

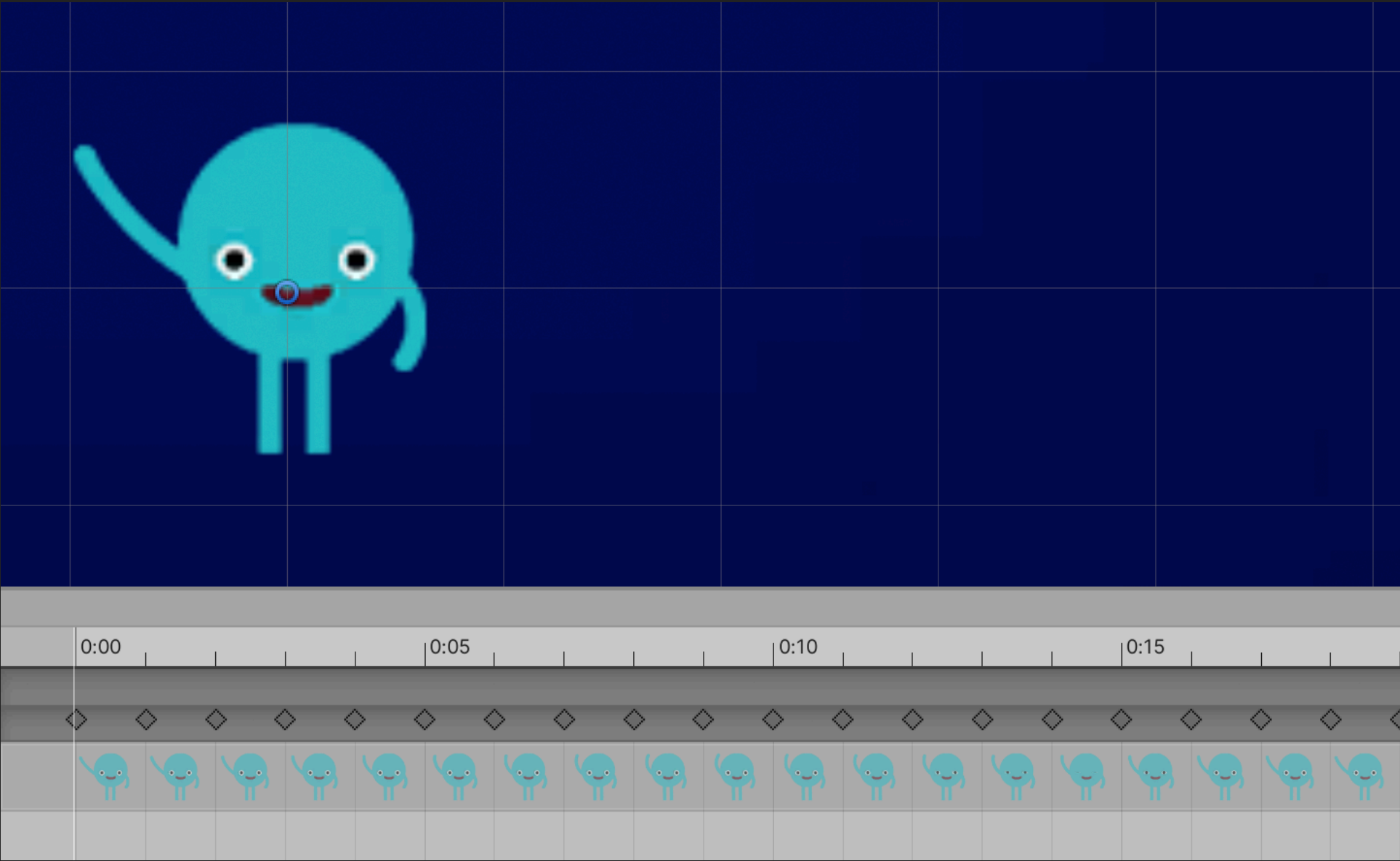
Introduction to Media Authoring

저작기초

Animation

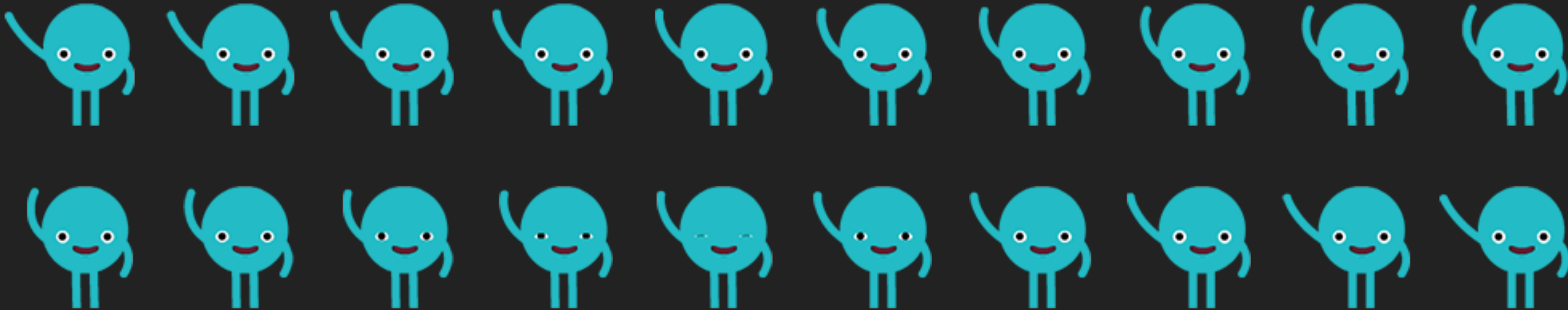
2021. 3. 19.

