

复制与销毁对象

大数据与物联网学院 邵亮

一、载入对象：Instantiate()

语法：GameObject . **Instantiate**(目标对象名);

- GameObject之方法。
- 复制一个对象到舞台。

复制操作的基础代码

- 1、定义一个对象（新元素） : `GameObject newObject;`
- 2、定义一个目标（复制谁？） : `GameObject tagObject=GameObject.Find("Cube");`
- 3、实施复制 : `newObject=GameObject.Instantiate(tagObject);`
- 4、为新元素定位: `newObject.transform.position=位置;`

案例：炮塔开炮

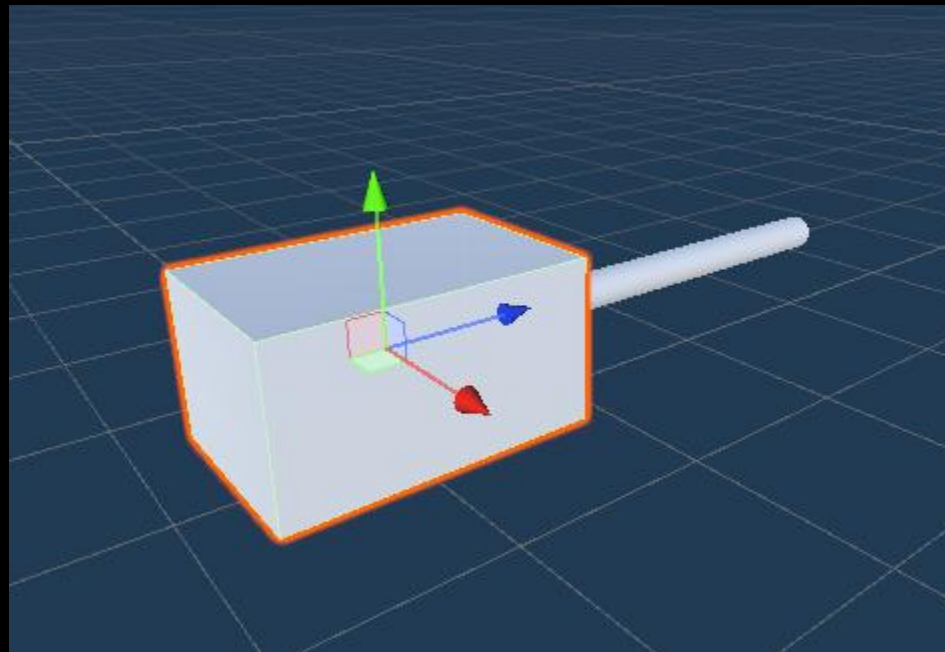
功能要求：

- 1、基于水平轴控制炮塔左右旋转；
- 2、按空格时，在炮口前方出现炮弹对象；
- 3、给炮弹施加力，使炮弹飞向前方（射击）。



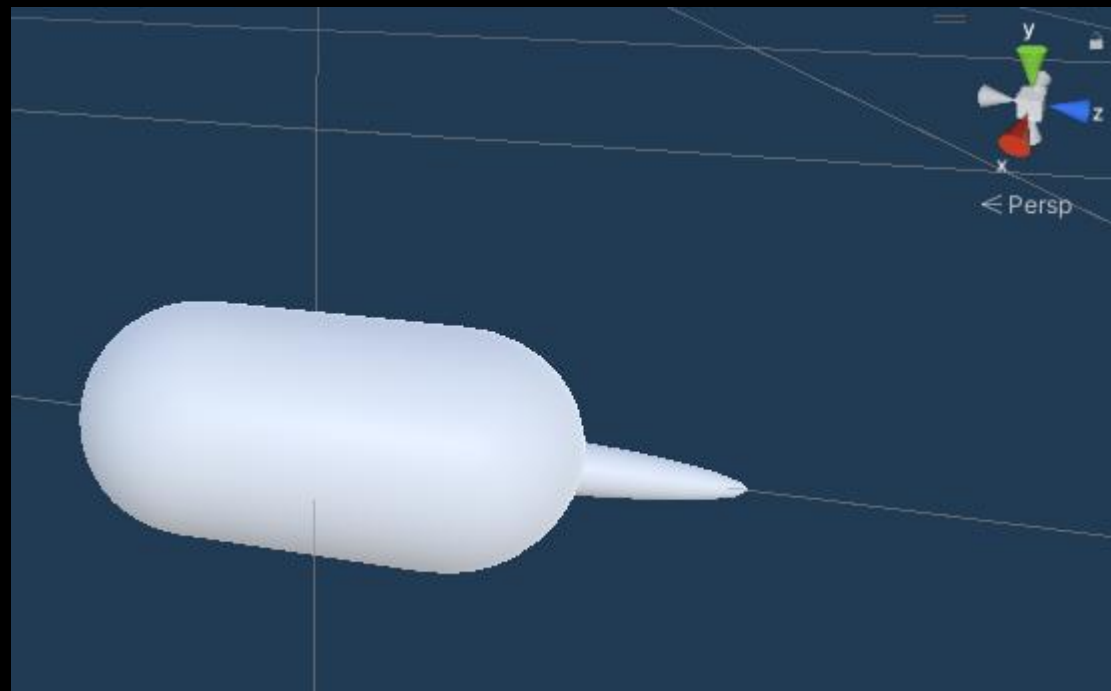
1、使用预制体制作炮塔

- (1) 创建预制体"tank";
- (2) 由立方体和圆柱体变形组合炮塔形状;
- (3) 注意各子元素的位置坐标、方向与预制体内部坐标系的关系。



2、使用预制体制作炮弹

- (1) 创建预制体“paodan”;
- (2) 由胶囊体和球体变形组合炮弹形状;
- (3) 注意各子元素的位置坐标、方向与预制体内部坐标系的关系。



3、编写脚本（加载到炮塔）

```
public class tankFire : MonoBehaviour
{
```

```
    public GameObject paodan;
```

```
    Rigidbody pdrd;
```

```
    GameObject pd,pGuan; // 定义炮弹和炮管对象;
```

```
    float rOffset;
```

```
    public float forceNum=5000f;
```

```
    void Start()
