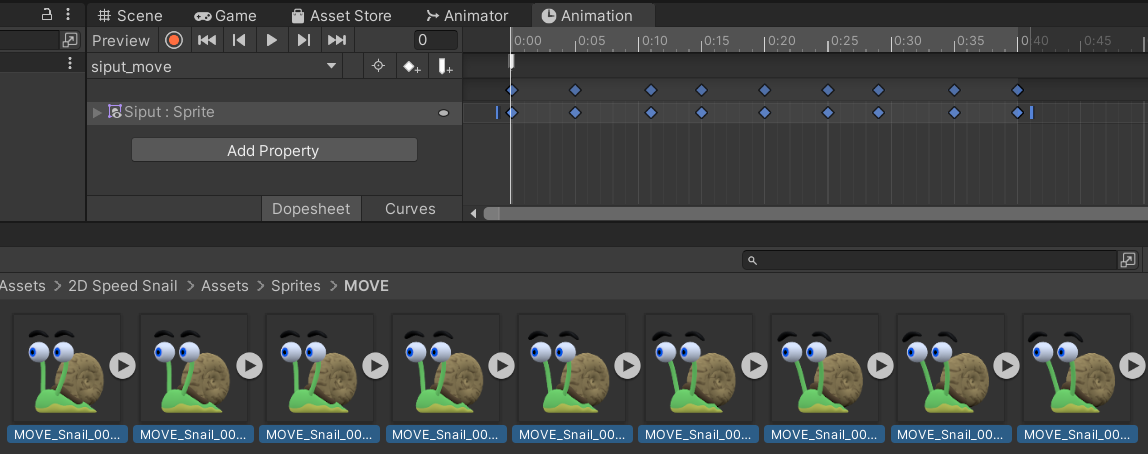
### 10.6 Membuat *Animation* Baru

1. Beri nama ‘siput\_idle’ dan *drag drop* aset IDLE\_Snail ke halaman kerja *Animation.* Beri rentang hingga 0:40



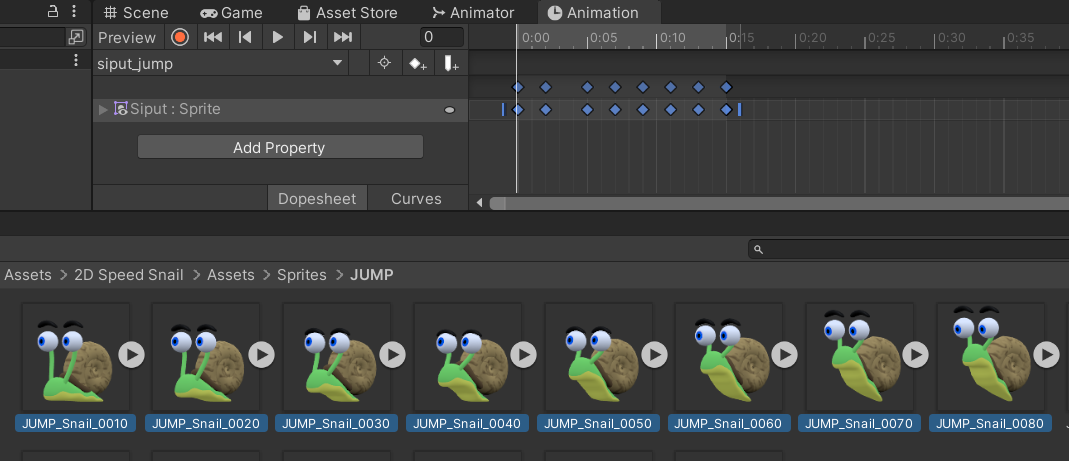
### 10.7 Membuat Animasi *Idle* Siput

1. Buat *clip* baru dengan nama ‘siput\_move’ dan *drag drop* aset MOVE\_Snail ke halaman kerja *Animation.* Beri rentang hingga 0:40



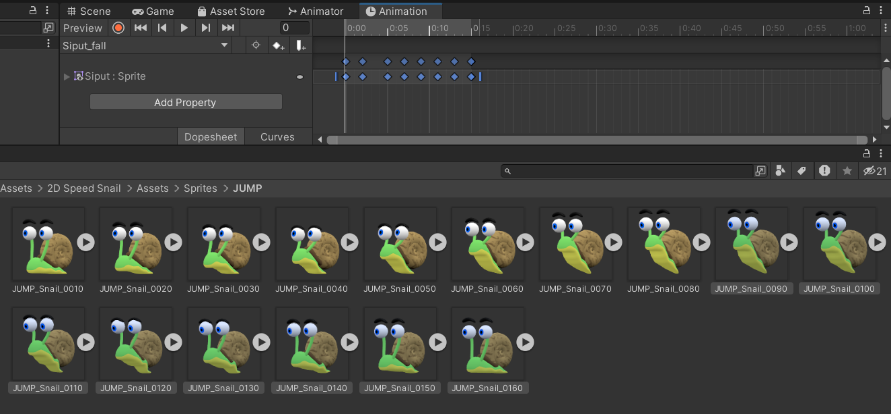
### 10.8 Membuat Animasi Siput Bergerak

1. Buat *clip* baru dengan nama ‘siput\_jump’ dan *drag drop* aset JUMP\_Snail dari 0010 hingga 0080 ke halaman kerja *Animation.* Beri rentang hingga 0:15



