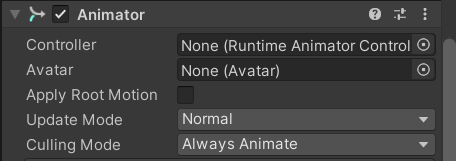
# 9 GAME ANIMATION

|  |  |  |
| --- | --- | --- |
| **NIM** | : | 2118126 |
| **Nama** | : | Prita Patricia Lakzmi |
| **Kelas** | : | D |
| **Asisten Lab** | : | Wisando Berlian P. (2218095) |

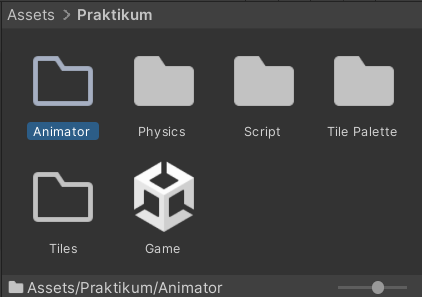
## Tugas 9 : Membuat Character Game Animation

1. **Pergerakan Player**
2. Buka *Project Unity* pada 2118126\_BAB8 sebelumnya yang telah dibuat sebelumnya. Lalu pada *Hierarchy player-id Add Component* dengan nama *Animator* pada *inspector*.

