



DuckController

说明：鸭子控制器主要是用来控制鸭子初始化、移动、死亡等操作

重要属性：

```
public new Animation animation;
public PolygonCollider2D polygonCollider;
public SortingGroup sortingGroup;
public Transform duckTargetIcon;
public SpriteRenderer duckSpriteRenderer;

public bool isGameDuck;
public bool isTargetDuck;
private List<Vector2> physicsShape = new List<Vector2>();
private Vector2 generatePoint;
// 鸭子现在的朝向
private bool isLeft;
private bool isUp;
// 目标点位置
private float targetX;
private float targetY;
```

初始化方法

1. 根据传入的配置设置当前duckSpriteRenderer的sprite和localRotation
2. <设置碰撞体形状>

Update

1. 如果当期不是游戏鸭子或者鸭子已经死亡则退出
2. 让鸭子在场景中进行移动<Move>

Move

1. 更新鸭子x坐标：如果当前鸭子向左并且已到达或者超过目标点，则更新目标点targetX为当前场景范围内一个随机x值；向右类似
2. 更新鸭子y坐标：如果当前鸭子向上并且已到达或者超过目标点，则更新目标点targetY为当前场景范围内一个随机y值；向下类似
3. 根据目前鸭子的朝向更新方向 `transform.Translate(direction.normalized * ConfigManager.Instance.duckMoveSpeed * Time.deltaTime);`

设置碰撞体形状

1. 根据传入的sprite获得对应的物理形状 `sprite.GetPhysicsShape(0, physicsShape);`
2. 由上述步骤得到的结果设置Collider的Path `polygonCollider.SetPath(0, physicsShape);`

初始化菜单鸭子

1. 设置isTargetDuck=false、isGameDuck=false
2. 播放鸭子音效
3. 调用<初始化方法>设置sprite、rotation和碰撞体形状
4. 在菜单鸭中间设置靶心，并将菜单鸭sortingLayerName设置为"Duck2"
5. 将菜单鸭子的位置设置为场景下方，并<使用协程将菜单鸭移动到指定Y轴地点>形成动态效果

使用协程将菜单鸭移动到指定Y轴地点

1. 循环判断当前transform.position.y是否达到目标点，如果没有到达则不断增加transform.position.y `transform.position += new Vector3(0, Time.deltaTime * ConfigManager.Instance.duckMoveSpeed, 0);`
2. 当菜单鸭子移动到目标位置后播放菜单鸭子ready音效

初始化游戏鸭子

1. 根据传入的鸭子生成信息DuckGenerateInfo判断当前鸭子是否是目标鸭子（带有靶心的高分鸭子），如果是则播放音效提醒玩家
2. 调用<初始化方法>设置sprite、rotation和碰撞体形状
3. 设置isGameDuck=true、isTargetDuck=duckGenerateInfo.isTargetDuck
4. 根据当前是否为目标呀设置鸭子预制体上的靶心是否显示
5. 根据鸭子生成信息设置当前鸭子的sortingLayerName、transform.position
6. 随机获取移动地点作为当前鸭子目标地点

鸭子死亡退场效果

1. 播放鸭子退场音效
2. 将鸭子向下移动出场景外
3. 调用DuckManager<回收鸭子资源>

鸭子死亡方法

1. 标记isDead=true
2. 播放死亡动画
3. 停止之前所有协程（生成鸭子的协程等）
4. 使用协程调用<鸭子死亡退场效果>