

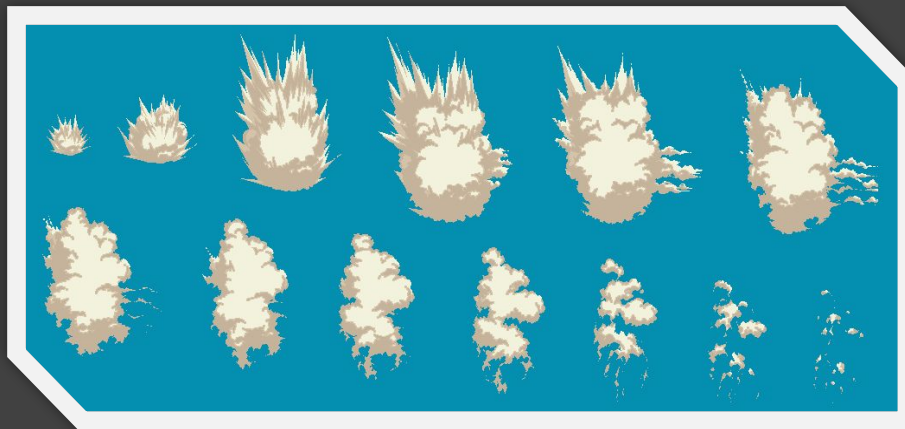
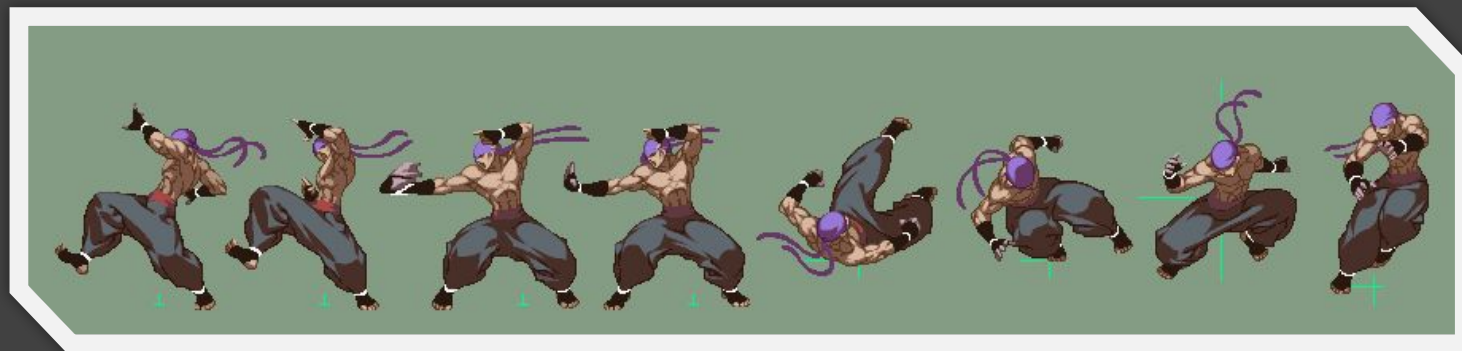
The background features a complex network of thin, light-colored lines connecting various blue and black 3D cubes of different sizes. The cubes are scattered across the frame, creating a sense of depth and connectivity. The overall color palette is muted, with shades of blue, black, and grey against a light, hazy background.

