



Điều khiển nhân vật bằng Joystick - Mouse

Hàm thường dùng

playerControl player = GameObject.Find("Player").GetComponent<playerControl>();

- ▶ playerControl: file script của GameObject Player
- ▶ Find: lấy component playerControl của Player

GameObject.Find("Jump").GetComponent<Button>().onClick.AddListener(() => Jump());

- ▶ Đăng ký Hàm Jump() cho sự kiện click của Button "Jump"

Input.mousePosition

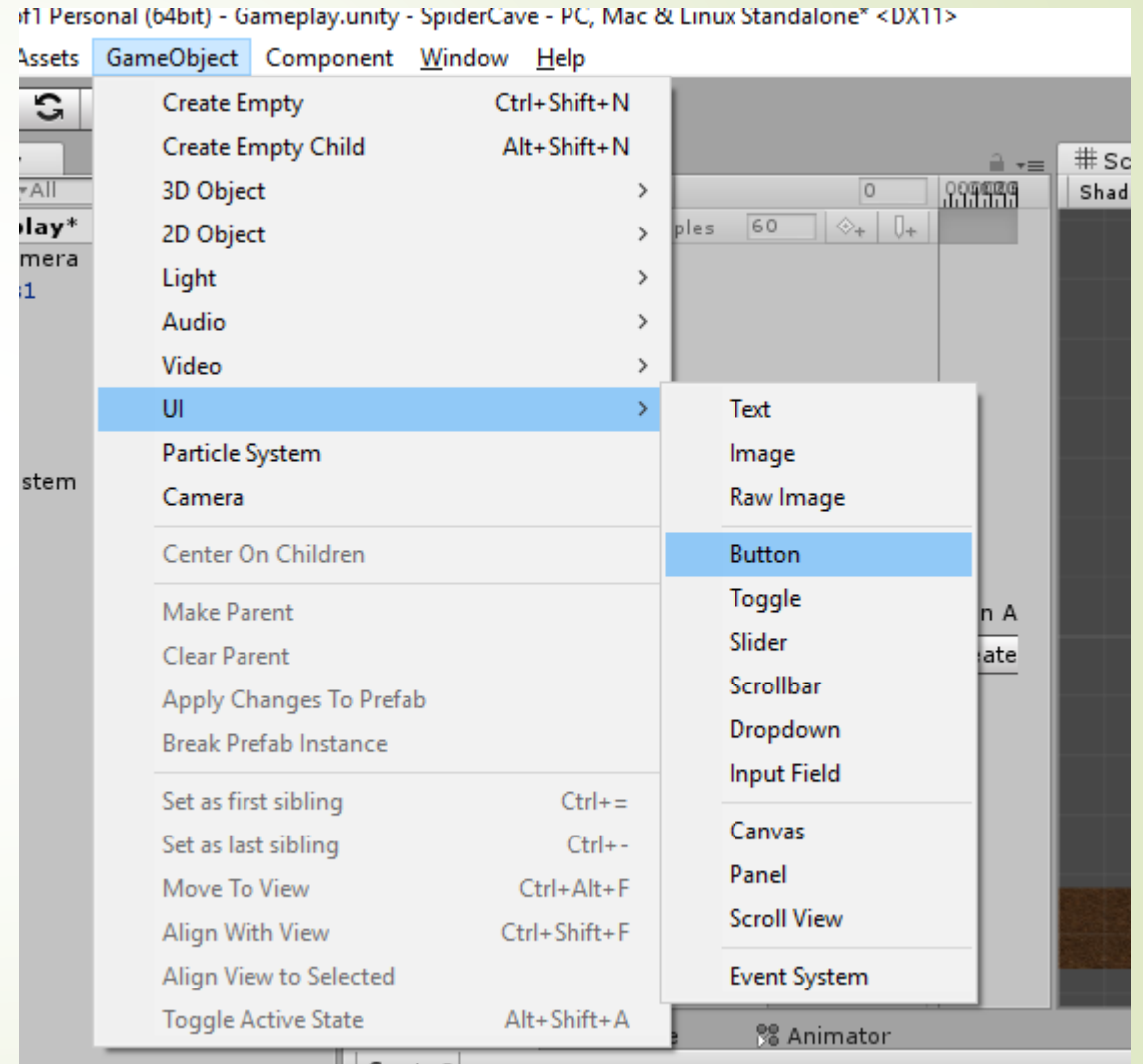
- ▶ Lấy vị trí chuột

Camera.main.ScreenToWorldPoint(Input.mousePosition);