****

### *Add Component Animator*

1. Buat 1 buah *folder* baru dengan nama “*Animator*” di dalam *folder* Praktikum.

****

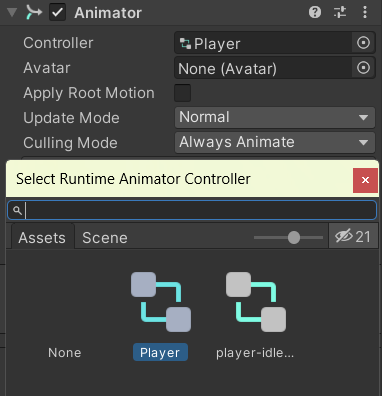
### Membuat *Folder* *Animator*

1. Buat 1 buah *file Animator Controller* dengan nama “*Player*” pada *folder* *Animator*.



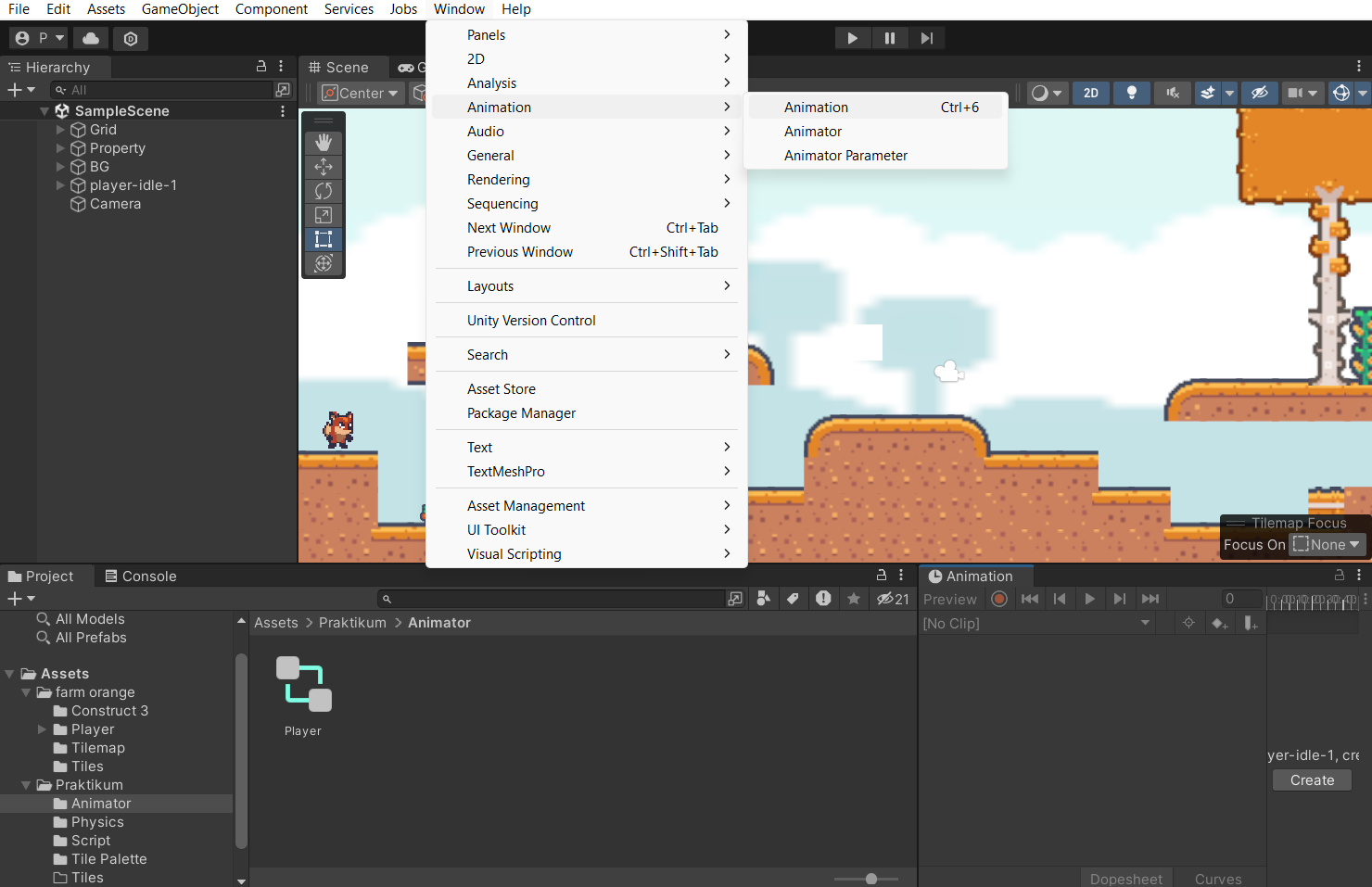
### Membuat *Animator Controller*

1. Pada *player-idle* di dalam *component Animator* ubah *Controller* menjadi *Player*.



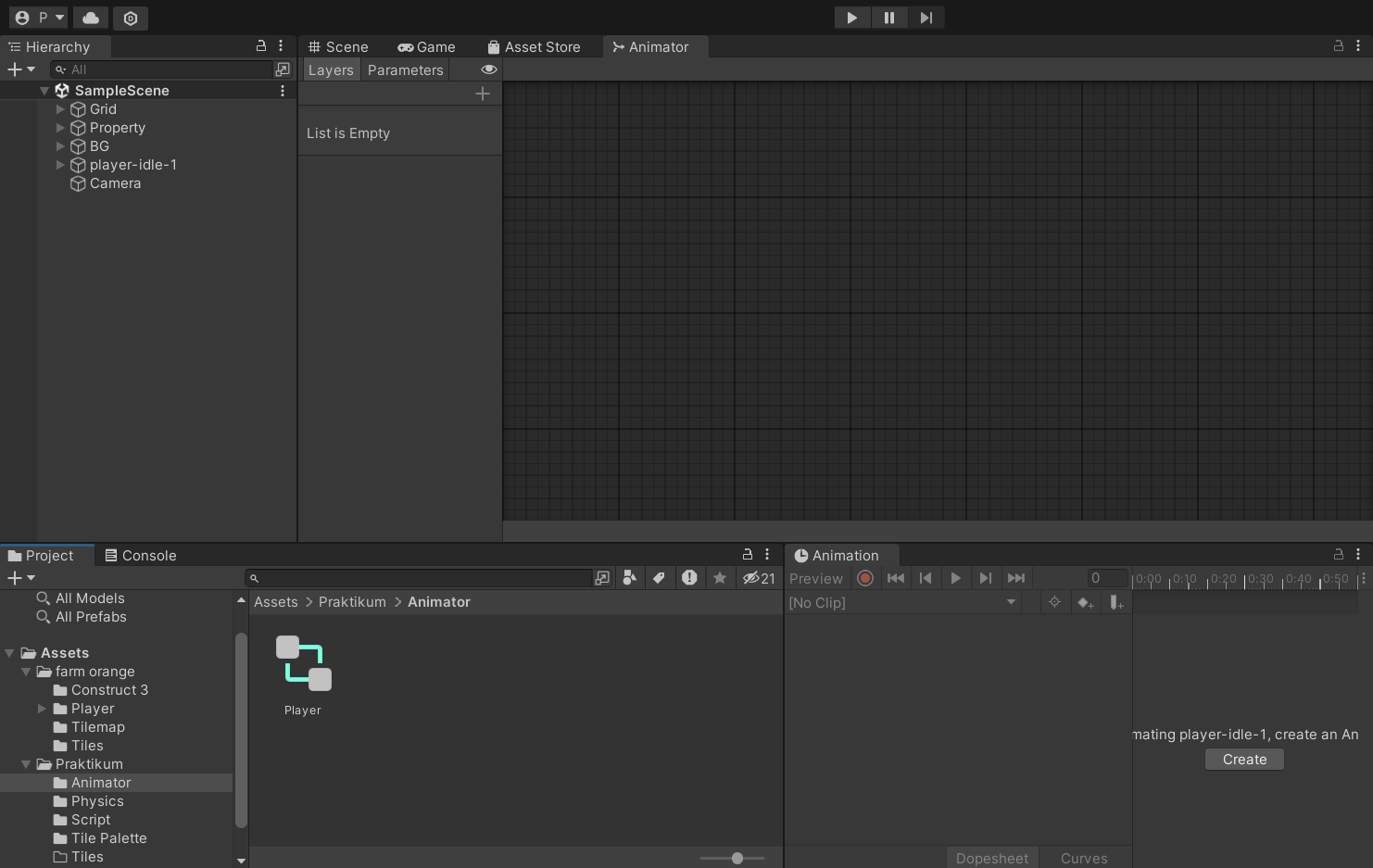
### *Select Player Controller*

1. Tambah menu panel *animator* dengan cara klik menu *Window* pilih *Animation* > *Animation*. Kemudian posisikan panel dengan posisi *split Panel Project* dan *Animation* bersampingan.