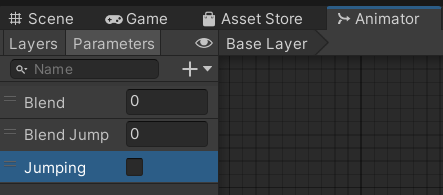
### 10.9 Membuat Animasi Siput Lompat

1. Buat *clip* baru dengan nama ‘siput\_fall’ dan *drag drop* aset JUMP\_Snail dari 0090 hingga 00160 ke halaman kerja *Animation.* Beri rentang hingga 0:15



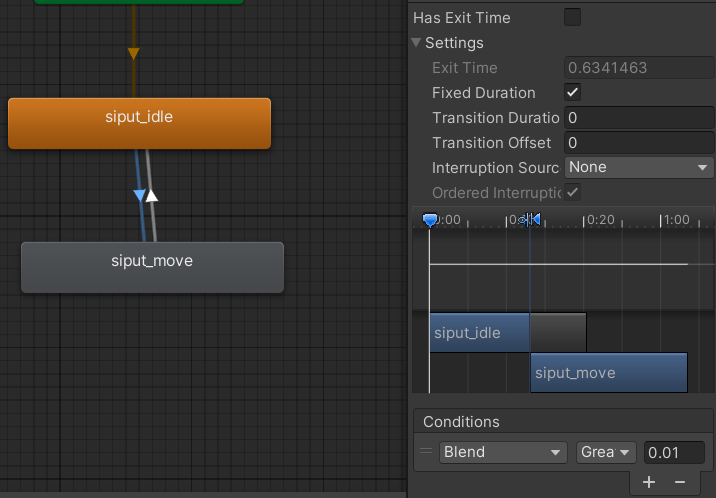
### 10.10 Membuat Animasi Siput Jatuh

1. Buka halaman kerja *Animator* dan pindah ke menu *Parameter*. Buat variabel ‘Blend’ dan ‘Blend Jump’ dengan tipe data *float* sedangkan ‘Jumping’ dengan tipe data *bool*



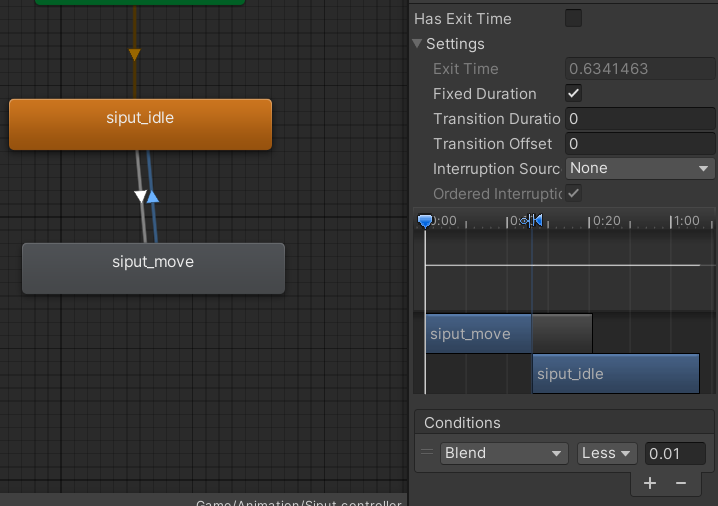
### 10.11 Membuat Parameter di Animator

1. Buat garis antara siput\_idle dan siput\_move dengan klik kanan ke siput\_idle, pilih ‘Make Transition’ dan arahkan ke siput\_move. Klik garis arah bawah dan sesuaikan konfigurasinya seperti gambar di bawah



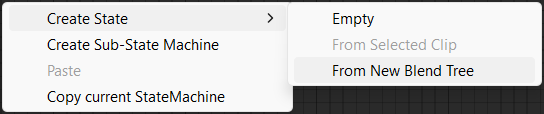
### 10.12 Konfigurasi *Transition* siput\_idle ke siput\_move

