public class Settings

{

public const float fadeDuration=0.35f;

public const float fadeAlpha = 0.7f;

}

public class ItemFader : MonoBehaviour

{

private SpriteRenderer spriteRenderer;

private void Awake()

{

spriteRenderer = GetComponent<SpriteRenderer>();

}

public void FadeIn()

{

Color targetColor = new Color(1, 1, 1, 1);

spriteRenderer.DOColor(targetColor, Settings.fadeDuration);

}

public void FadeOut()

{

Color targetColor = new Color(1, 1, 1, Settings.fadeAlpha);

spriteRenderer.DOColor(targetColor, Settings.fadeDuration);

}

}使用 DOTween 插件来控制精灵的颜色过渡，从而实现物体的**淡入淡出效果**。FadeIn 方法用于从透明变为不透明，而 FadeOut 方法则是从不透明变为半透明或完全透明。

public class TriggerFader : MonoBehaviour

{

private void OnTriggerEnter2D(Collider2D collision)

{

ItemFader[] faders = collision.GetComponentsInChildren<ItemFader>();

if (faders!=null)

{

foreach (var target in faders)

{

target.FadeOut();

}

}

}

private void OnTriggerExit2D(Collider2D collision)

{

ItemFader[] faders = collision.GetComponentsInChildren<ItemFader>();

if (faders != null)

{

foreach (var target in faders)

{

target.FadeIn();

}

}

}

}

public class SwitchBounds : MonoBehaviour

{

private void Start()

{

SwitchConfinerShape();

}

private void SwitchConfinerShape()

{

PolygonCollider2D ConfinerShape2D = GameObject.FindGameObjectWithTag("BoundConfiner").GetComponent<PolygonCollider2D>();

CinemachineConfiner2D confiner2D = GetComponent<CinemachineConfiner2D>();

confiner2D.BoundingShape2D = ConfinerShape2D;

}

}