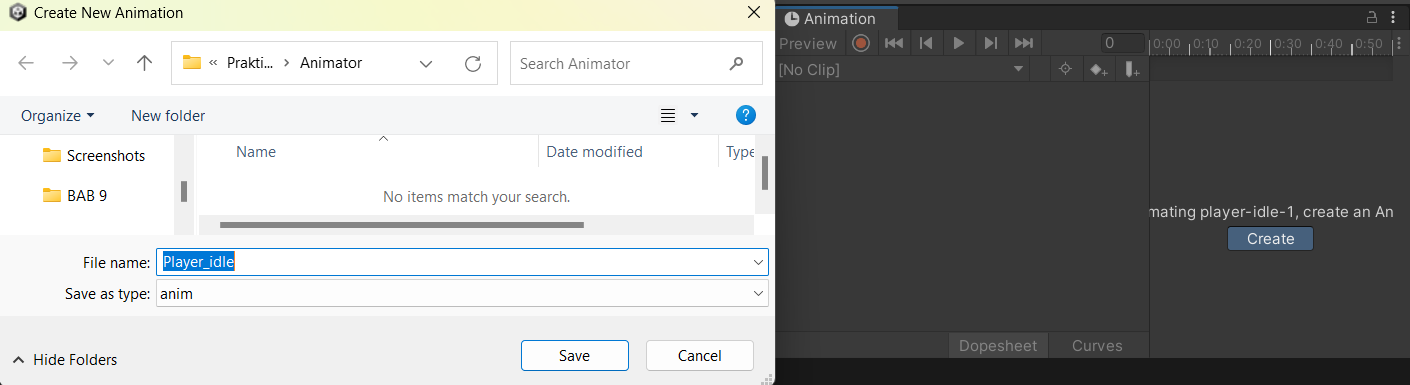
### Menambah Panel *Animation*

1. Tambahkan menu panel *Animator*, dengan cara menu *Window* pilih *Animation* > *Animator*.



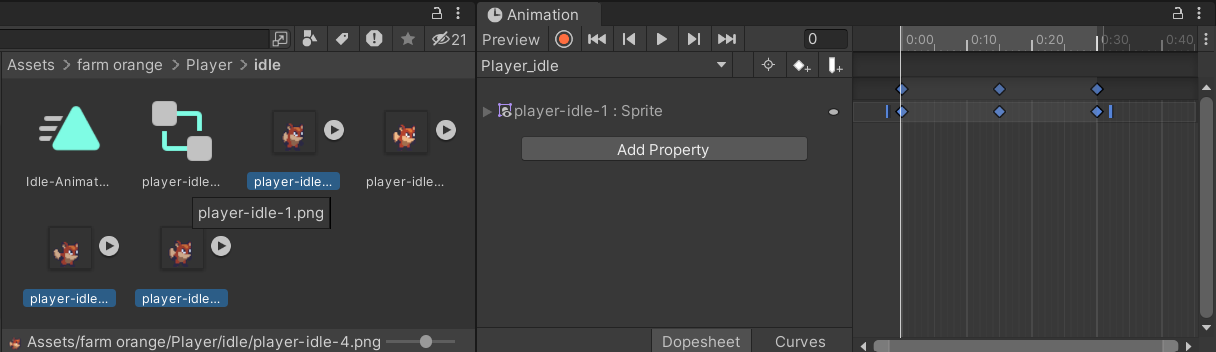
### Menambah Panel *Animator*

1. Buat animation pada *Hierarchy* *player-idle* dan pada panel *Animation* klik *button Create* dan beri nama “*Player\_idle*”.



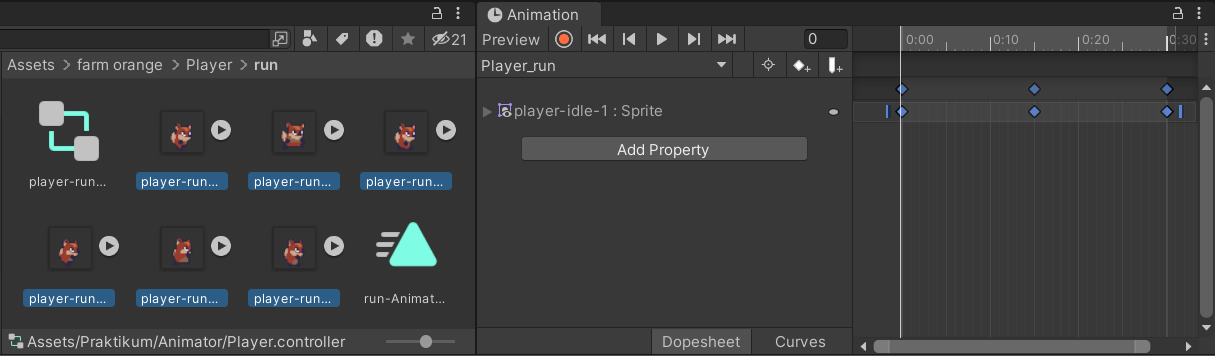
### *Create New Clip Player\_idle*

1. *Drag and drop player-idle* 1, 3, dan 4 pada *timeline frame* 0 hingga 30.



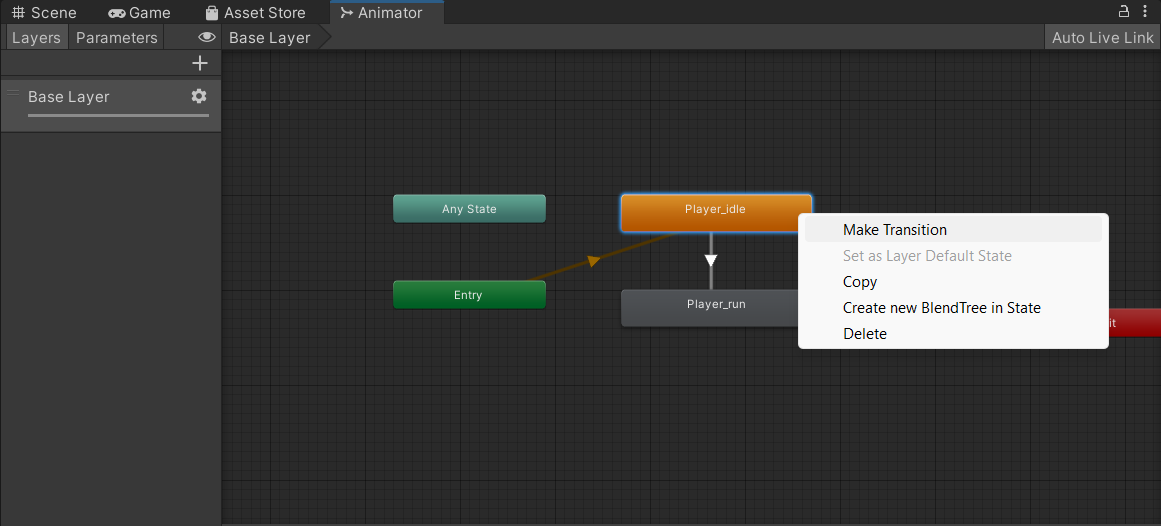
### *Drag And Drop Player\_idle* ke dalam *Frame*

1. Selanjutnya *drag and drop player-run* pada *timeline frame* 0 hingga 30.



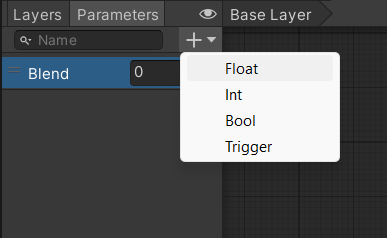
### *Drag And Drop Player\_run* ke Dalam *Frame*