Animation

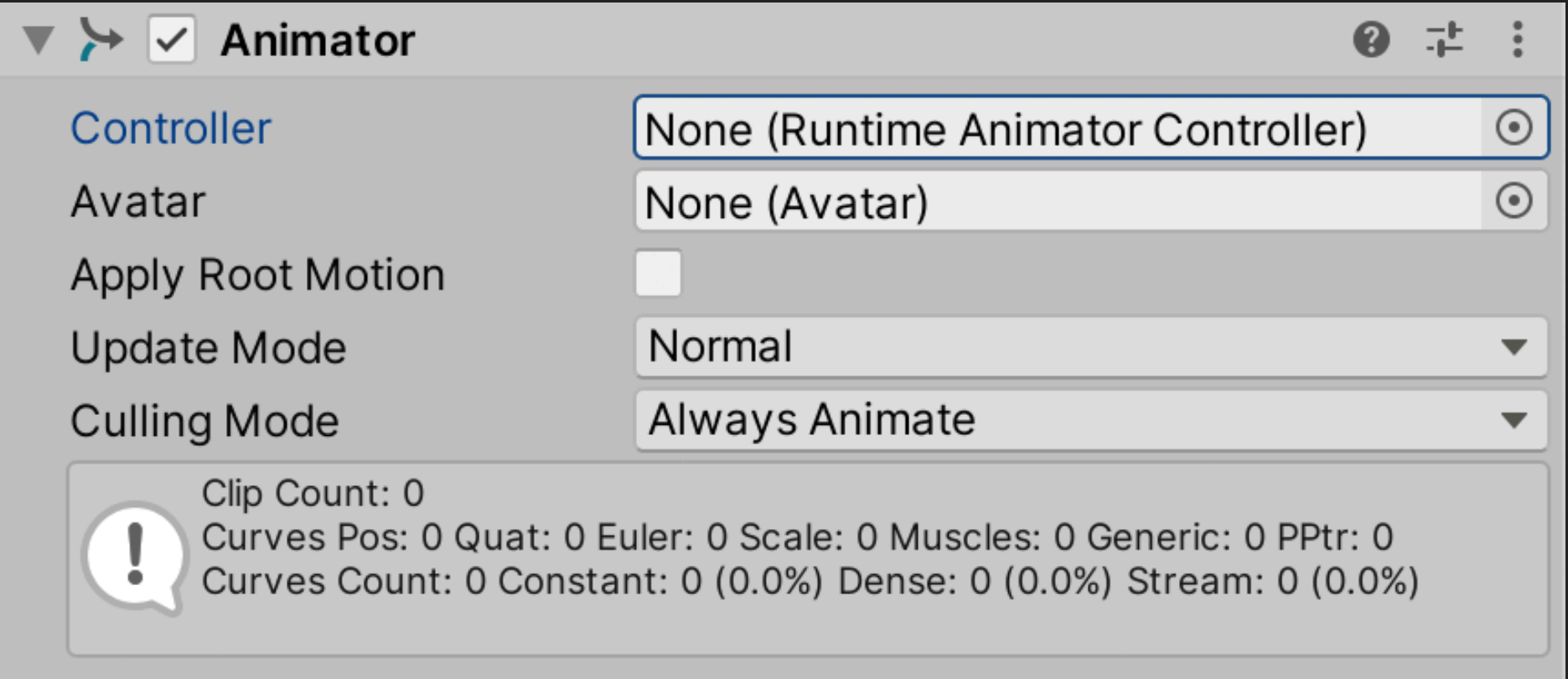
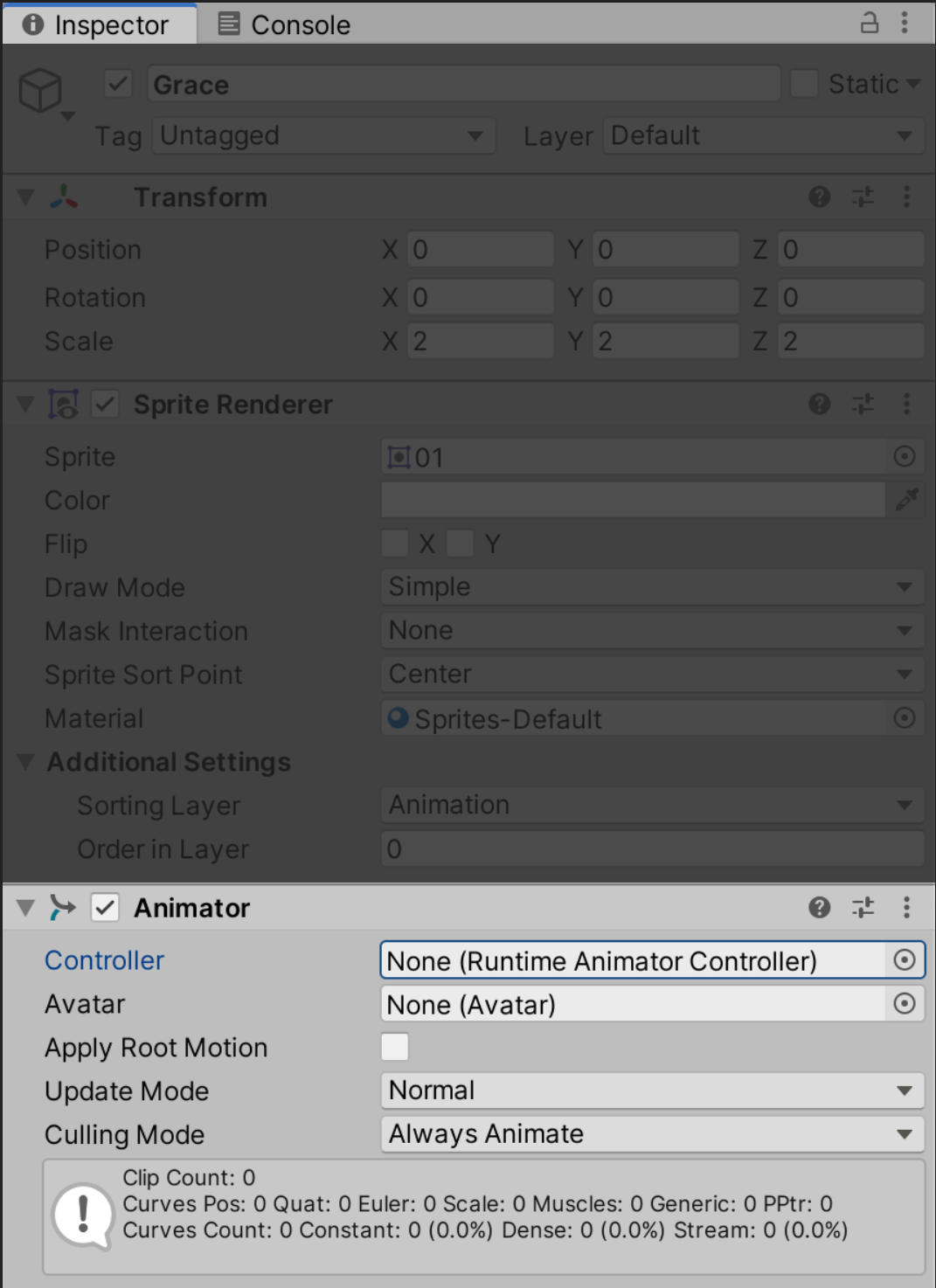


