

Scripting Jump

Now, we will script Jumping action

① Let's modify "Player Movement"

```
public class PlayerMovement : MonoBehaviour
```

```
{  
    // (same as previous code)
```

```
    // Jump
```

```
    public float JumpHeight = 3f;
```

```
    // Jump method
```

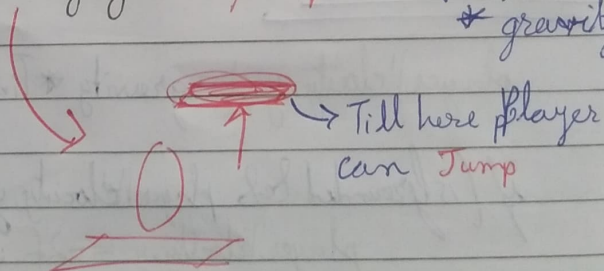
```
    public void Jump()
```

```
{  
    // Check if Player is grounded
```

```
    if (isGrounded) // Jump only if Player is on ground
```

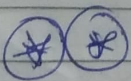
```
{  
        playerVelocity.y = Mathf.Sqrt(JumpHeight * 0.3f  
                                         * gravity);
```

```
    }  
}
```



We will use `Jump()` to ~~set~~ set player velocity's y position

→ [Only when Jump Action is performed]



Reduce JumpHeight if Player jumps too high

② Let's modify "Input Manager"

```
public class InputManager : MonoBehaviour
```

```
{
    // (same as previous)
```

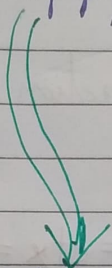
```
    void Awake()
    {
```

```
        movement = GetComponent<PlayerMovement>();
```

```
        onFoot.Jump.performed += ctx => movement.Jump();
```

```
    }
```

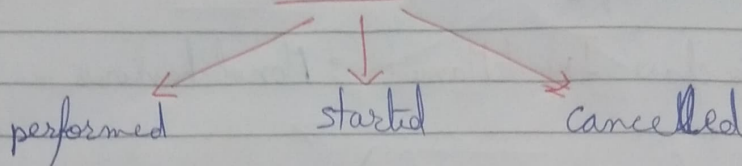
```
}
```



Let's break down this statement

- 1) onFoot.Jump → represents action in Action Map
- 2) performed → this event is triggered when input associated with Jump has been pressed (in this case, its SPACE)
- 3) += → it is not arithmetic addition here.
• performed is an event, hence += acts as event subscription operator
- 4) ctx => movement.Jump() → it is a lambda expression
It defines behaviour that should happen when Jump action is performed
ctx → stands for context (acts as a pointer → Jump())
movement.Jump() → handles logic for player jumping logic.

All actions have 3 states



① started:-

This phase triggers when input action begins

Eg:-

If you want to start a jump action animation the moment the jump button is pressed

onFoot.Jump.started += ctx \Rightarrow movement.JumpAnim();

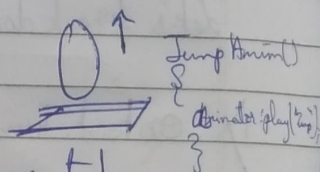
② performed:-

This phase occurs when input action is fully executed

Eg:-

If you want to change state of player velocity / make player jump

onFoot.Jump.performed += ctx \Rightarrow movement.Jump();



③ cancelled:-

This phase ~~to~~ occurs when input action is cancelled

Eg:-

If you want to switch to power up animation [like in Legend of Zelda]

onFoot.Jump.cancelled += ctx \Rightarrow movement.CancelJump();