1. Pada menu *Animator Make Transition pada Player\_idle ke Player\_run*.



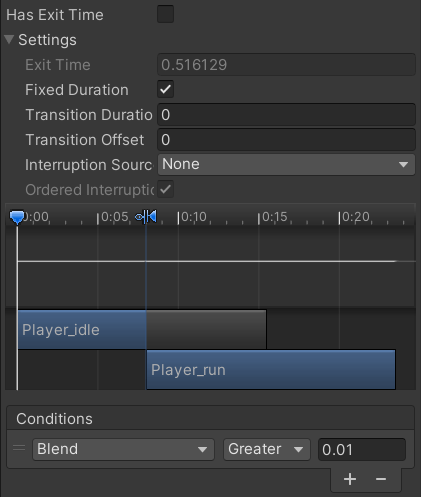
### *Make Transition Player\_idle ke Player\_run*

1. Masuk ke tab parameter, tambah 1 buah parameter dengan tipe data *Float* dan *rename* menjadi “*Blend*”.



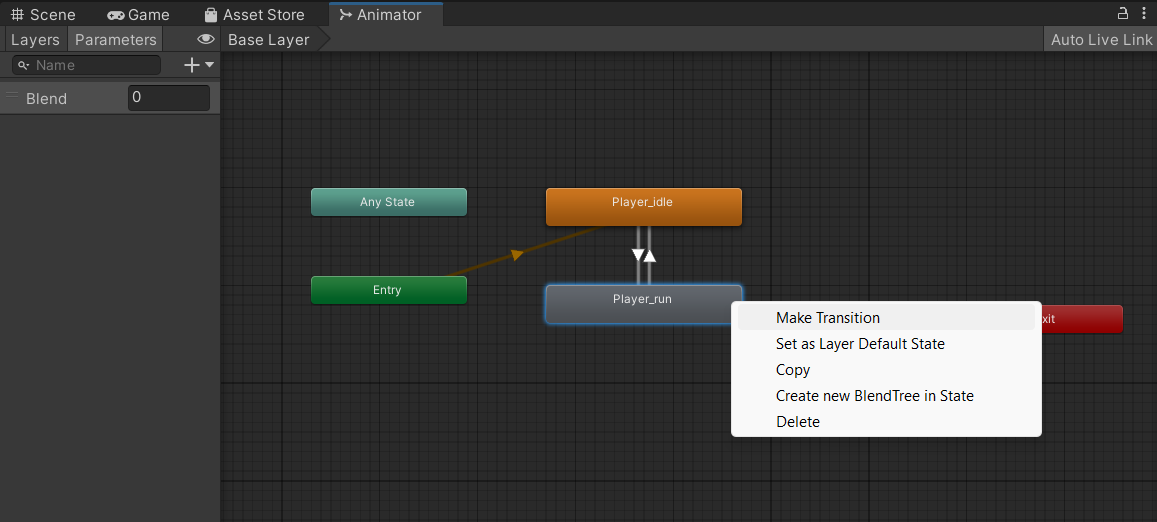
### *Parameter Blend*

1. Klik tanda panah *Make Transition* pada *Player\_idle* ke *Player\_run*. Kemudian pada bagian *inspector* tambah 1 *condtitions* dan ubah *conditions* menjadi *Blend*, atur nilai menjadi 0.01 dan sesuaikan *Settings* seperti gambar dibawah ini.



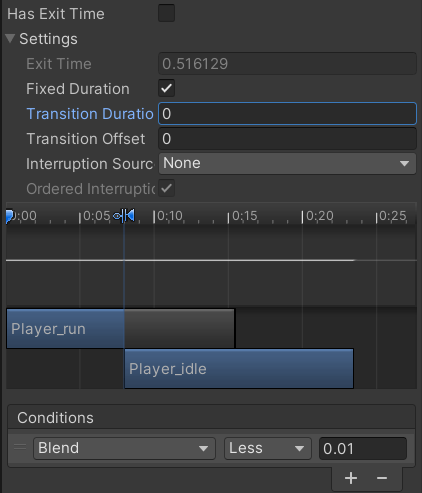
### *Inspector Player\_idle* ke *Player\_run*

1. *Make Transition* dari *Player\_run* ke *Player\_idle*.



### *Make Transition Player\_run ke Player\_idle*

1. Klik tanda panah *Make Transition* pada *Player\_run* ke *Player\_idle*. Kemudian pada bagian *inspector* tambah 1 *condtitions* dan ubah *conditions* menjadi *Blend*, atur nilai menjadi 0.01 dan sesuaikan *Settings* seperti gambar dibawah ini.



### *Inspector* *Player*\_*run* ke *Player\_idle*

1. Buka *script* *Player*, kemudian tambahkan *source code* di bawah ini. Letakkan dalam kelas *Player* dan paling atas.

|  |
| --- |
| public Animator animator; |

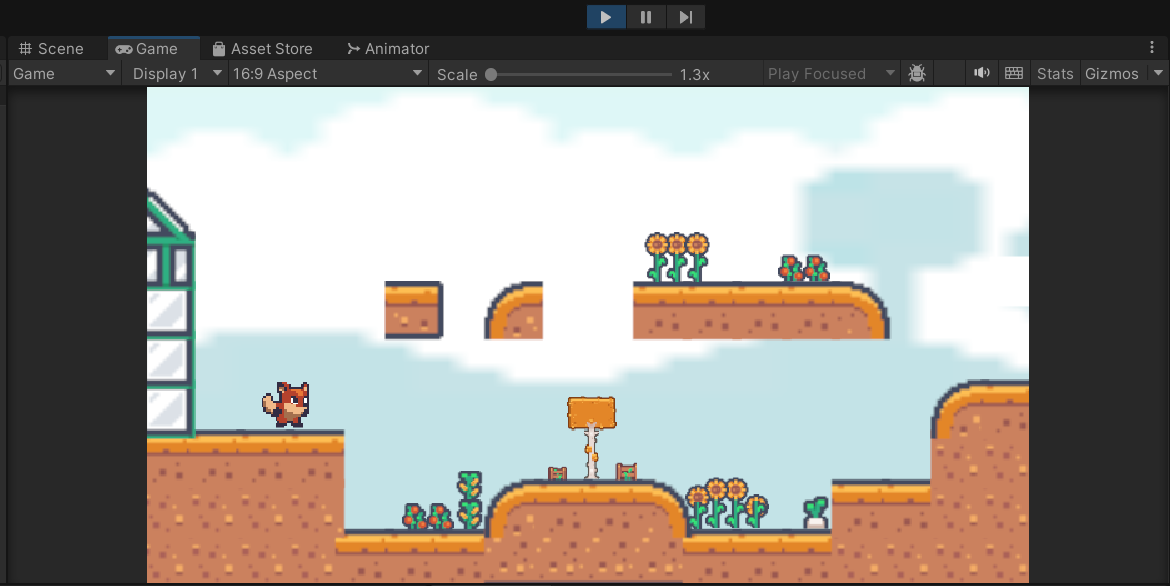
1. Tambahkan *script* komponen *Animator* pada *fungsi void Awake().*