Animation

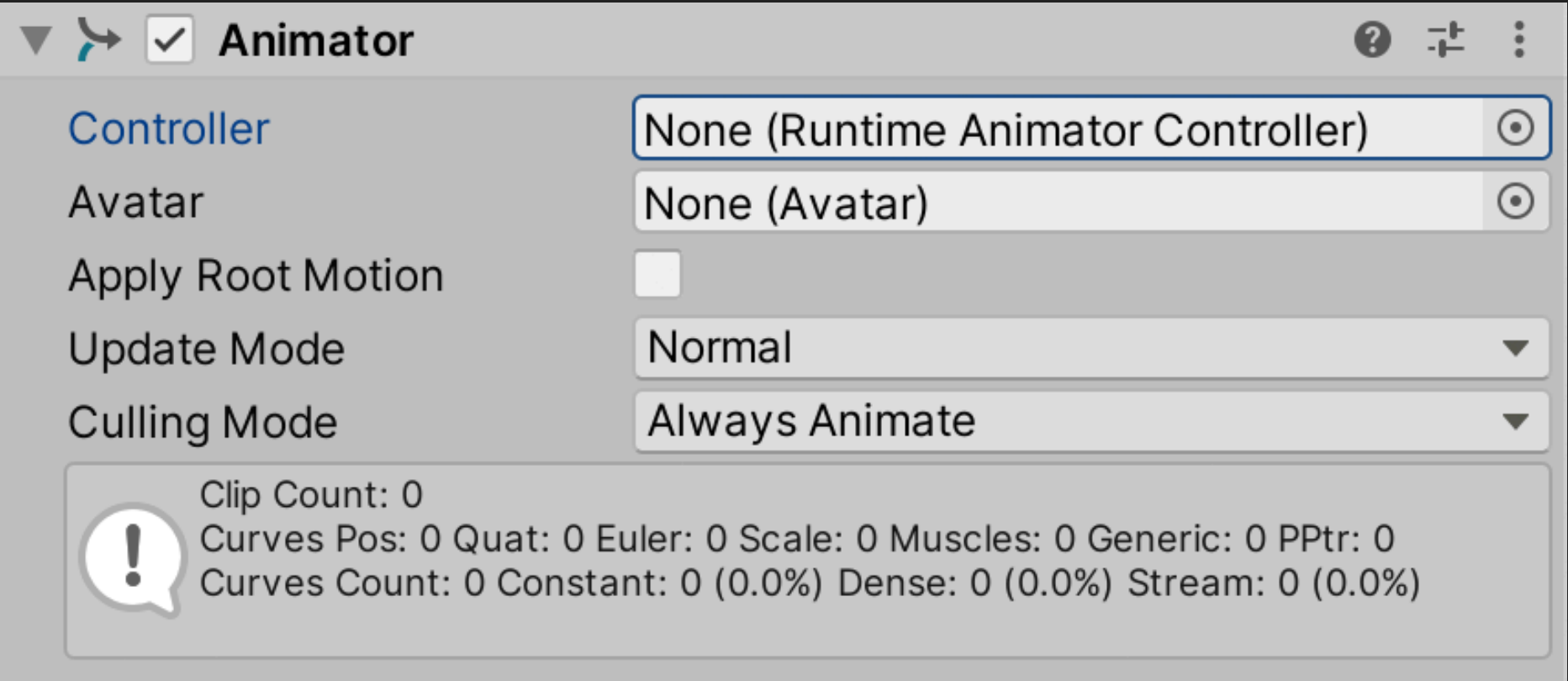
Sprite Animation



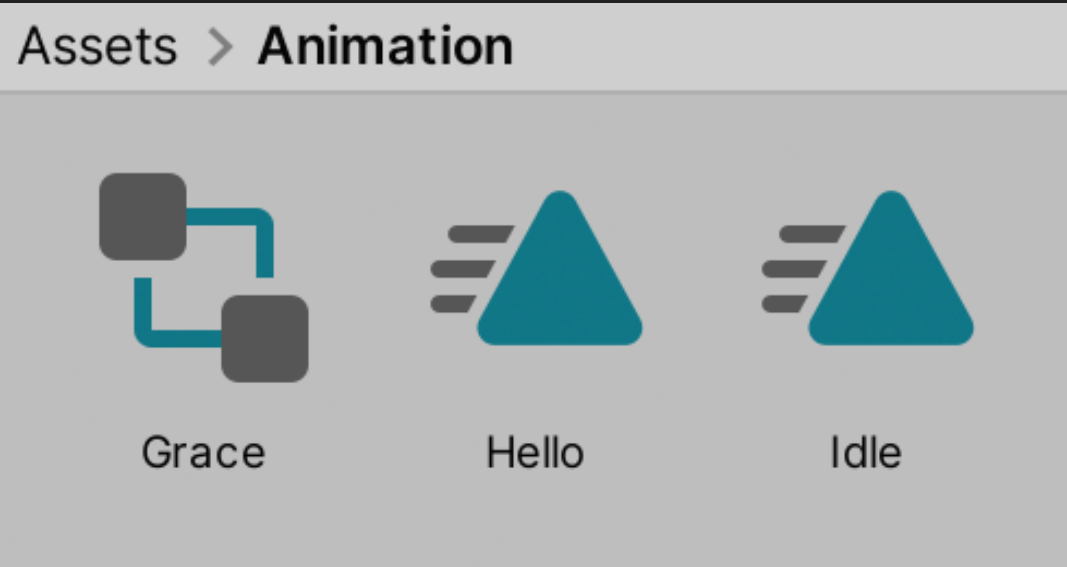
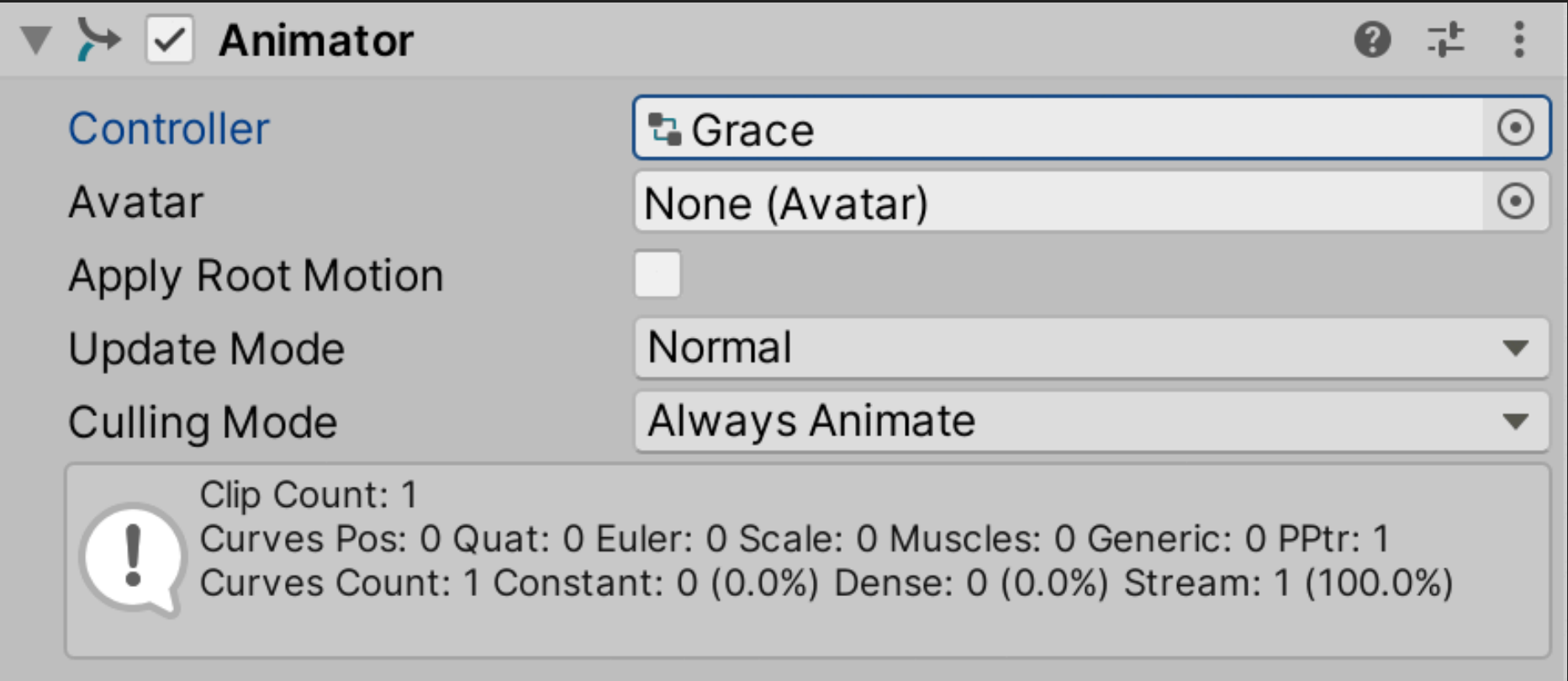
Animator Component - 애니메이션 기능을 관리



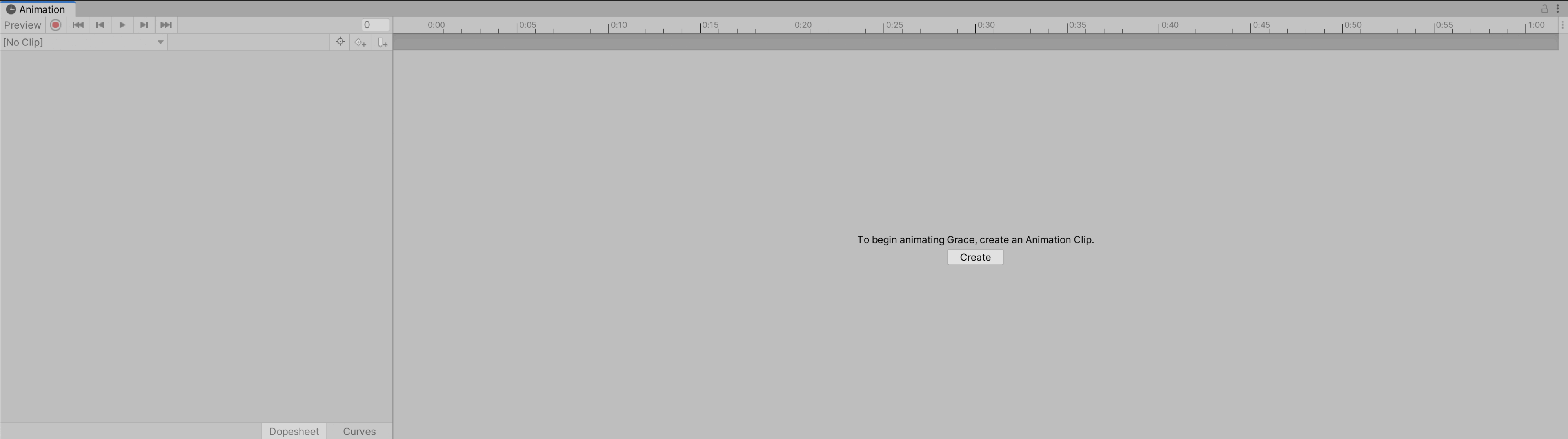
Animator Component - 애니메이션 기능을 관리



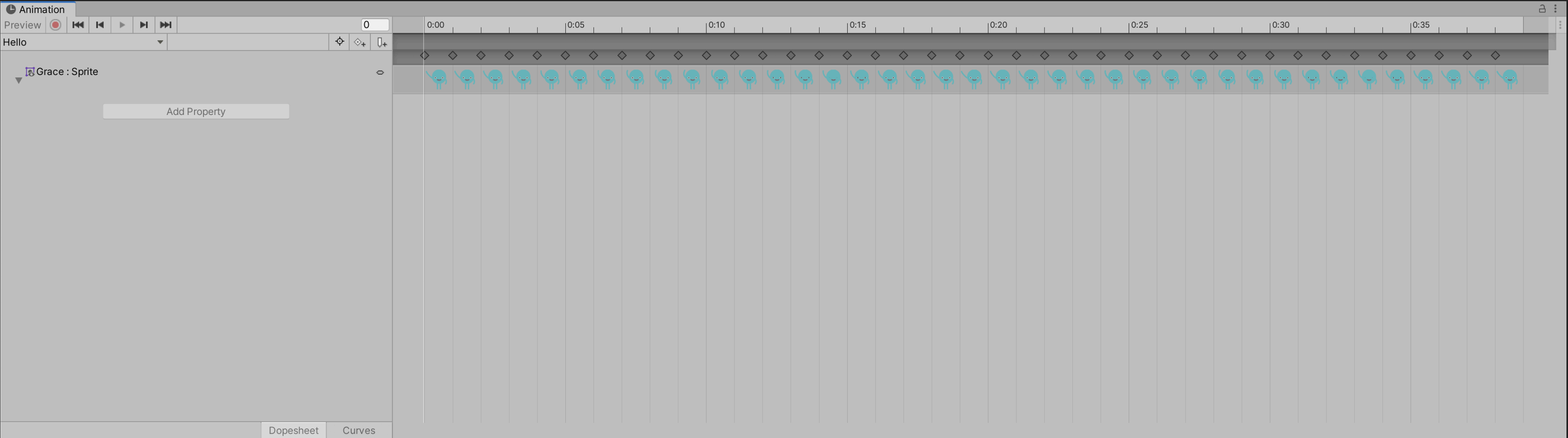
Animator Controller - 애니메이션 상태를 조작하는 Asset