Đưa vị trí chuột về cùng hệ trục tọa độ (Screen: tọa độ 0,0 ngay góc trái màn hình, World: tọa độ 0,0 ở chính giữa màn hình)

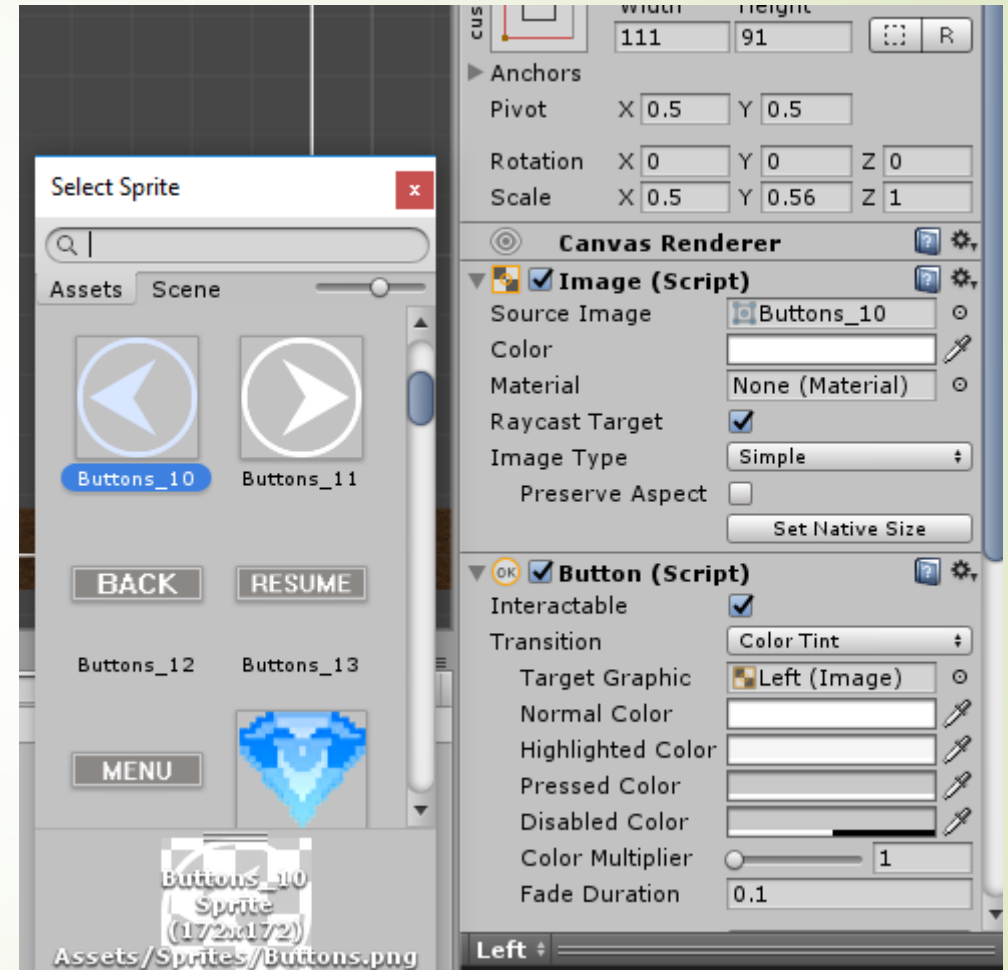
Joystick

- Tạo button Left- Right – Jump



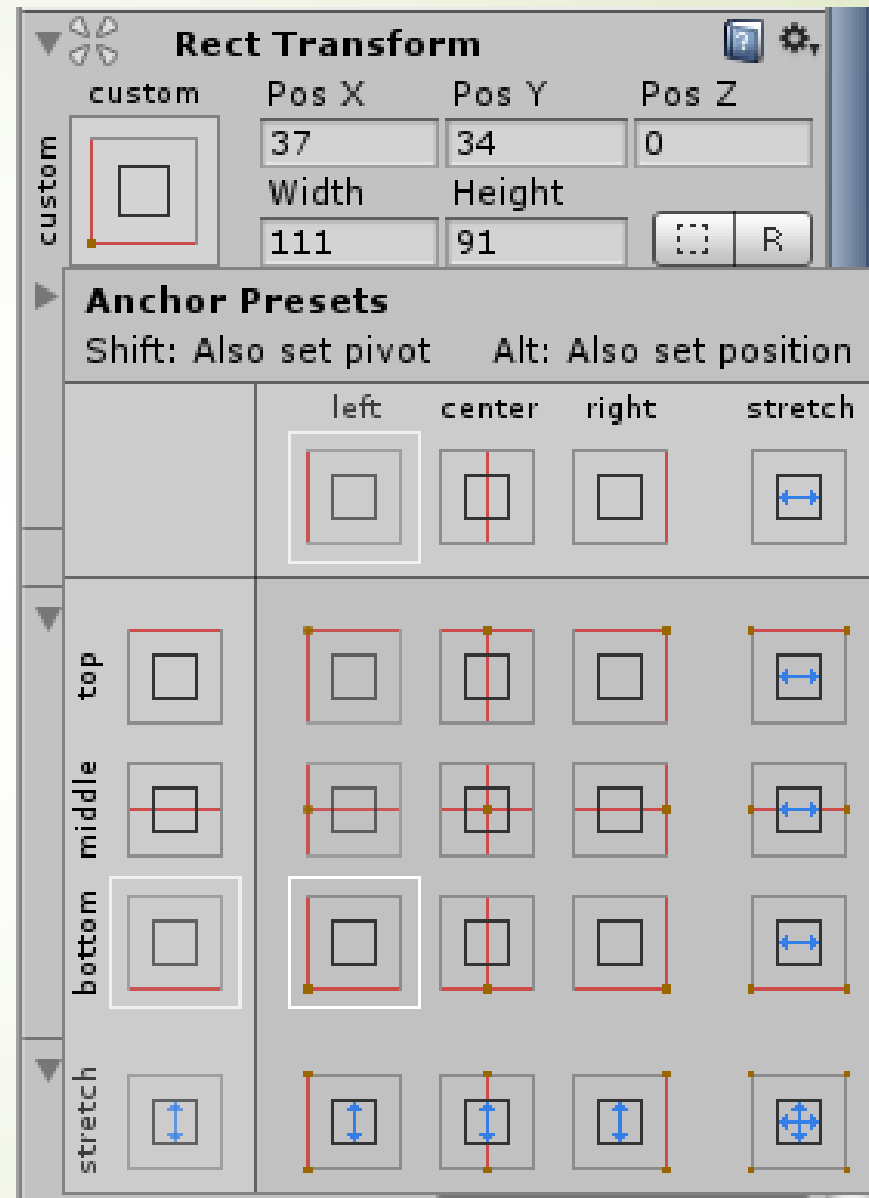
Joystick

- Gán Source Image vào Button



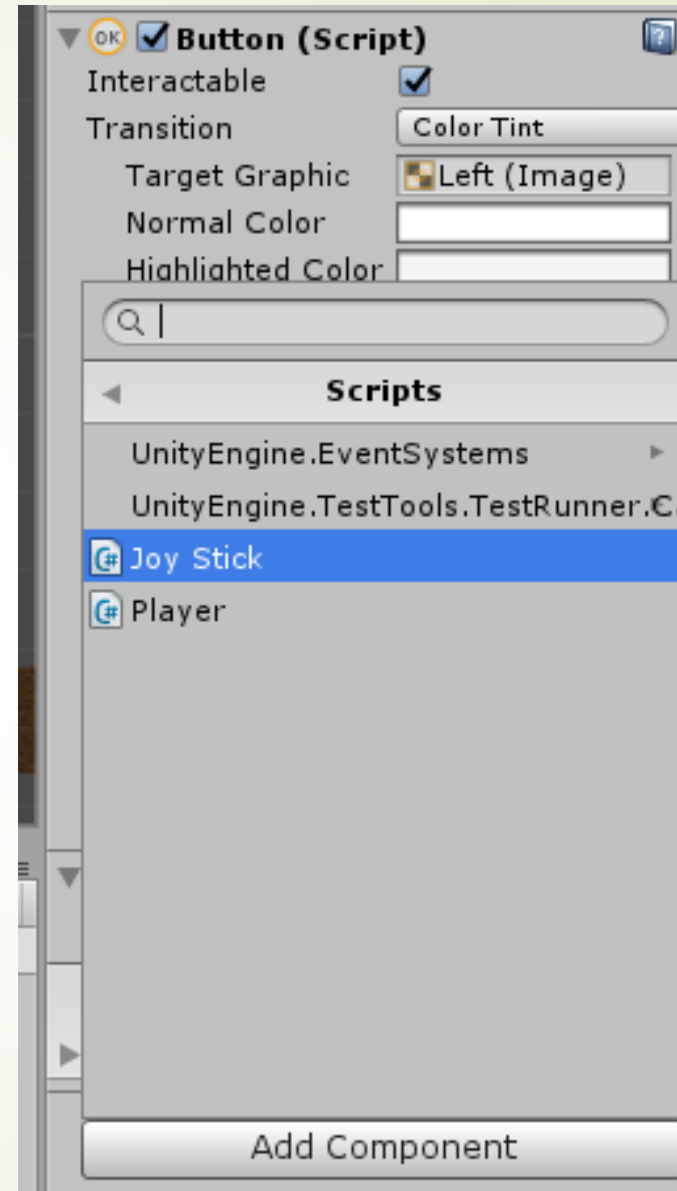
Joystick

- Thiết lập Anchors



Joystick

- Tạo file script Joystick.cs
- Gán Joystick script vào Button





Joystick script

```
using UnityEngine.EventSystems;
```

```
public class Joystick : MonoBehaviour, IPointerUpHandler, IPointerDownHandler {  
    private playerControl player;  
    void Awake () {  
        player = GameObject.Find("Player").GetComponent<playerControl>();  
    }  
    public void OnPointerUp(PointerEventData data)  
    {  
        player.StopMoving();  
    }  
}
```



Joystick script

```
public void OnPointerDown(PointerEventData data)
{
    if (gameObject.name == "Left")
        player.setMoveLeft(true);

    if (gameObject.name == "Right")
        player.setMoveLeft(false);
}
```




playerControl scripts

```