|  |
| --- |
| animator = GetComponent<Animator>(); |

1. Dan pada fungsi *FixedUpdate* tambahkan *source code* berikut.

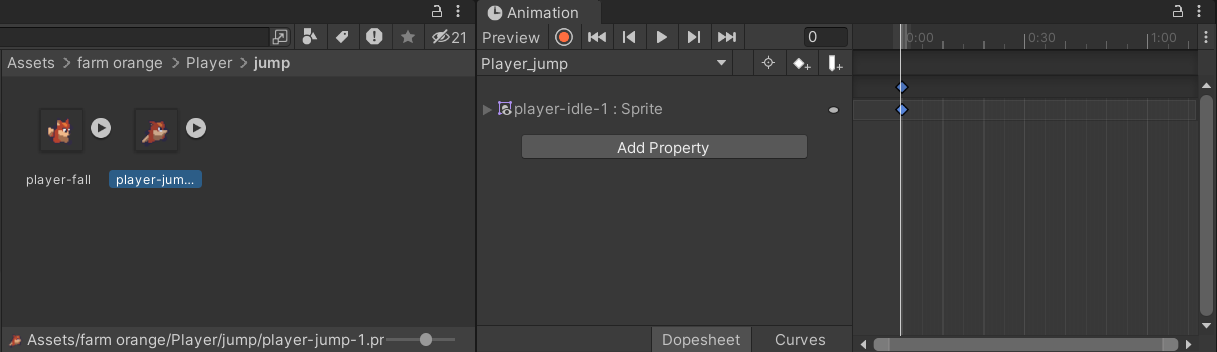
|  |
| --- |
| animator.SetFloat("Blend", Mathf.Abs(rb.velocity.x)); |

1. Tekan *play* untuk melihat apakah *player* telah memiliki animasi ketika *player* berhenti berjalan.



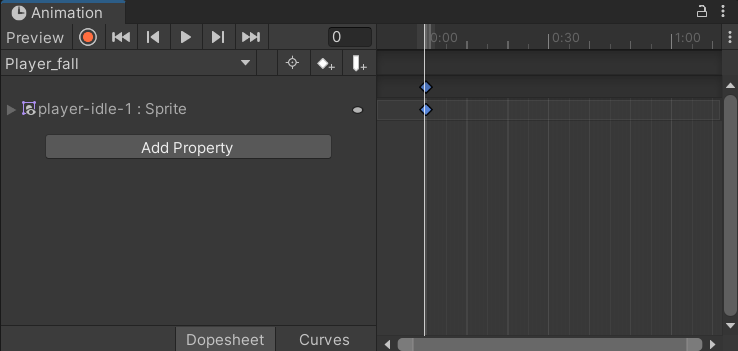
### Tampilan Saat Di *Play*

1. *Create New Clip* untuk membuat animasi *Player\_jump*. Kemudian *drag and drop player-jump* kedalam *timeline* *frame*.



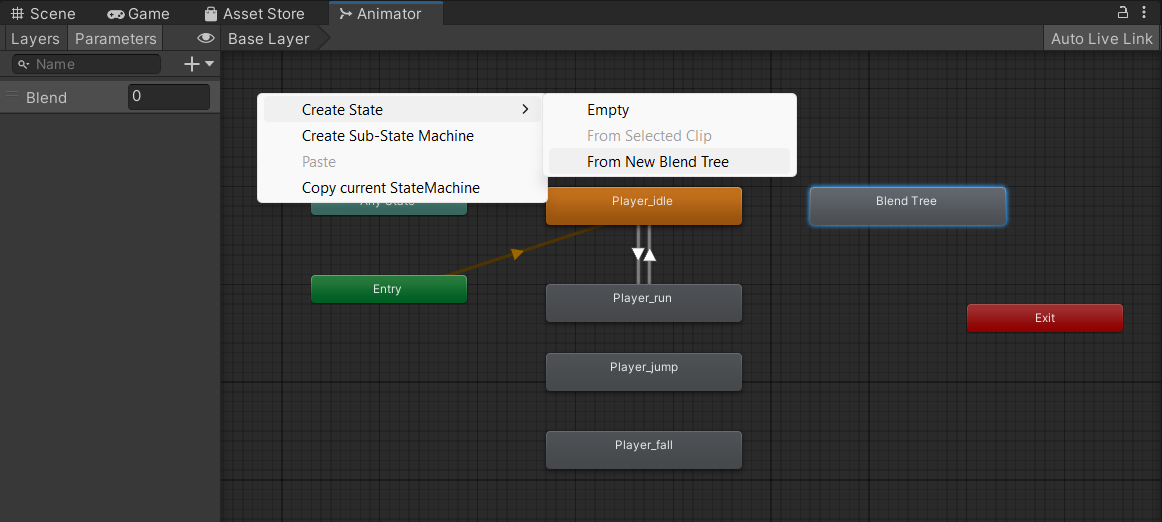
### *Create New Clip Player\_jump*

1. *Create New Clip* untuk membuat animasi *Player\_fall*. Kemudian *drag and drop player-fall* kedalam *timeline frame*.