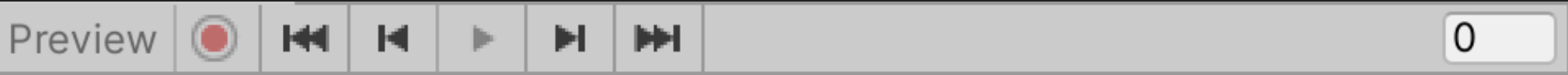
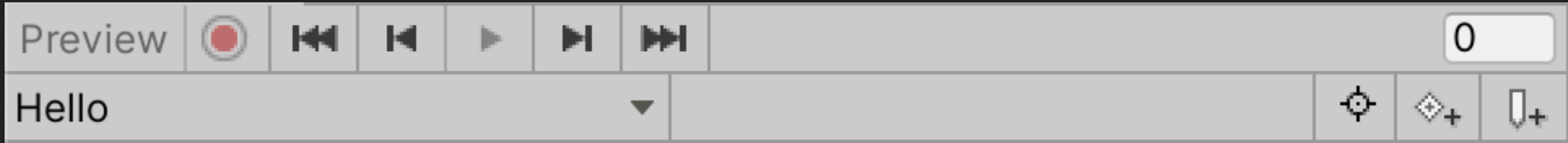
Animation View - 애니메이션을 구성