void FixedUpdate () {  
    PlayerJoyStick();  
}  
  
public void setMoveLeft(bool left)  
{  
    MoveLeft = left;  
    MoveRight = !left;  
}  
  
public void StopMoving()  
{  
    MoveLeft = false;  
    MoveRight = false;  
}
```



playerControl scripts

```
void PlayerJoyStick()  
{  
    float vel = Mathf.Abs(body.velocity.x);  
    float x = 0, y = 0;  
    if(MoveLeft)  
    { //.....  
    }  
    else if(MoveRight)  
    { //.....  
    }  
    else  
    {  
        //.....  
    }  
    body.AddForce(new Vector2(x, y));  
}
```



playerControl scripts

```
public void Jump()
{
    if (grounded)
    {
        grounded = false;
        body.AddForce(new Vector2(0, forceY));
    }
}
```

```
void Awake () {
    //...
```

```
GameObject.Find("Jump").GetComponent<Button>().onClick.AddListener(() => Jump());
}
```



Điều khiển nhân vật bằng - Mouse

Điều khiển player bằng Mouse

- ➡ Di chuyển player mỗi khi click chuột

```
private void Start()
{
    mousePos = transform.position;
}
void Update()
{
    if (Input.GetMouseButton(0))
    {
        mousePos = Camera.main.ScreenToWorldPoint(Input.mousePosition);
