

🎨 6. 克隆.clone()和复制.copy()

克隆 `.clone()` 、复制 `.copy()` 是threejs很多对象都具有的方法, 比如三维向量对象Vector3、网格模型Mesh、几何体、材质。

克隆 `.clone()`

克隆 `.clone()` 简单说就是复制一个和原对象一样的新对象,下面以三维向量对象Vector3给大家举例, 其他的threejs对象都可以参照类似的写法。

```
const v1 = new THREE.Vector3(1, 2, 3);
console.log('v1',v1);
//v2是一个新的Vector3对象, 和v1的.x、.y、.z属性值一样
const v2 = v1.clone();
console.log('v2',v2);
```

js

复制 `.copy()`

复制 `.copy()` 简单说就是把一个对象属性的属性值赋值给另一个对象,下面以三维向量对象Vector3给大家举例, 其他的threejs对象都可以参照类似的写法。

```
const v1 = new THREE.Vector3(1, 2, 3);
const v3 = new THREE.Vector3(4, 5, 6);
//读取v1.x、v1.y、v1.z的赋值给v3.x、v3.y、v3.z
v3.copy(v1);
```

js

Mesh克隆 `.clone()`

通过mesh克隆 `.clone()` 一个和mesh一样的新模型对象mesh2。

```
const mesh2 = mesh.clone();
mesh2.position.x = 100;
```

js

通过克隆 `.clone()` 获得的新模型和原来的模型共享材质和几何体

```
// 改变材质颜色，或者说改变mesh2颜色，mesh和mesh2颜色都会改变
// material.color.set(0xffff00);
mesh2.material.color.set(0xffff00);
```

js

几何体和材质克隆 `.clone()`

```
const mesh2 = mesh.clone();
// 克隆几何体和材质，重新设置mesh2的材质和几何体属性
mesh2.geometry = mesh.geometry.clone();
mesh2.material = mesh.material.clone();
// 改变mesh2颜色，不会改变mesh的颜色
mesh2.material.color.set(0xff0000);
```

js

练习： `mesh.position.copy()`

改变mesh的位置，使之位于mesh2的正上方(y)，距离100。

```
mesh.position.copy(mesh2.position); //1. 第1步位置重合
mesh.position.y += 100; //1. 第2步mesh在原来y的基础上增加100
```

js

练习： `mesh.rotation.copy()`

两个模型的姿态角度始终保持一样。

```
// 渲染循环
function render() {
    mesh.rotateY(0.01); // mesh旋转动画
    // 同步mesh2和mesh的姿态角度一样，不管mesh姿态角度怎么变化，mesh2始终保持同步
    mesh2.rotation.copy(mesh.rotation);
    renderer.render(scene, camera);
    requestAnimationFrame(render);
}
render();
```

js