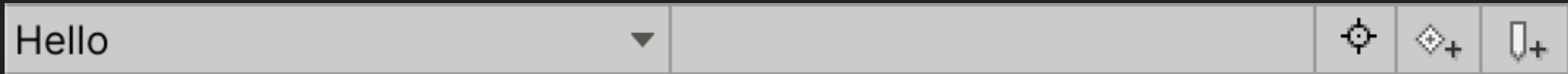
Animation View - 애니메이션을 구성



Animation View - 애니메이션을 구성

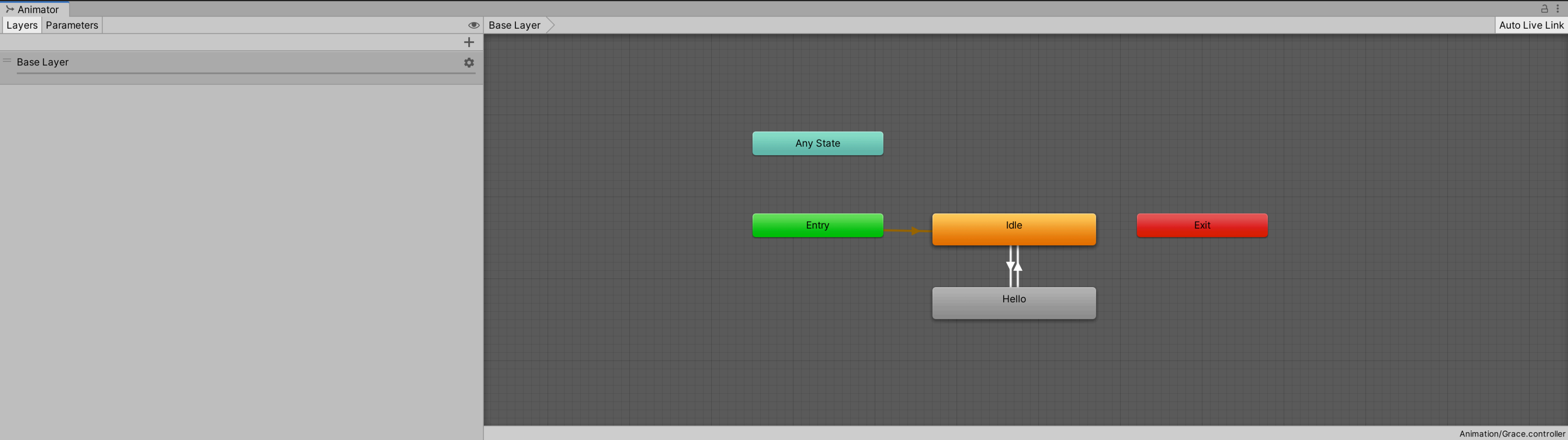


녹화 / 첫 프레임 / 이전 프레임 / 재생 / 다음 프레임 / 마지막 프레임 현재 프레임

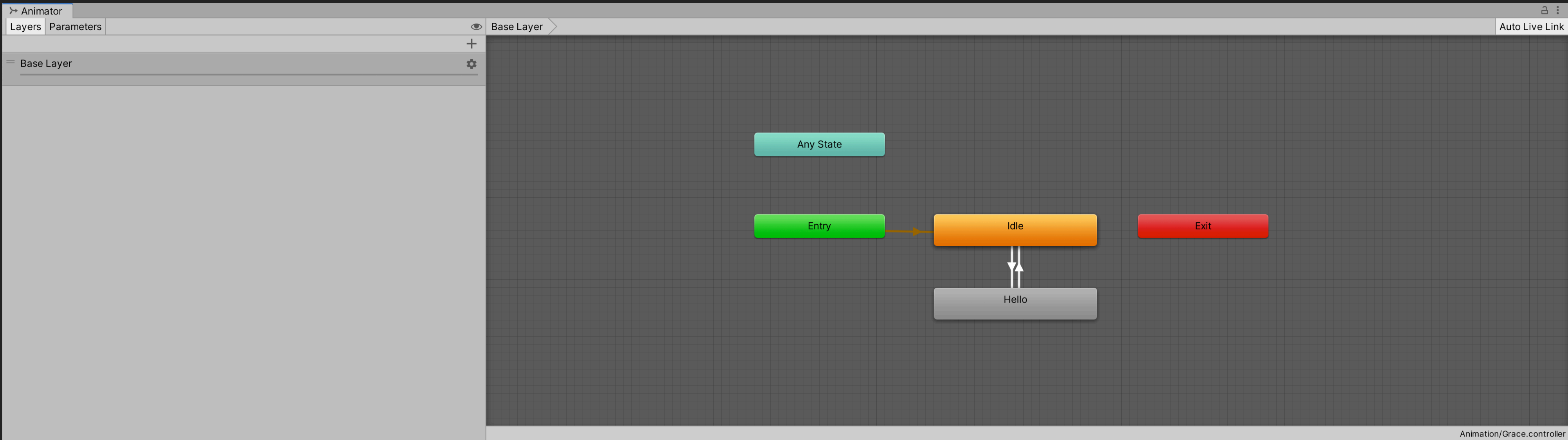


애니메이션 선택

Animator View - 애니메이션 상태를 구성



Layers - 애니메이션 레이어를 구성



Any State - 상태 전환 대기, 모든 상태

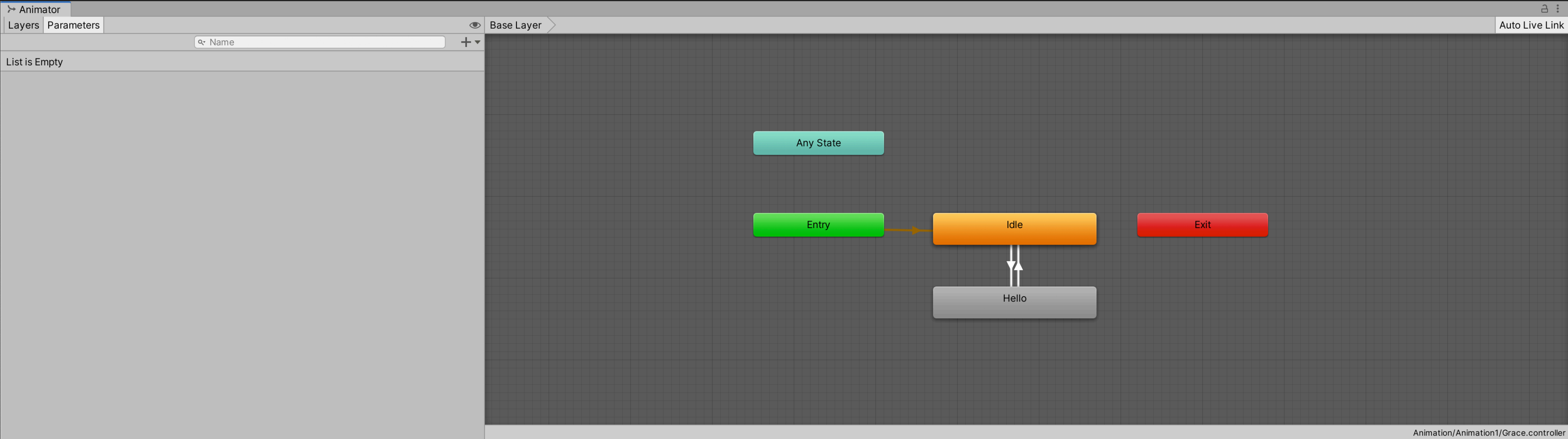
Entry State - 애니메이션 시작 상태를 지정