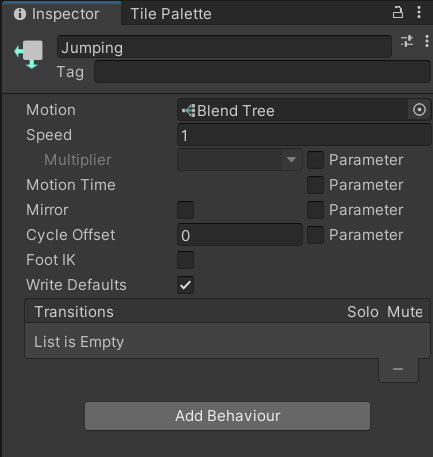
### *Create New Clip Player\_fall*

1. Kemudian untuk menambahkan animasi ketika melompat. Klik kanan pada menu *animator* pada area kosong dan pilih *Create State* > *From New Blend Tree.*



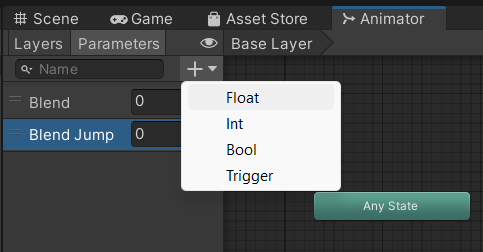
### Membuat *Blend Tree*

1. Pada panel *Animator* klik *Button Blend Tree*. Di dalam menu *inspector* ubah namanya menjadi *Jumping*.



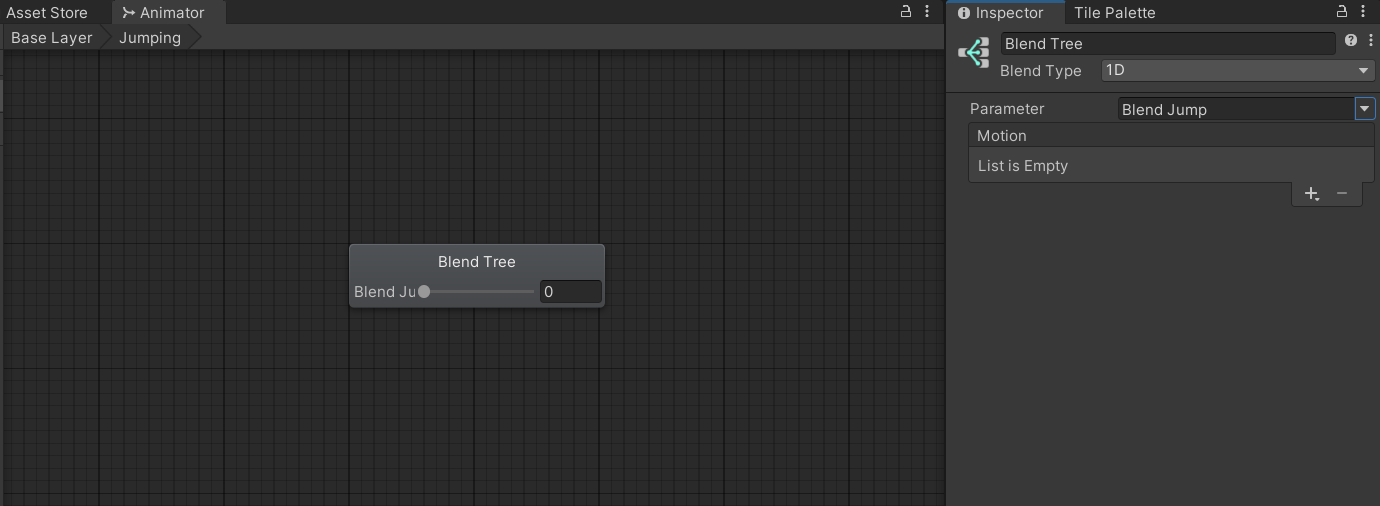
### *Inspector Blend Tree*

1. Buat 1 buah parameter baru dengan tipe data *Float* dan *rename* menjadi *Blend Jump*.



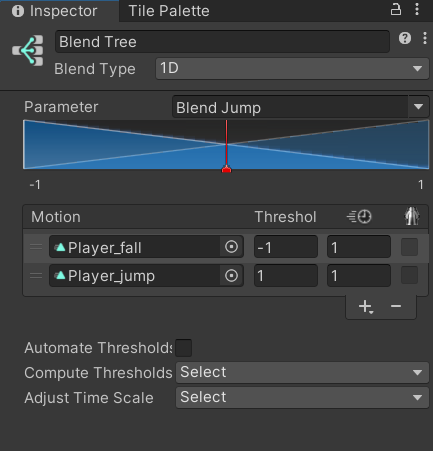
### Parameter *Blend Jump*

1. Klik 2x pada *Blend Tree* di dalam panel *Animator*. Jika sudah maka tampilannya akan seperti gambar dibawah ini.



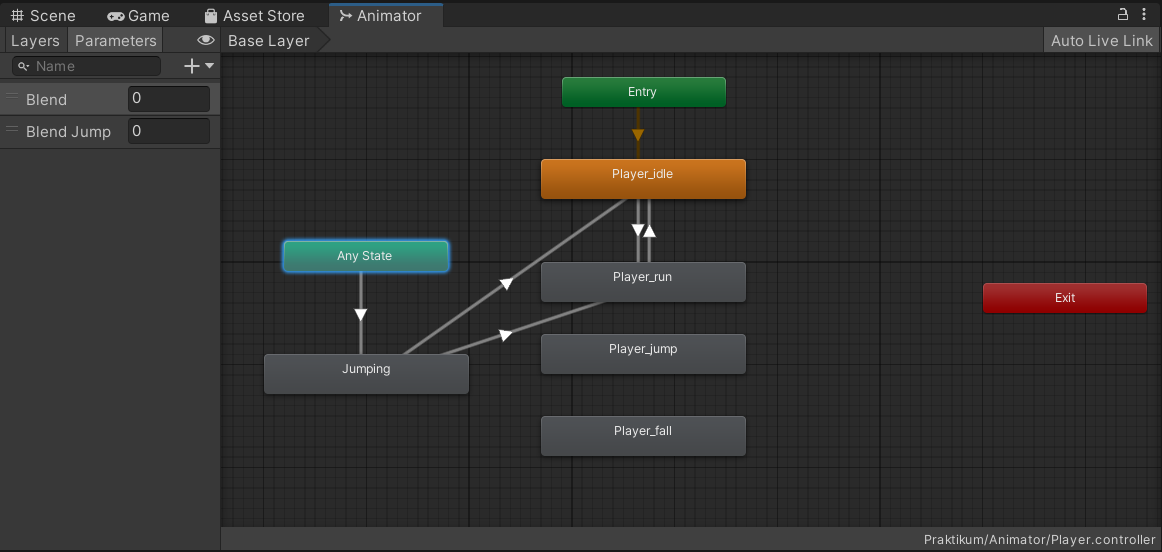