```

```
    {   pGuan=GameObject.Find("paoguan");   } // 获取炮管对象
```

```
    void Update()
```

```
    {
```

```
        rOffset=Input.GetAxis("Horizontal")*Time.deltaTime*50;
        this.transform.Rotate(0f,rOffset,0f);
```

// 炮塔基于水平轴输入旋转。

```
    if (Input.GetKeyDown(KeyCode.Space))
```

```
    {   // 复制炮弹并定位;
```

```
        pd = GameObject.Instantiate(paodan);
```

```
        pd.transform.position = pGuan.transform.position+pGuan.transform.up*1.5f;
```

```
        pd.transform.forward = this.transform.forward;
```

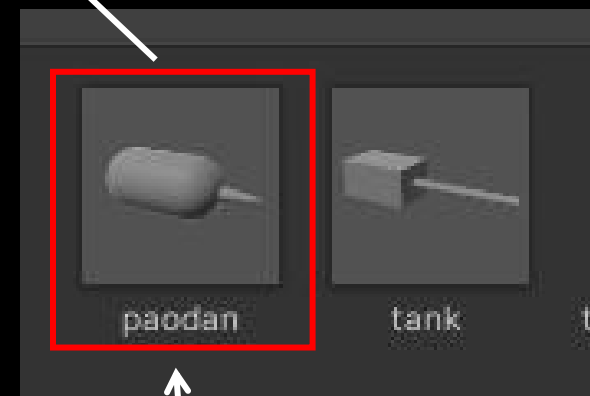
```
        // 给炮弹施加力;
```

```
        pdrd = pd.GetComponent<Rigidbody>();
```

```
        pdrd.AddRelativeForce(pd.transform.forward*forceNum);
```

```
    }
```

```
}
```



扩展自研：炮的俯仰

功能要求：

- 1、基于垂直轴控制炮管的俯仰；
- 2、按下空格，在对应的俯仰角发射炮弹。



二、销毁对象：Destroy()

语法：GameObject . Destroy(目标对象);

- GameObject之方法。
- 将指定对象从舞台上删除。

扩展自研：炮塔自毁

功能要求：

- 1、按下K键，炮塔从场景中消失；
- 2、炮塔消失时，播放爆炸烟雾粒子特效。