Exit State - 애니메이션 종료 상태를 지정 - Entry State로 다시 돌아옴

Transition - 애니메이션 상태 전환 및 조건

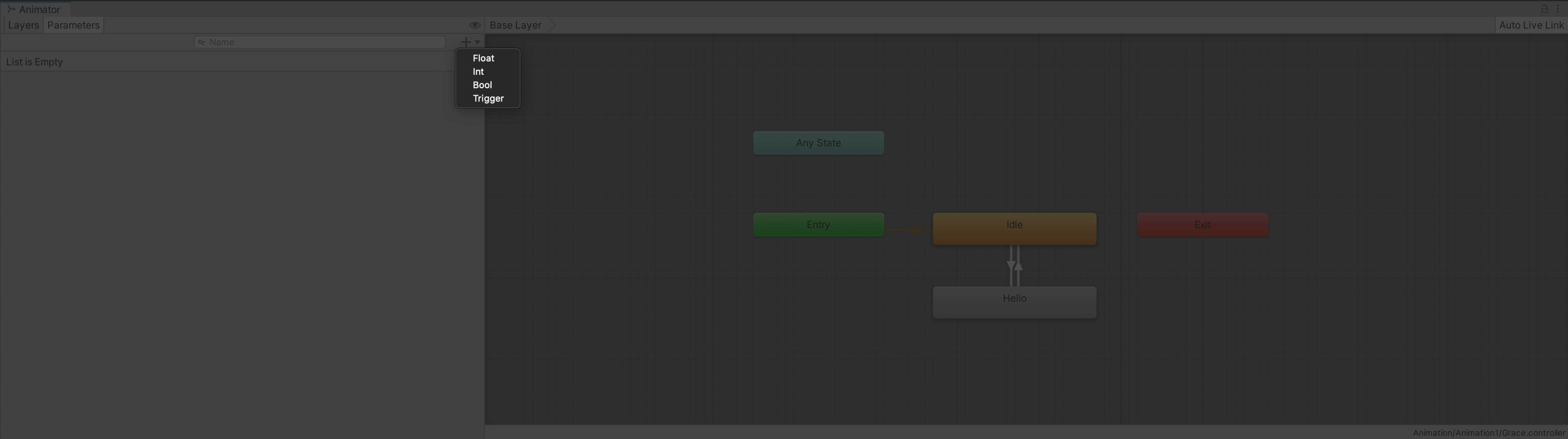
Animation - Animator View : Parameters

Parameters - 애니메이션 전환을 위한 값을 설정

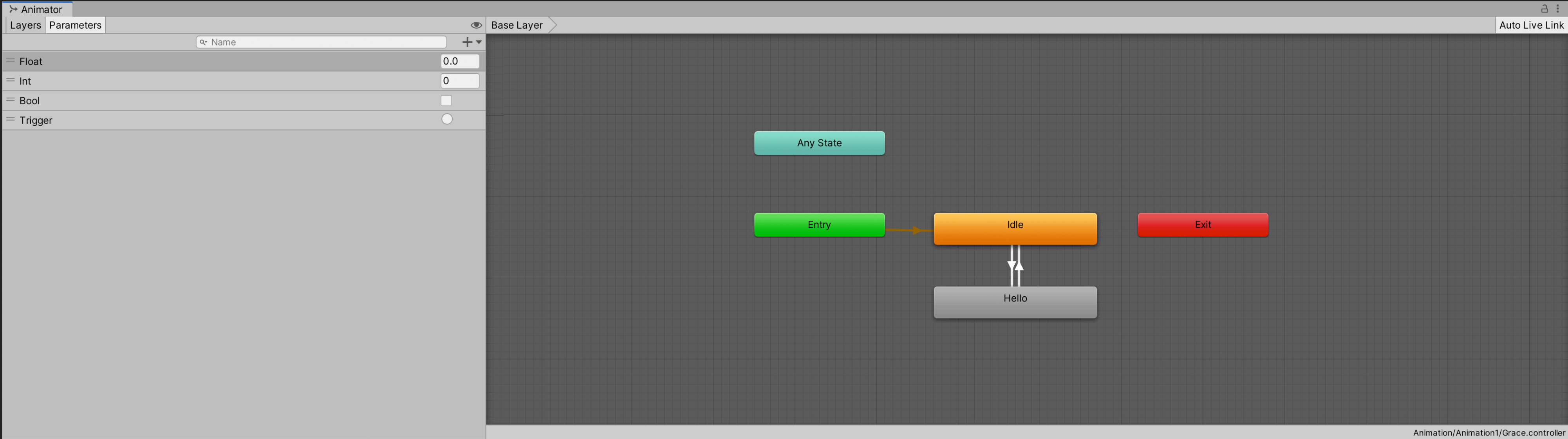


Animation - Animator View : Parameters

Parameters - 애니메이션 전환을 위한 값을 설정



Parameters - 애니메이션 전환을 위한 값을 설정

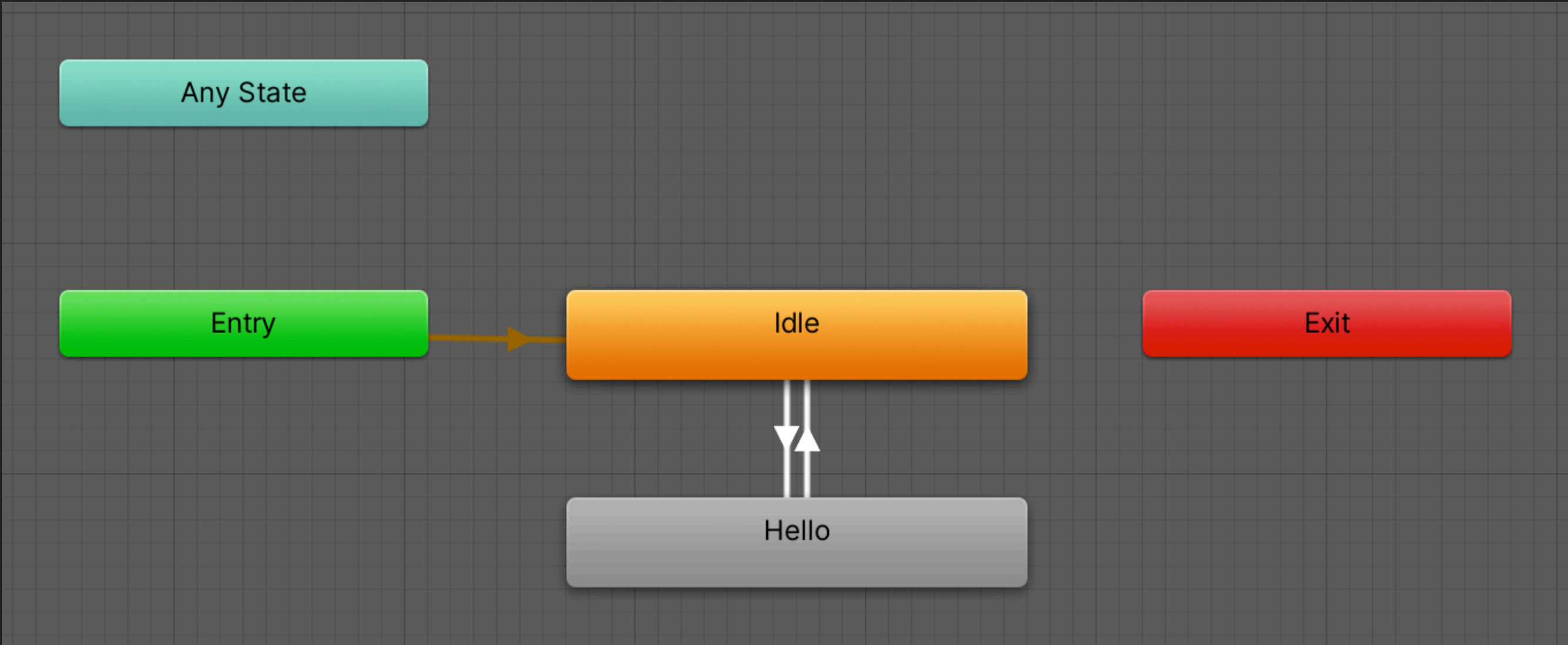
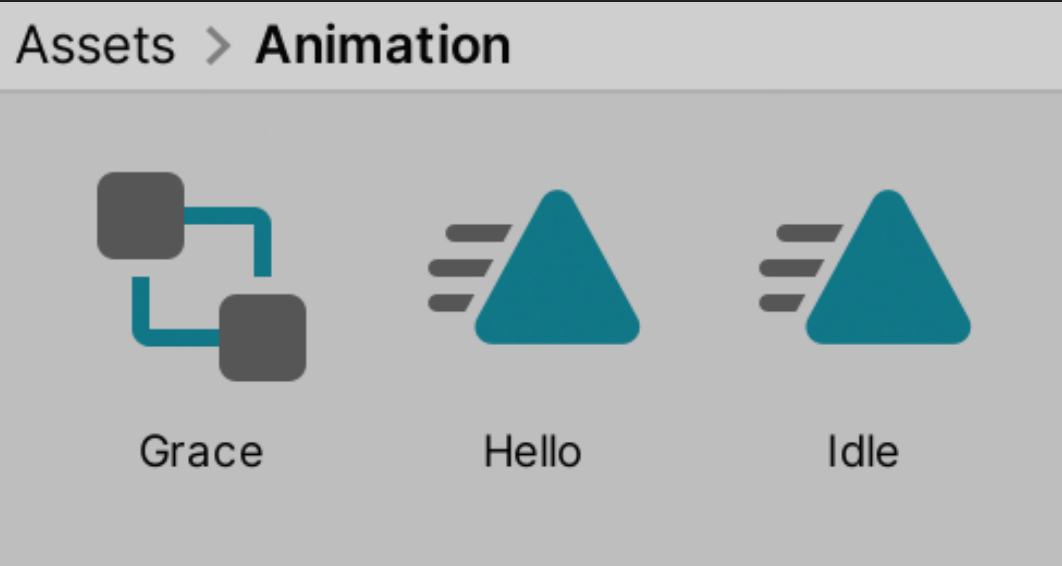