### *Blend Tree*

1. Pada inspector *Blend Jump* ubah parameter menjadi ‘*Blend Jump’*. Dan *Add Motion Field* sebanyak 2x lalu sesuaikan *motion* seperti gambar dibawah ini dan ubah motion menjadi *Player\_fall* dan *Player\_jump*.

****

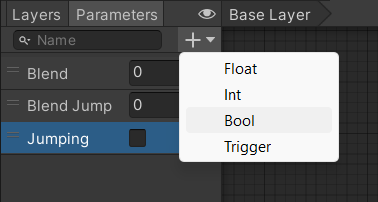
### *Inspector Blend Jump*

1. *Make Transition* pada *Any State* ke *Jumping*. Dan *Make Transition* dari *Jumping* ke *Player\_idle* dan *Player\_run*.

****

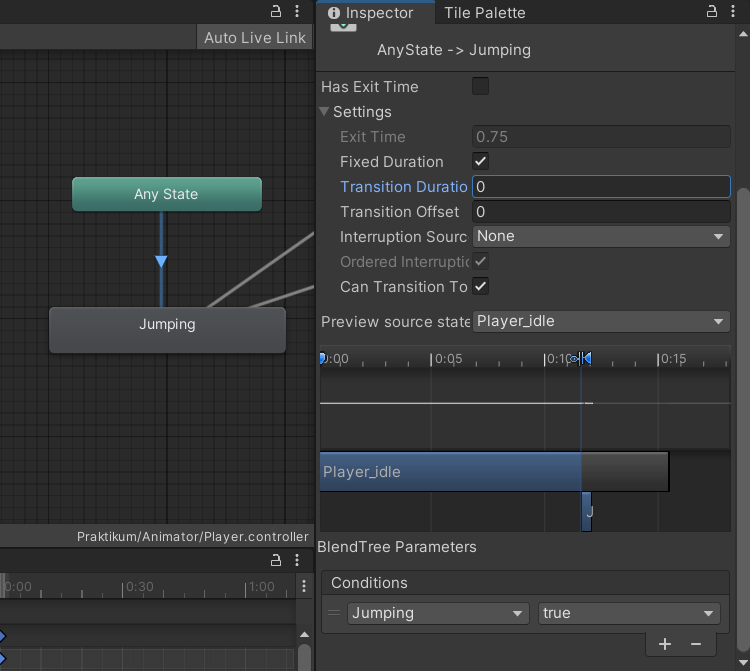
### *Inspector Any State* ke *Jumping*

1. Buat 1 buah parameter dengan tipe data *Bool* dan ubah namanya menjadi “*Jumping*”.

****

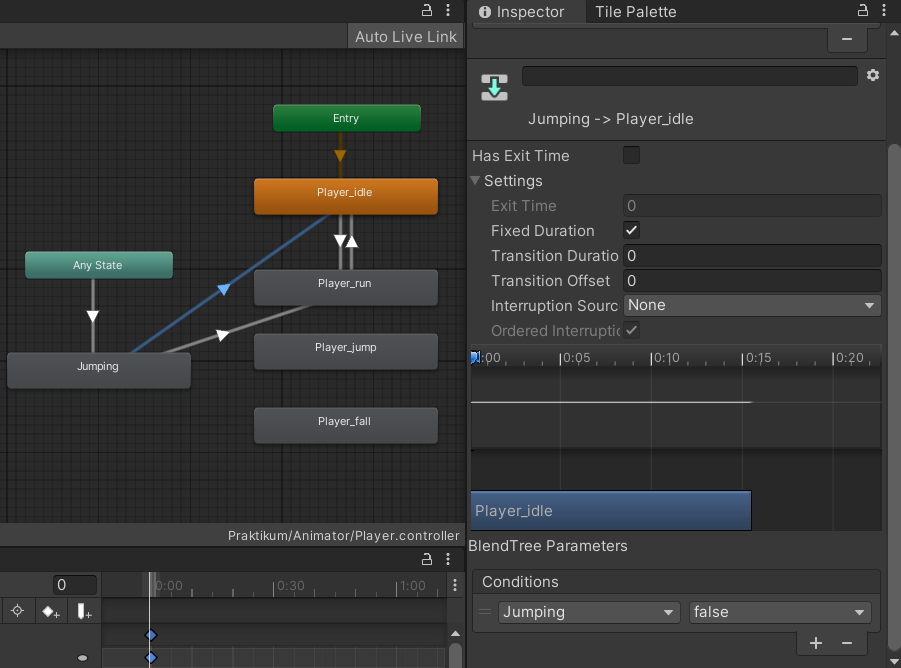
### Parameter *Jumping*

1. Klik tanda panah *Make Transitio*n dari *Any State* ke *Jumping*. Dan tambahkan *Condiotion* *Jumping* pada *inspector* dan ubah nilai nya menjadi *true*.