        mousePos = new Vector3(mousePos.x, mousePos.y, 0f);
    }
    Vector3 temp = Vector3.Lerp(transform.position, mousePos, 1 * Time.deltaTime);
    transform.position = temp;
}
```

Điều khiển player bằng Mouse

➡ Di chuyển player theo vị trí chuột

```
private void Start()
{
    mousePos = transform.position;
}
void Update()
{
    mousePos = Camera.main.ScreenToWorldPoint(Input.mousePosition);
    mousePos = new Vector3(mousePos.x, mousePos.y, 0f);
    Vector3 temp = Vector3.Lerp(transform.position, mousePos, 1 * Time.deltaTime);
    transform.position = temp;
}
```

Giới hạn đường biên di chuyển

```
private void Start()
{
    mousePos = transform.position;
}
void Update()
{
    mousePos = Camera.main.ScreenToWorldPoint(Input.mousePosition);
    mousePos = new Vector3(Mathf.Clamp(mousePos.x, minX, maxX),
Mathf.Clamp(mousePos.y, minY, maxY), 0f);
    Vector3 temp = Vector3.Lerp(transform.position, mousePos, 1 * Time.deltaTime);
    transform.position = temp;
}
```


Demo: Điều khiển Cừu bằng mouse tránh đạn do Sói bắn tự động sau mỗi 3 giây về phía cừu