- Float
- Int
- Bool
- Trigger

Float - 실수 0.0f
Int - 정수 0
Bool - 참 / 거짓 true / false
Trigger - 실행

```
GetFloat("Float"); SetFloat("Float", 0.0f);  
GetInteger("Int"); SetInteger("Int", 0);  
GetBool("Bool"); SetBool("Bool", false); SetBool("Bool", true);  
SetTrigger("Trigger");
```

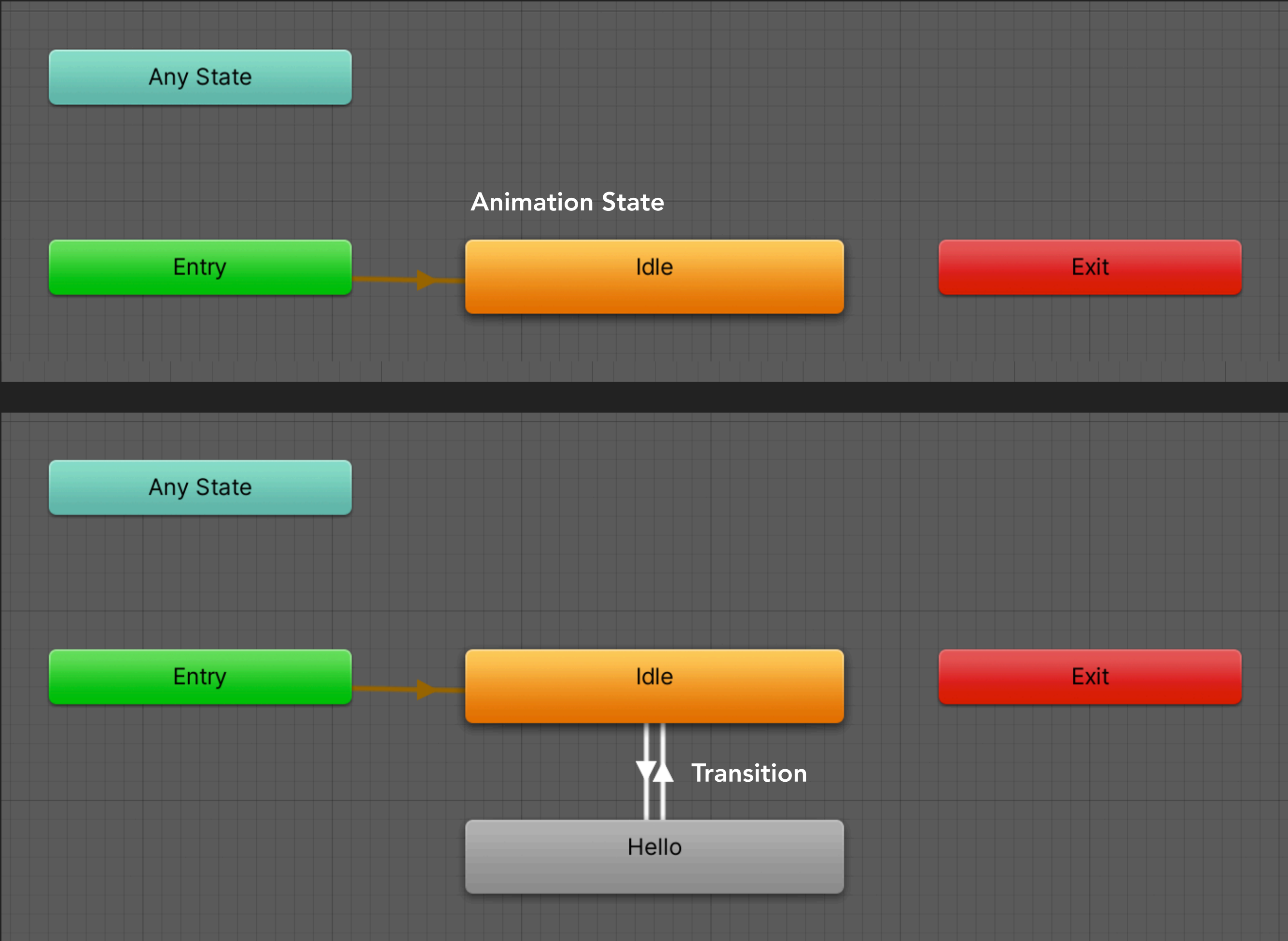

Animator Controller - 애니메이션 상태를 조작하는 Asset



Animator Controller - 애니메이션 상태를 조작하는 Asset

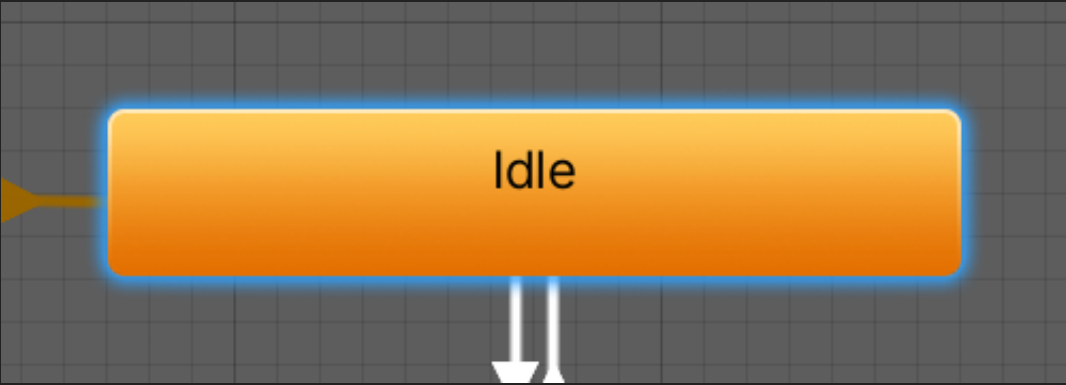
Animation State - 현재 애니메이션의 상태

Transition - 애니메이션 상태 전환 및 조건



Animator Controller - Animation State

Animation State - 현재 애니메이션의 상태 : Idle Animation Clip



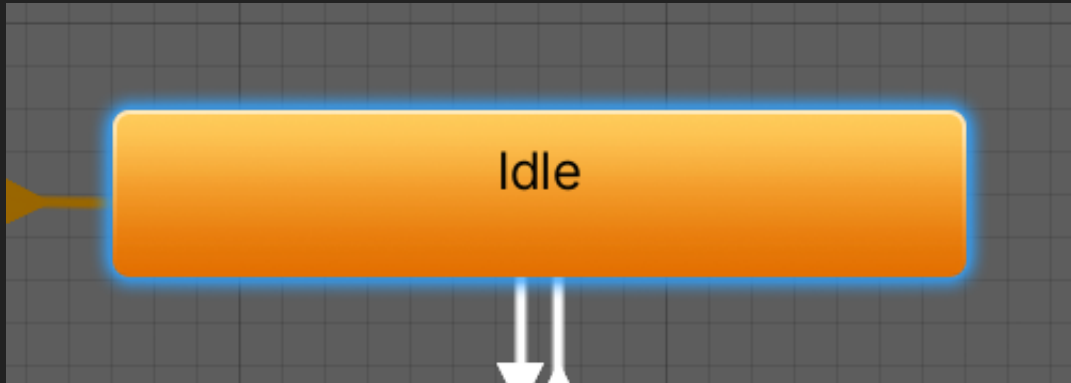
Idle State



Animation



Animation State - 현재 애니메이션의 상태 : Idle Animation Clip



Idle State



Inspector

Console

Idle

Tag

Motion

Speed

1

Multiplier

Parameter

Motion Time

Parameter

Mirror

Parameter

Cycle Offset

0

Parameter

Foot IK

Write Defaults

Transitions

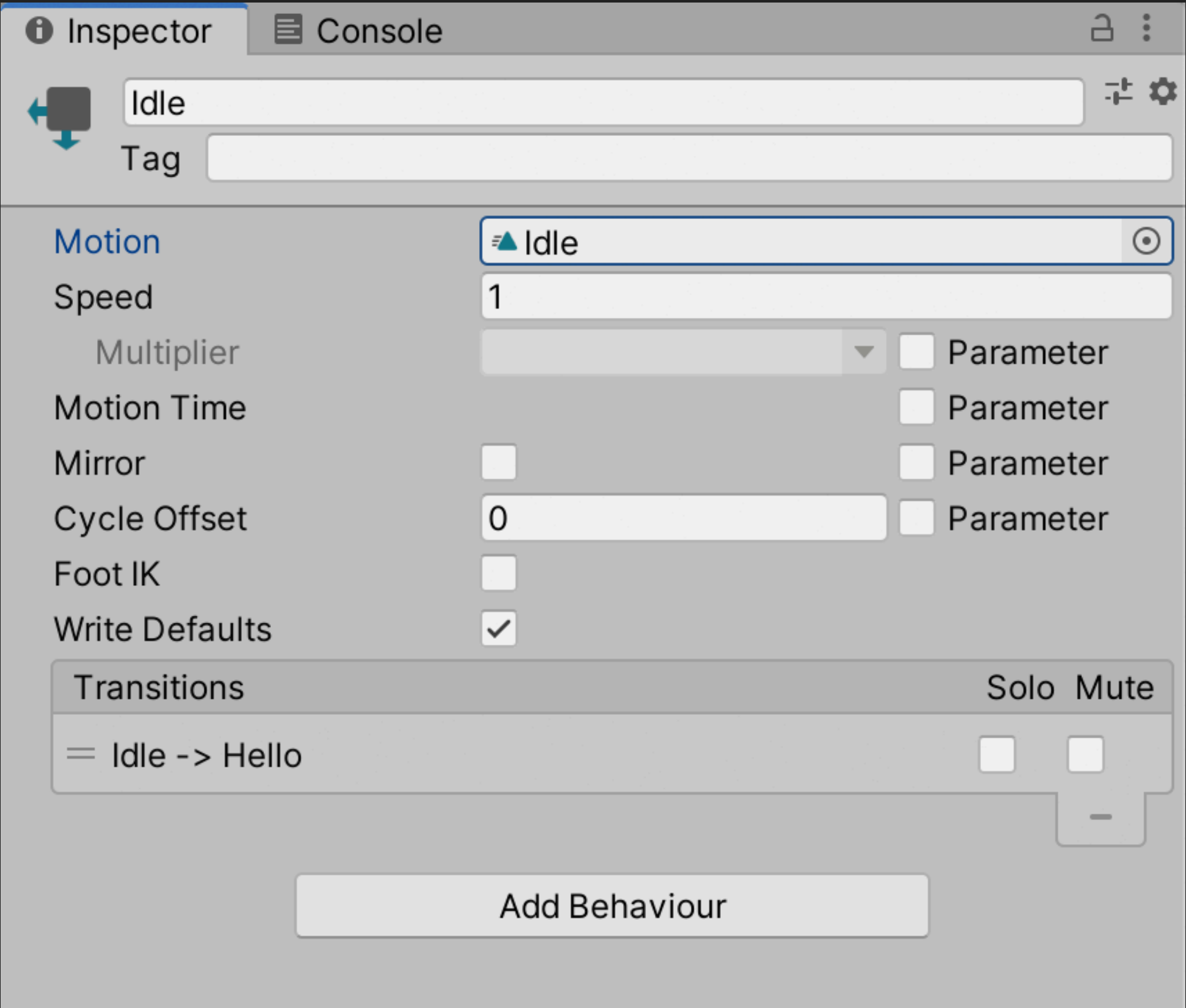
Solo

Mute

= Idle -> Hello

Add Behaviour

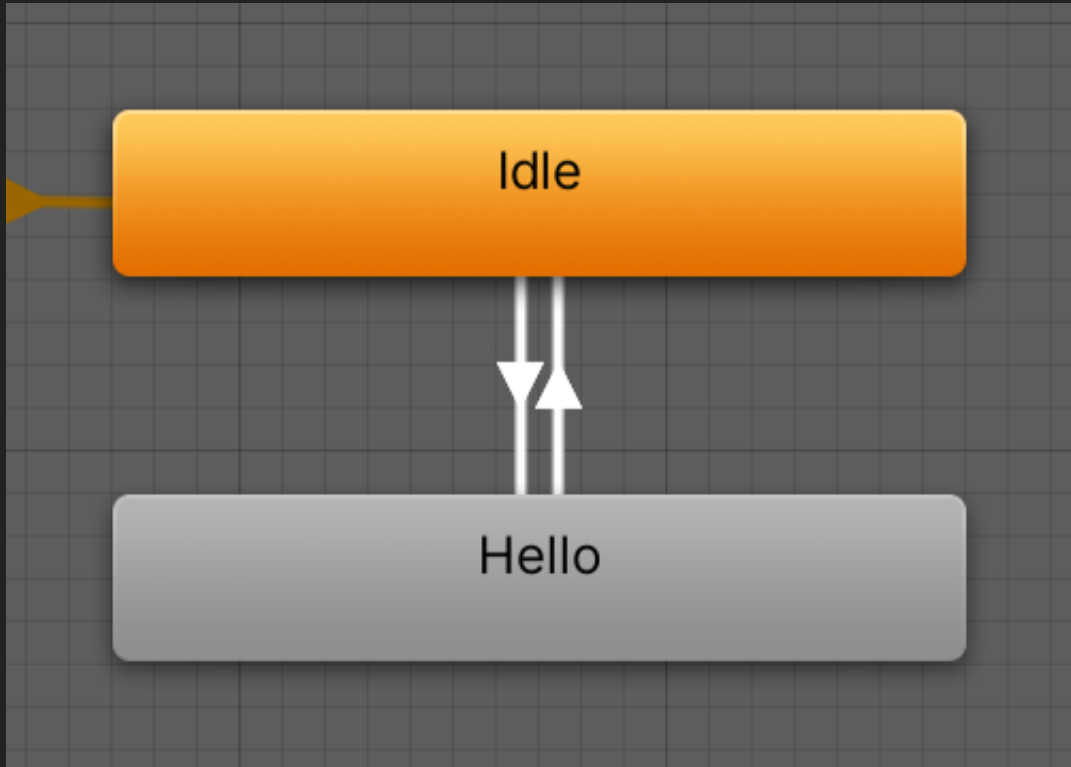
Animation State - 현재 애니메이션의 상태 : Idle Animation Clip



Motion - Animation Clip

Speed - 애니메이션 속도 조절

Transition - 애니메이션 상태 전환 및 조건



Inspector

Console

Idle -> Hello

1 AnimatorTransitionBase

Transitions

Solo

Mute

Idle -> Hello

Has Exit Time

☒

Settings

Exit Time

0.25

Fixed Duration

☒

Transition Duration (s)