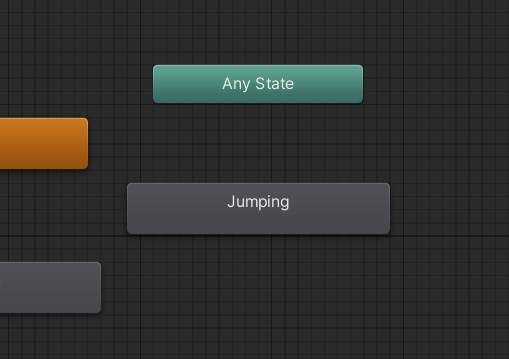
1. Buat garis sebaliknya dengan cara yang sama seperti sebelumnya dan sesuaikan konfigurasi seperti gambar di bawah



### 10.13 Konfigurasi *Transition* siput\_move ke siput\_idle

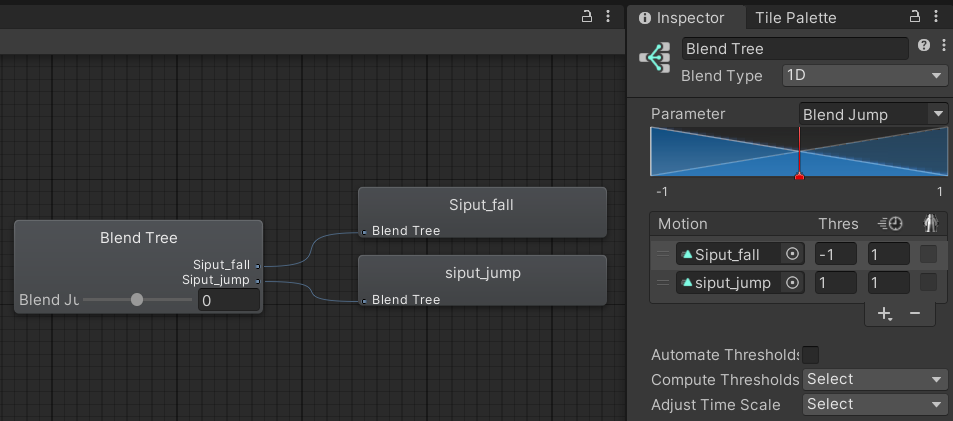
1. Buat *State* baru dengan cara klik kanan pada halaman kerja *Animator* dan beri nama ‘Jumping’