# Анимация игрового персонажа

# Основные способы анимирования

- Покадровая 2D анимация
- Скелетная 2D и 3D анимация
- Захват движений

# Покадровая 2D анимация





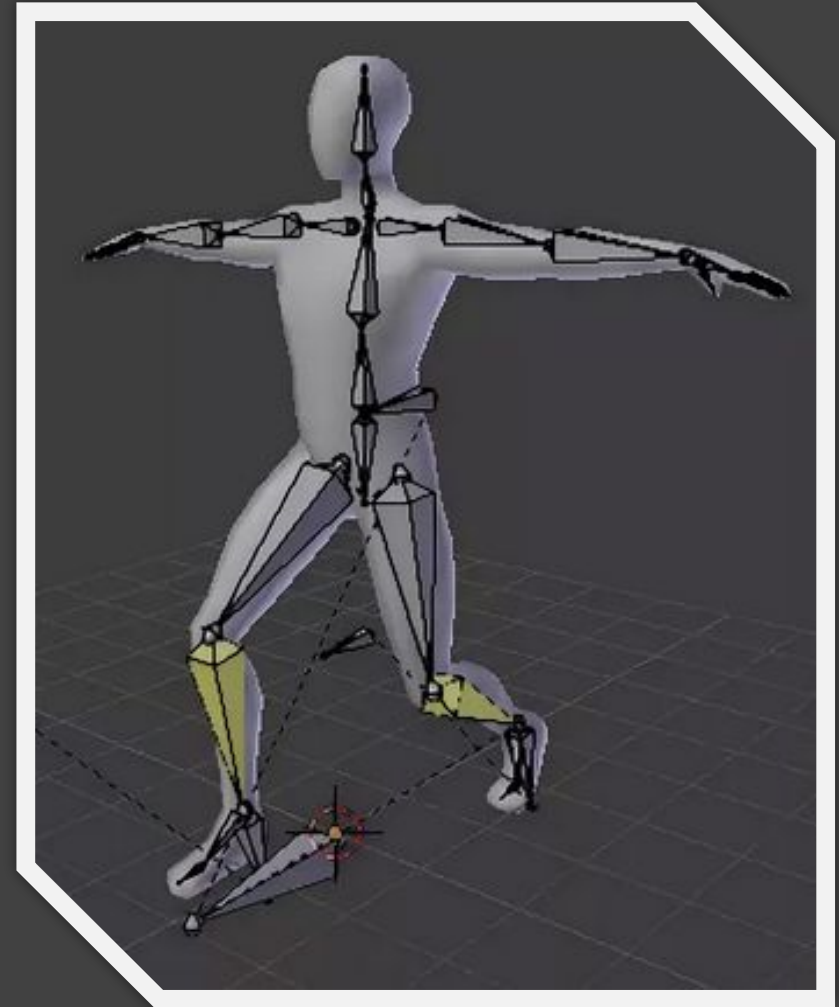
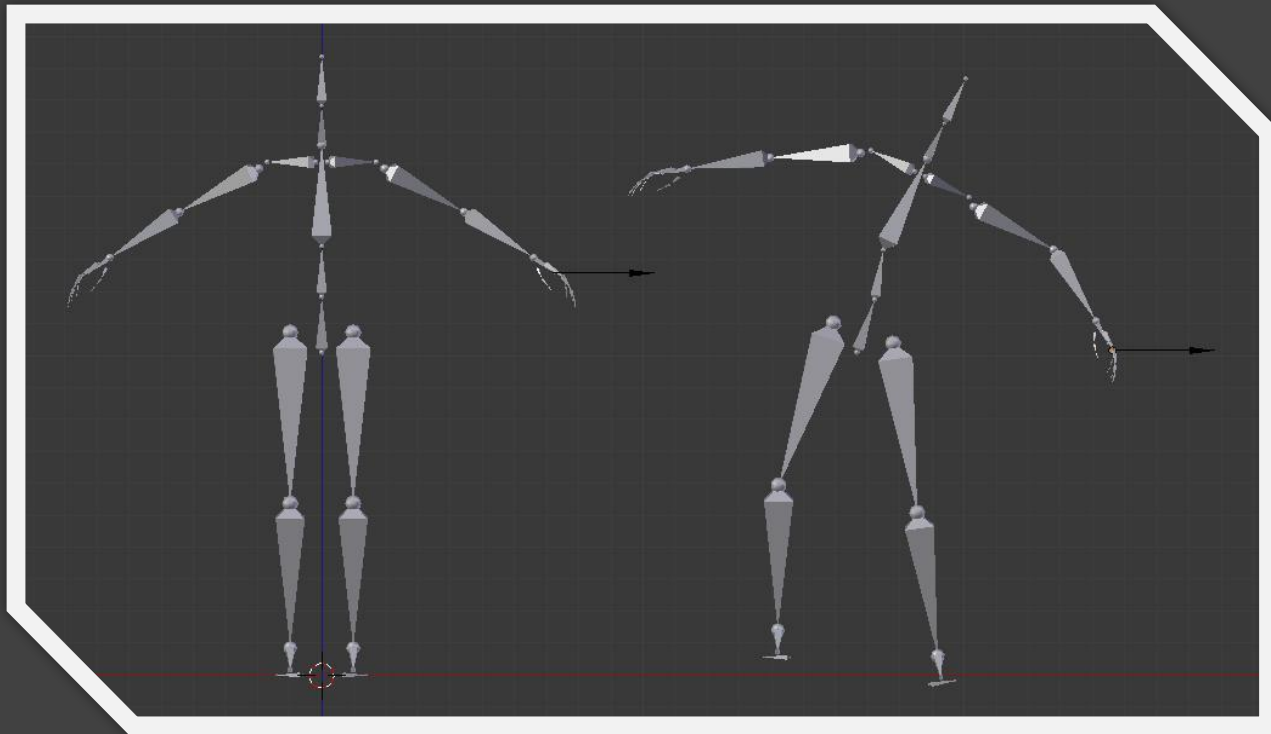
# **Работа покадровой 2D анимации**