0.25

Transition Offset

0

Interruption Source

None

Ordered Interruption

☒

0:00

0:05

0:10

0:15

0:20

0:25

1:00

1:05

1:10

1:15

1:20

Idle

Hello

Conditions

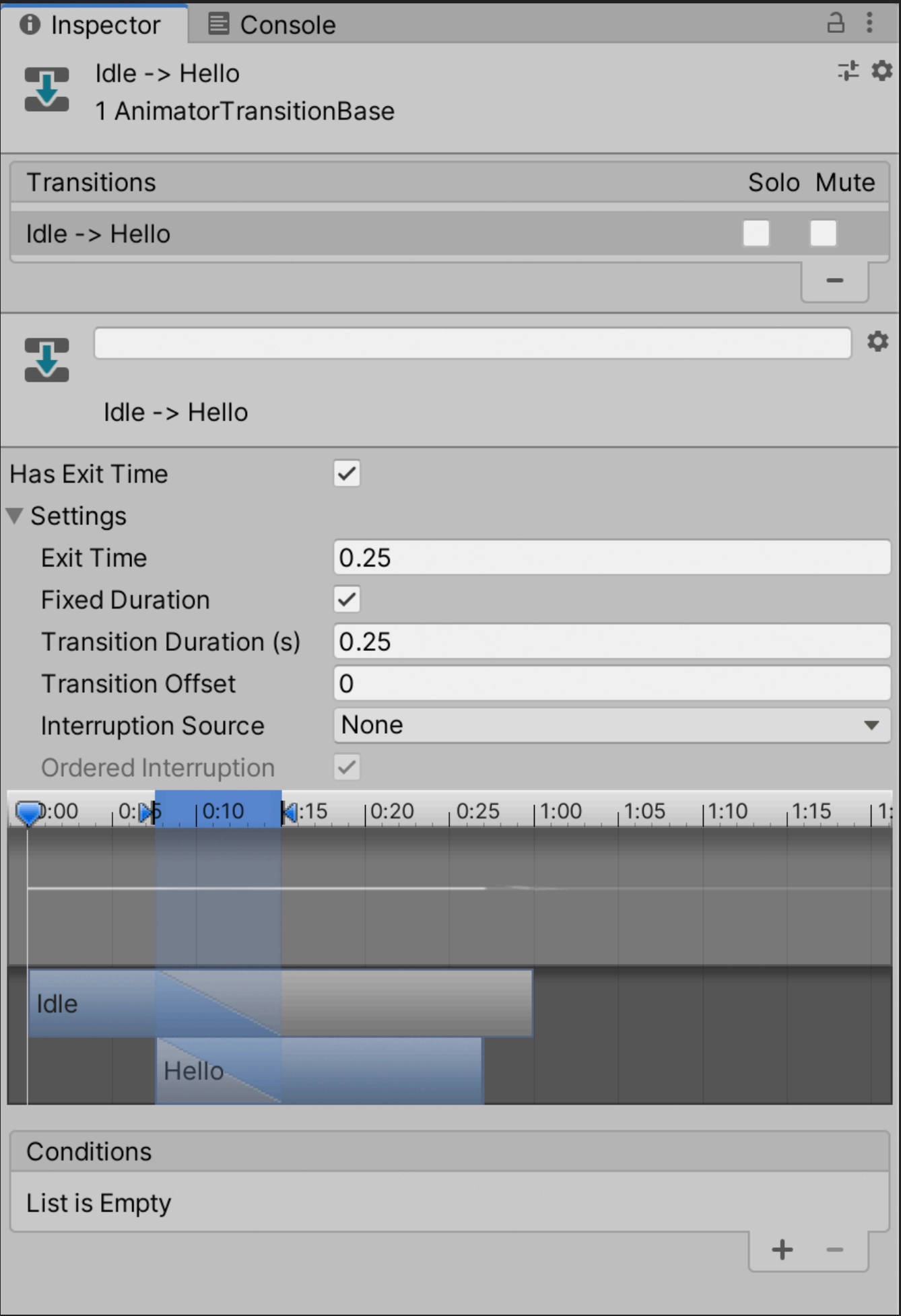
List is Empty

+

-

Animator Controller - Transition

Transition - 애니메이션 상태 전환 및 조건



Has Exit Time - 상태 변경될 때 애니메이션 종료 후 전환 / 바로 전환 여부

Setting - Exit Time - 현 상태의 애니메이션 종료 지점 설정

Setting - Transition Duration - 다음 상태의 애니메이션에 겹치는 전환 구간 설정

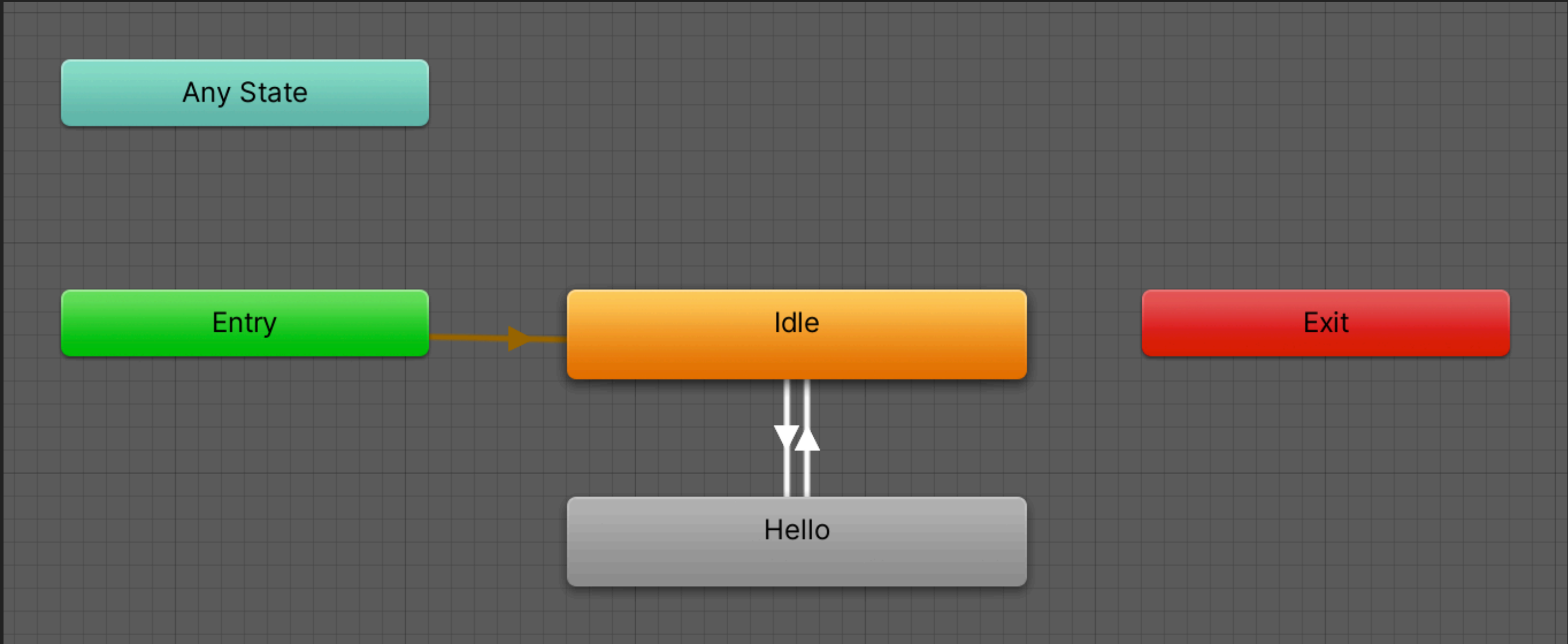
Condition - 애니메이션 상태 전환의 조건

Animation 실습

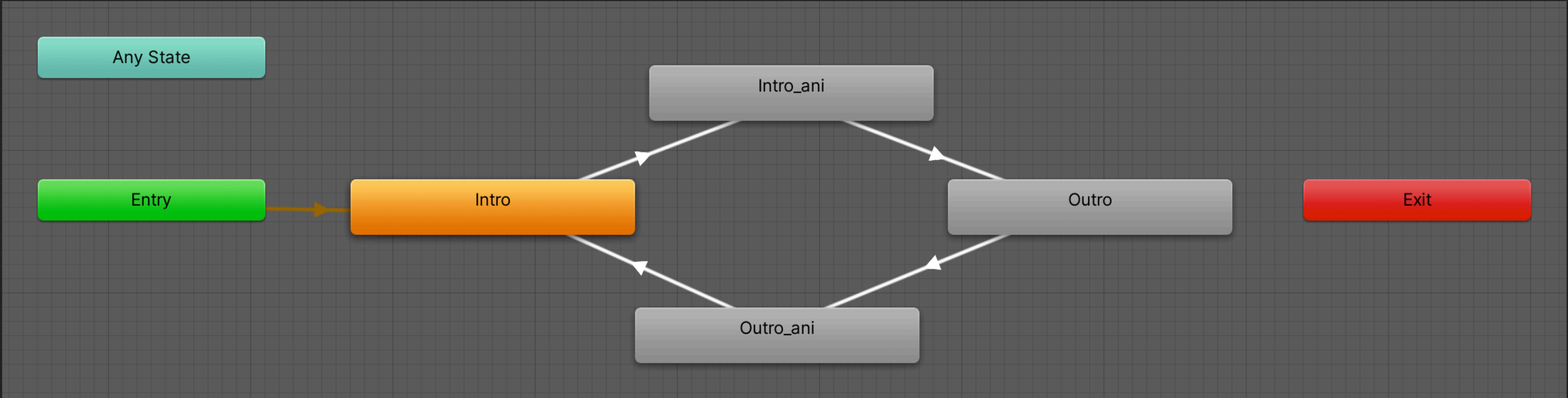
Animation 만들기

1. Window > Animation 선택하여 창 열기(Animation)
2. Project 창에서 애니메이션을 만들 게임 오브젝트(Game Object)를 선택
3. Animation 창에서 Create 버튼을 클릭하여 새로운 애니메이션을 생성 저장 후 제작
4. Project 창에서 오른쪽 마우스 클릭 Create > Animator Controller 선택, 새로운 Animator를 제작
5. Window > Animation > Animator 를 선택하여 창 열기(Animator)
6. Animator 창에 제작한 Animation Clip 을 드래그 앤 드랍으로 붙여넣기
7. 기본으로 지정할 Animation Clip 에 오른쪽 마우스 클릭하여 Set as Layer Default State 선택
8. 각 Animation Clip 에 오른쪽 마우스 클릭하여 Make Transition 선택 후 다른 Animation Clip 으로 화살표 연결(Transition)
9. Animation Clip 연결 완료 후 Hierarchy 창에서 게임 오브젝트를 선택하고 제작된 Animator 를 Project 창에서 찾아서 드래그 앤 드랍으로 연결

실습 1 구조



실습 2 구조



Button 만들기

1. Sprite 이미지를 이용하여 버튼으로 사용할 게임 오브젝트(Game Object) 제작
2. 게임 오브젝트 선택 후 Inspector 창에서 Add Component 버튼 클릭
3. Add Component 클릭 : Physics 2D > Box Collider 선택하여 Component 추가
4. Add Component 클릭 : New script 선택 후 'Button' 으로 기입하여 추가
5. Button.cs 파일 생성됨.(C# 파일)

```
public class Button : MonoBehaviour
{
    void OnMouseDown( )
    {
        Debug.Log( "Mouse Down" );
    }
    void OnMouseUp( )
    {
        Debug.Log( "Mouse up" );
    }
    void OnMouseEnter( )
    {
        Debug.Log( "Mouse enter" );
    }
    void OnMouseExit( )
    {
        Debug.Log( "Mouse exit" );
    }
}
```