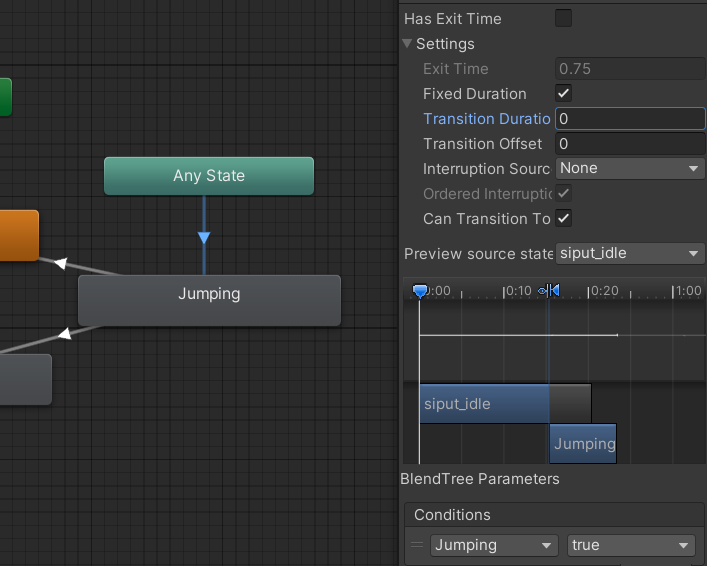
### 10.14 Membuat *State* Baru

1. Klik 2x *state* Jumping, klik *Blend Tree* dan atur konfigurasi *Blend Tree* di *inspector* seperti di gambar



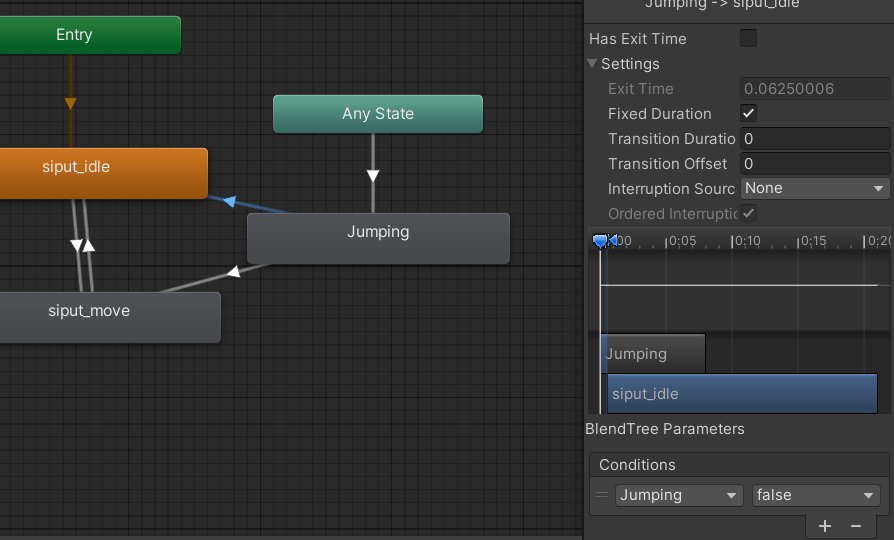
### 10.15 Uji Coba Lompat

1. Kembali ke pengaturan awal dan hubungkan Any State dengan Jumping. Klik pada garis panah dan sesuaikan konfigurasinya seperti gambar di bawah



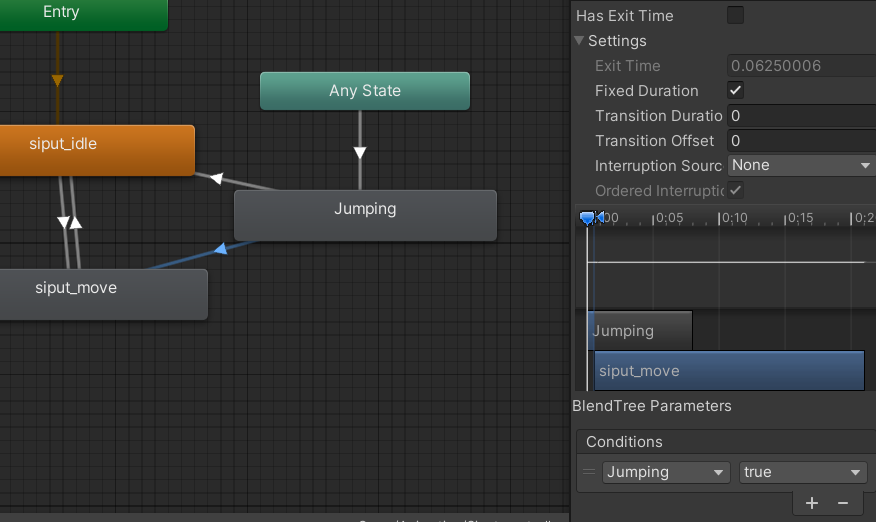
### 10.16 Mengatur Konfigurasi *Transition* *Any State* ke Jumping

1. Buat transition Jumping ke siput\_idle dan atur konfigurasinya seperti gambar di bawah



### 10.17 Mengatur Konfigurasi *Transition* Jumping ke siput\_idle

1. Buat transition Jumping ke siput\_move dan atur konfigurasinya seperti gambar di bawah



### 10.18 Menyisipkan CameraTrack ke Kamera

1. Perbarui kode Coding seperti di bawah ini. Hanya beberapa function saja yang perlu diganti

|  |
| --- |
| public Animator animator;  private void Awake(){          rb = GetComponent<Rigidbody2D>();          animator = GetComponent<Animator>();}      void Update (){  horizontalValue = Input.GetAxisRaw("Horizontal");          if (Input.GetButtonDown("Jump")){              jump = true;              animator.SetBool("Jumping", 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.position, groundCheckRadius, groundLayer);          if (colliders.Length > 0){ isGrounded = true;  }animator.SetBool("Jumping", !isGrounded);} |

1. Jalankan proyek dan coba menggerakkan siput agar animasi bekerja



### 10.19 Uji Coba Animasi Siput

1. **Link Github Pengumpulan**

<https://github.com/EsaAryaMahardika/PraktikumAnimasiGame/tree/main/Pertemuan%209>

1. **Kuis**

|  |
| --- |
| 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);      }  } |

Analisis

Kesalahan kode di atas adalah:

1. Pada percabangan *if* pertama seharusnya (move = 0) karena isi percabangan ini adalah kondisi karakter tidak bergerak dengan bukti adanya kode (animator.SetBool("isIdle", true);)
2. Kode transform.Translate(Vector3.left\*move\*Time.deltaTime seharusnya berada di luar percabangan (move = 0) karena kode tersebut berfungsi untuk menjalankan karakter
3. Pada percabangan *if* kedua seharusnya (move < 0) karena *else* di bawah menandakan (move > 0) maka sebelumnya pasti sebaliknya
4. Seharusnya karakter mempunyai ukuran yang sama saat menghadap ke kanan maupun kiri