# Скелетная 2D анимация



# Скелетная 3D анимация



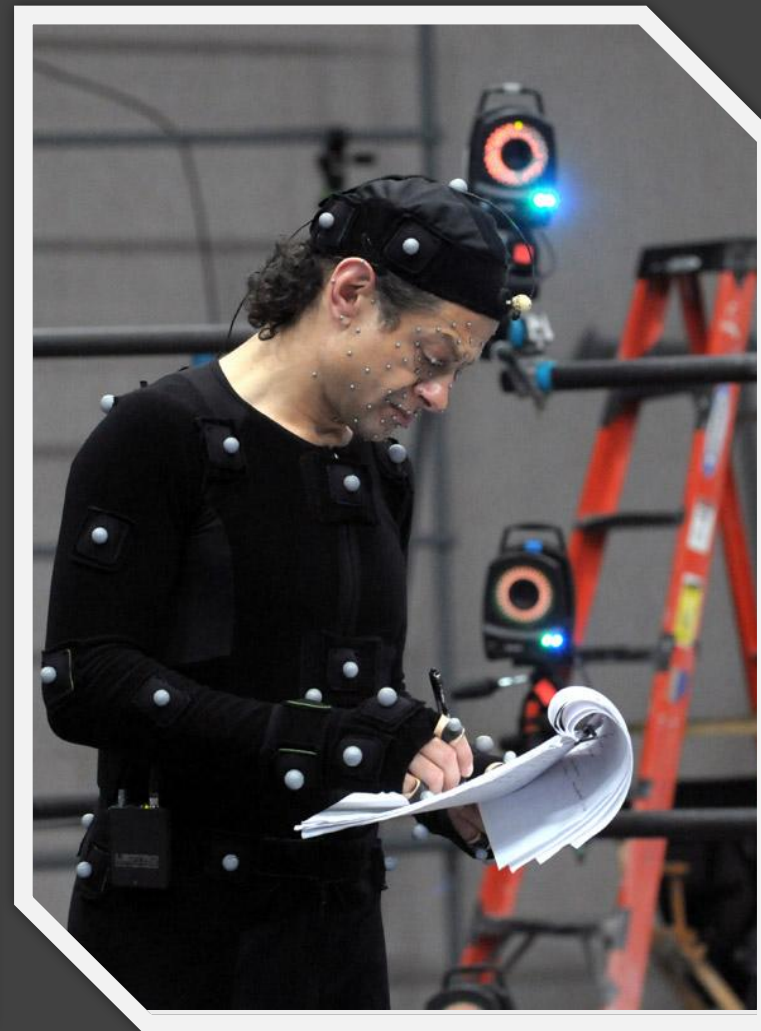


# Работа скелетной анимации





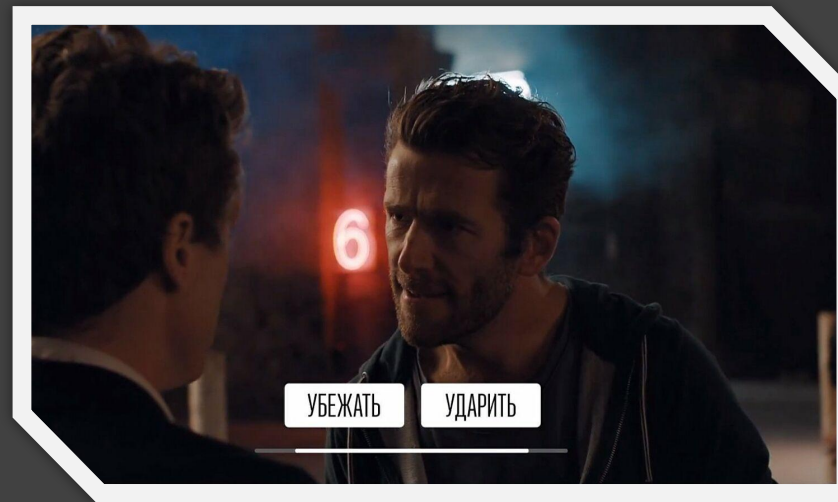
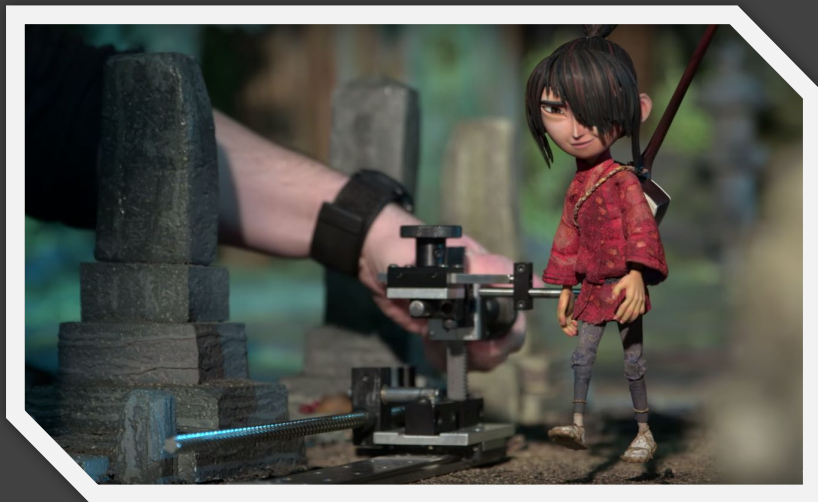
# Захват движений



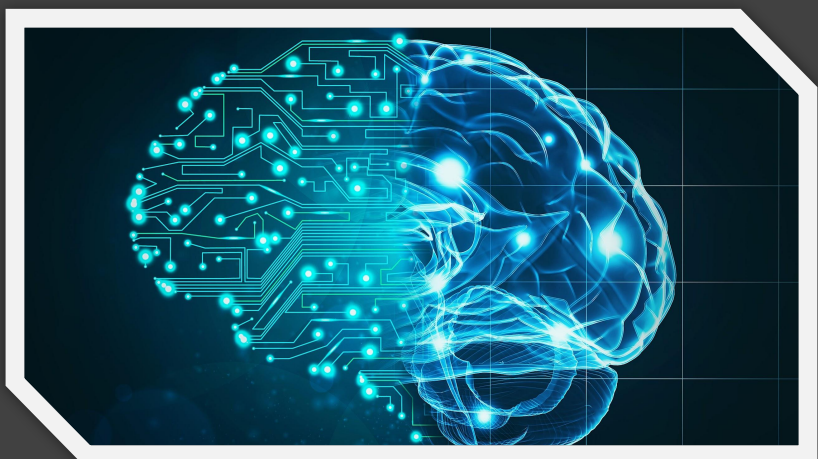


The image shows two individuals in a dark studio environment, wearing black motion capture suits with numerous white reflective markers. They are captured in dynamic poses. The person on the left is in a low, lunging position, while the person on the right is standing upright with one leg raised. The background is dark, with several red and blue circular lights visible, likely used for motion capture tracking. The floor is a light-colored, possibly green or blue, mat.