****

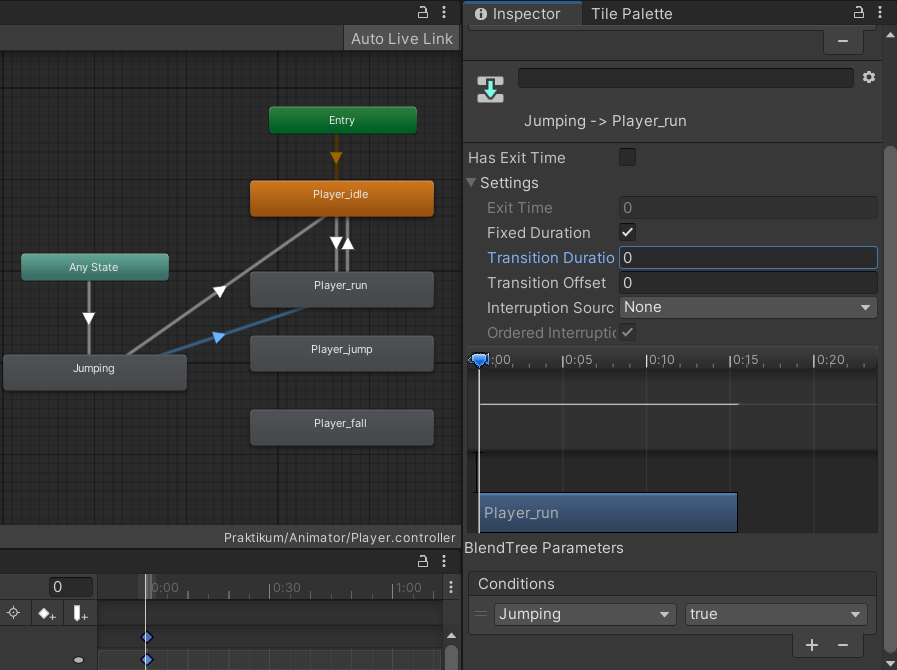
### *Inspector Any State* ke *Jumping*

1. Klik tanda panah *Make Transition* dari *Jumping* ke *Player\_idle*. Dan tambahkan *Condiotion* *Jumping* pada *inspector* dan ubah nilai nya menjadi *false*.

****

### *Inspector Jumping* ke *Player\_idle*

1. Klik tanda panah *Make Transition* dari *Jumping* ke *Player\_run*. Dan tambahkan *Condiotion* *Jumping* pada *inspector* dan ubah nilai nya menjadi *true*.

****

### *Inspector Jumping* ke *Player\_run*

1. Tambahkan *source code* yang berlabel kuning pada fungsi *update*.

|  |
| --- |
| void Update ()  {  horizontalValue = Input.GetAxisRaw("Horizontal");  if (Input.GetButtonDown("Jump")) {  animator.SetBool("Jumping", true);  jump = true;  } |

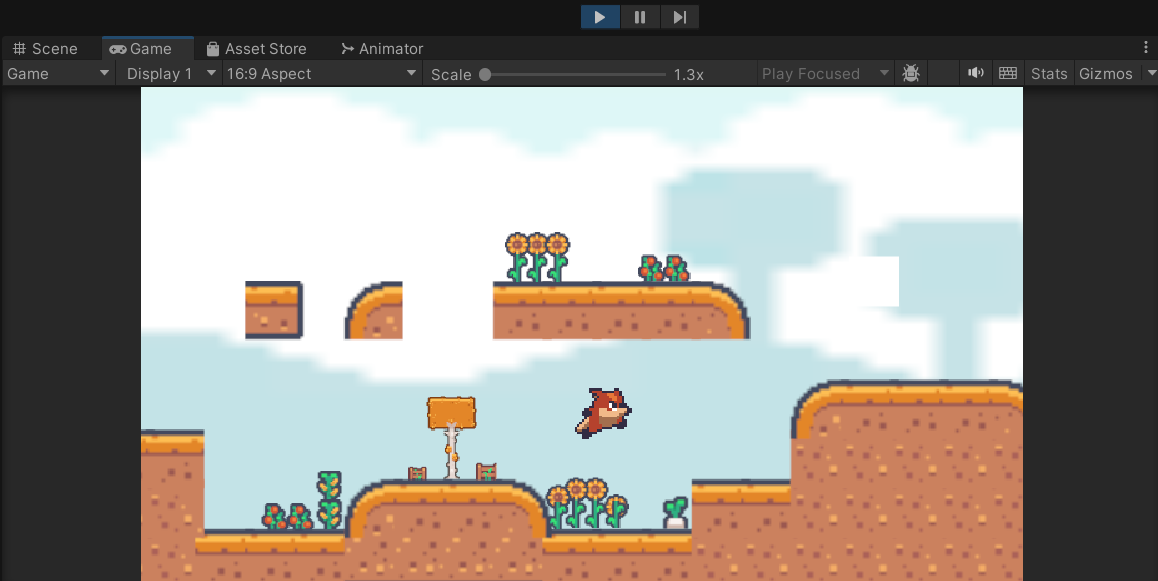
1. Pada fungsi *FixedUpdate* tambahkan *source code* yang dilabel kuning.

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

1. Tambahkan baris *code* yang berlabel kuning dalam *method* *GroundCheck*.

|  |
| --- |
| void GroundCheck()  {  isGrounded = false;  Collider2D[] colliders = Physics2D.OverlapCircleAll(groundcheckCollider.position, groundCheckRadius, groundLayer);  if (colliders.Length > 0) {  isGrounded = true;  }  animator.SetBool("Jumping", !isGrounded);  } |

1. Tekan *button play* untuk melihat apakah karakter sudah dapat bergerak dengan animasi.

****

### Tampilan Saat di *Play*

1. **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);  }  }  } |

Penjelasan :

Pada fungsi *HandleJumpInput*, *Boolean* diberi nilai *true* jika memenuhi kondisi yaitu ketika tombol *Space* ditekan, jika tidak maka diberi nilai *false*.

Memperbaiki logika di bagian fungsi *HandleMovementInput* agar *isIdle* diatur menjadi ‘*true’* hanya ketika ‘move == 0’. Dan menambahkan ‘animator.SetBool(“isWalking”, true)’ ketika ‘move’ tidak sama dengan 0.