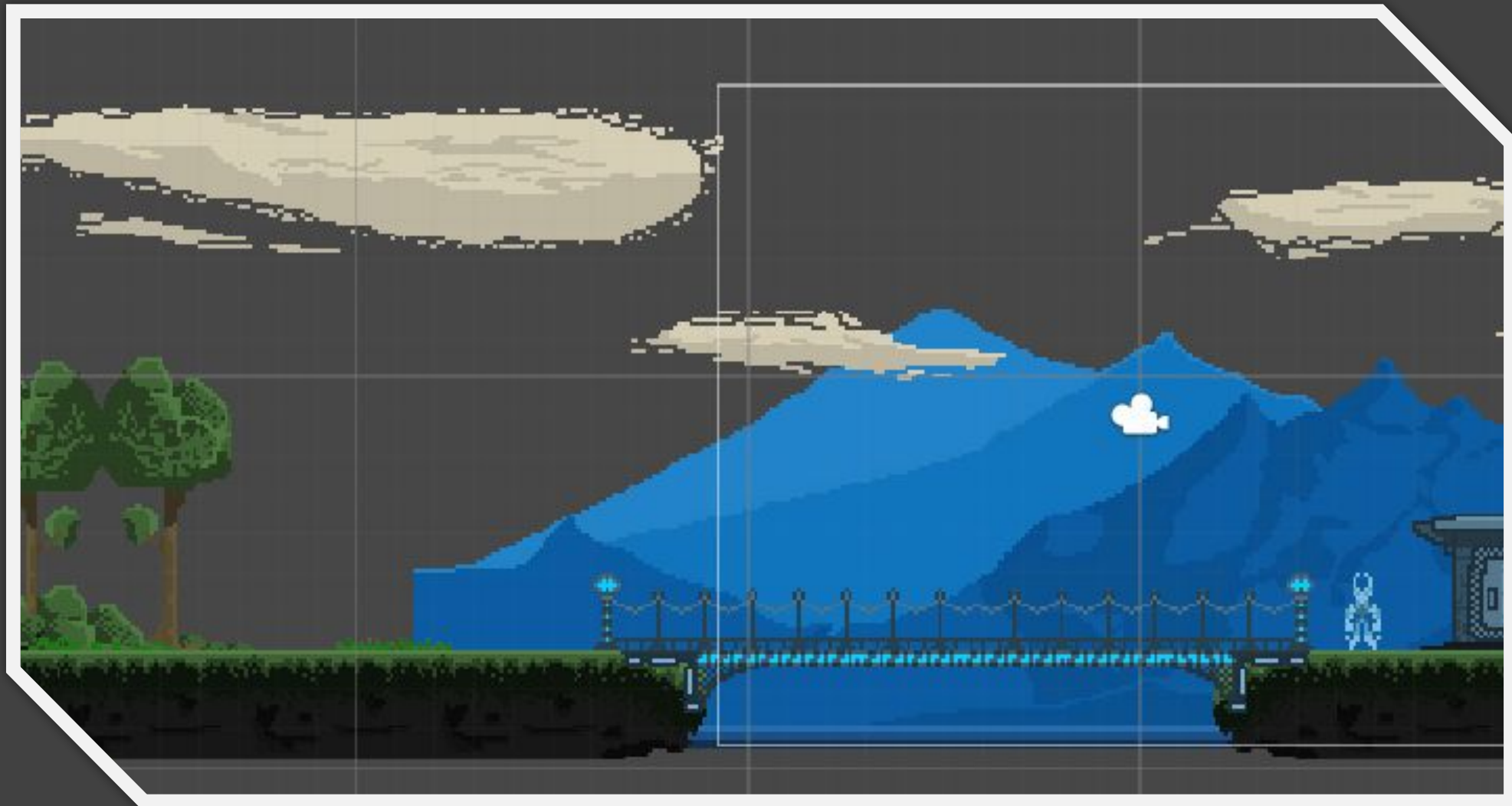
# **Работа захвата движений**



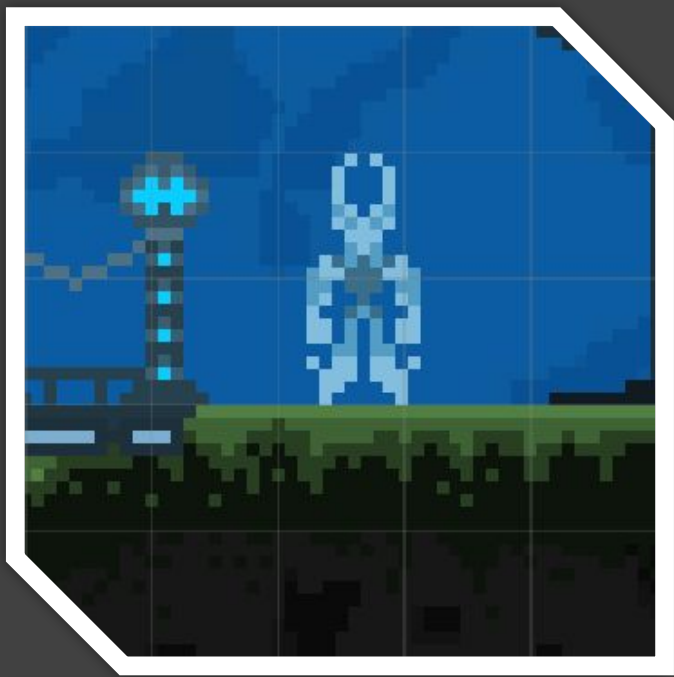
## Другие способы анимирования персонажа



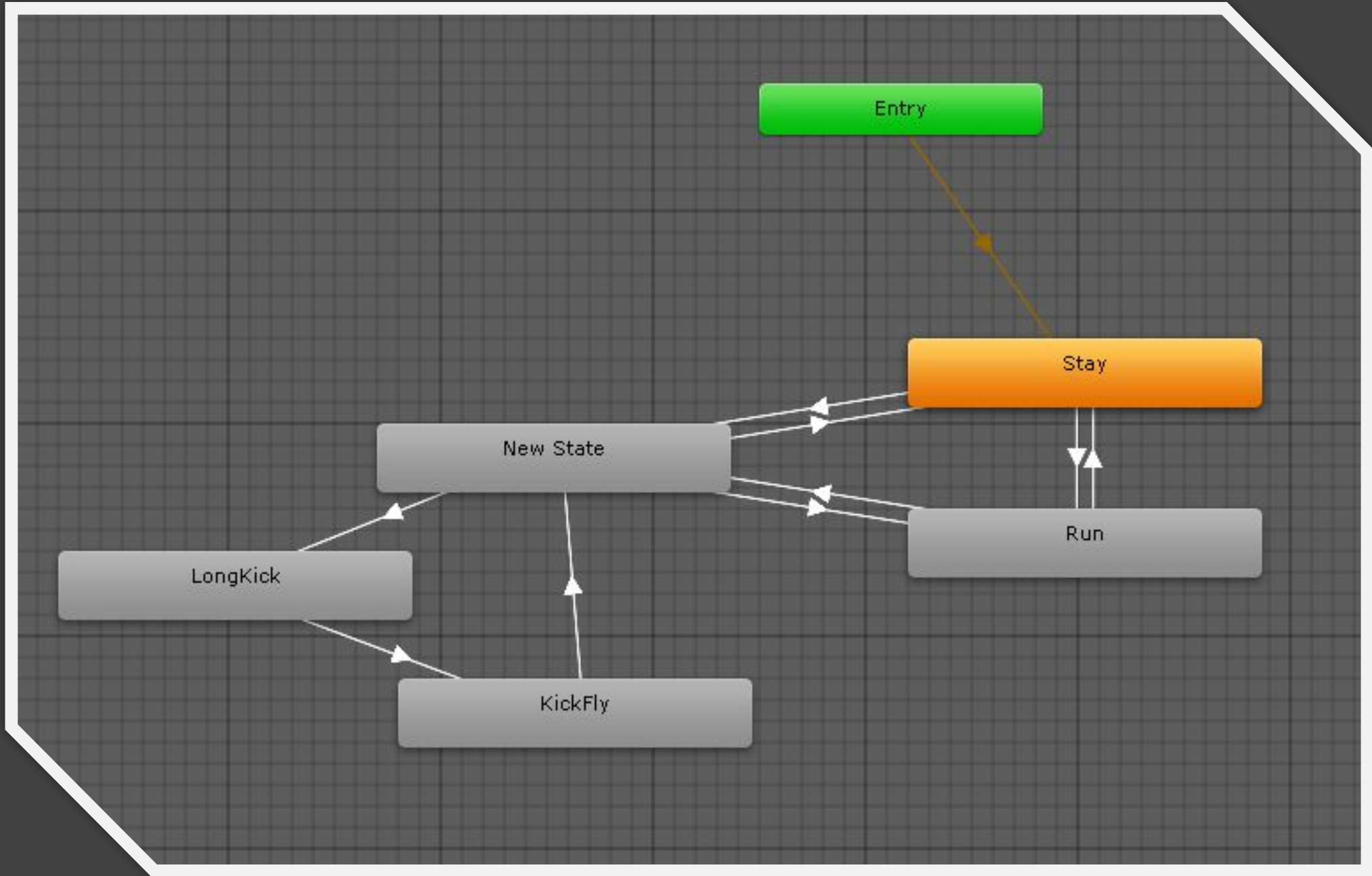
# Пример игровой анимации



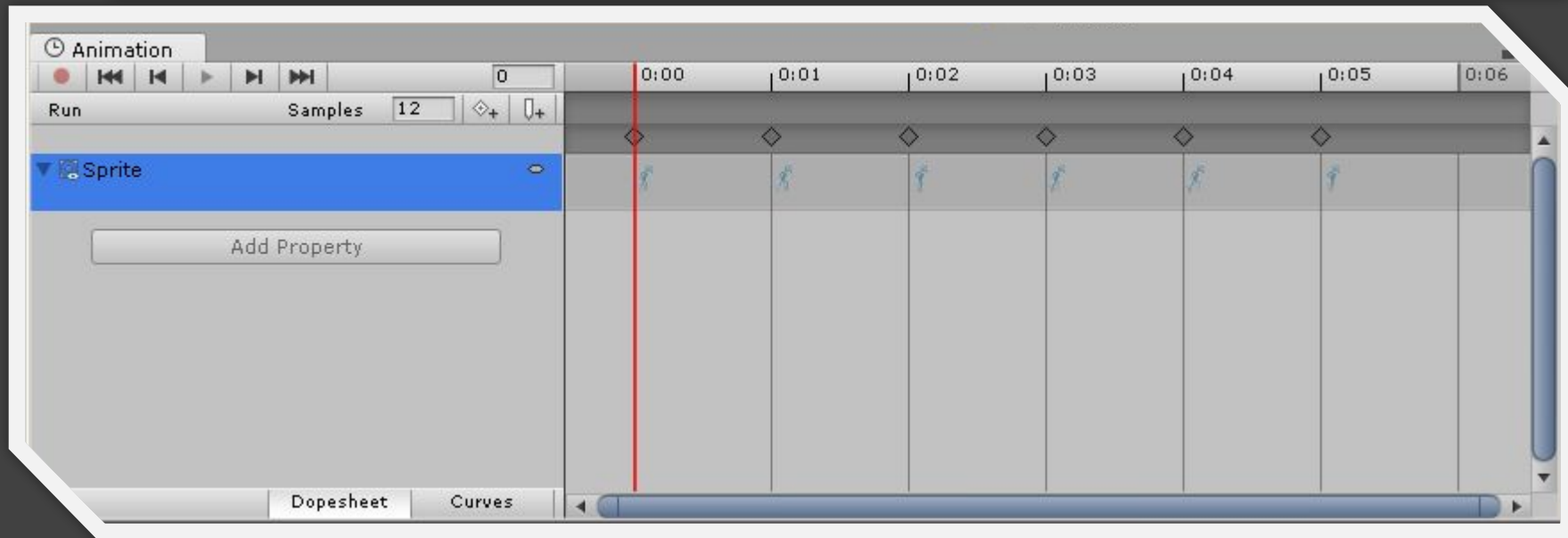
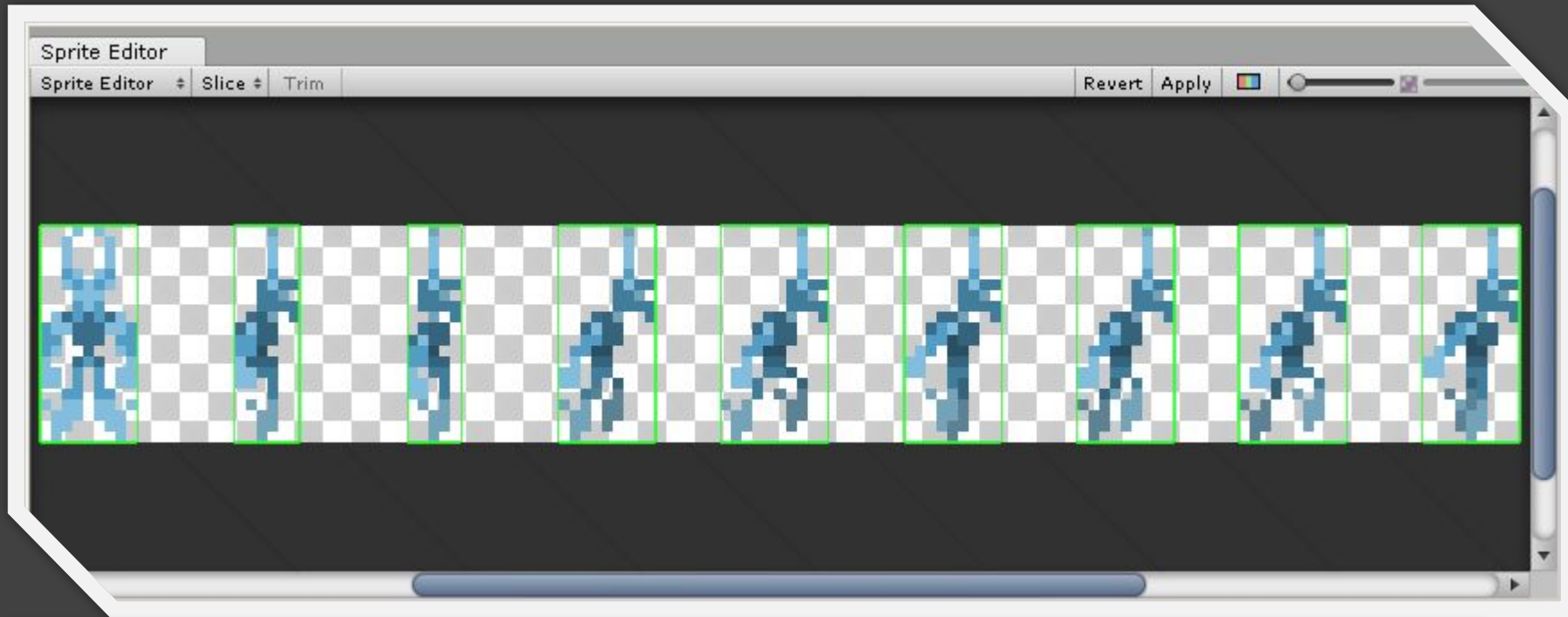








```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 ссылка: 0 public class PlayerAnimation : MonoBehaviour {
6
7
8     private Animator anim;
9     private Rigidbody2D rb;
10    public GameObject Character;
11
12
13    private bool stay = false, movement = false, jump = false, LongKick = false ;
14    private float KickTime, move;
15
16    ссылка: 0 void Start()
17    {
18        anim = GetComponent<Animator>();
19        rb = Character.GetComponent<Rigidbody2D>();
20    }
21
22
23    ссылка: 0 void Update ()
24    {
25        move = Character.GetComponent<Player>().move;
26        KickTime = Character.GetComponent<Player>().KickTime;
27
28        if (Input.GetMouseButton(0) && !LongKick && (move==1 || move==-1))
29        {
30            LongKick = true;
31            stay = false;
```







# Заключение