

🟡 1. 匀速动画(向量表示速度)

前面几节课给大家介绍过threejs的 `Vector3` 类可以表示物体的位置，也可以表示物体速度、位移等有方向的量。

下面给写一个物体匀速运动的动画案例，已知物体的初始位置、物体的速度，然后写一个物体的运动动画效果。

向量表示物体速度

```
const v = new THREE.Vector3(10,0,10); //物体运动速度
```

js

知识回顾

回顾下基础部分[1.11动画渲染循环](#)讲解过的知识点，通过 `clock` 类计算渲染循环两帧渲染间隔时间。

```
const clock = new THREE.Clock(); //时钟对象
// 渲染循环
function render() {
  const spt = clock.getDelta(); //两帧渲染时间间隔(秒)

  renderer.render(scene, camera);
  requestAnimationFrame(render);
}
render();
```

js

速度 x 时间计算位移

```
const v = new THREE.Vector3(10, 0, 10); //物体运动速度
const clock = new THREE.Clock(); //时钟对象
let t = 0;
const pos0 = mesh.position.clone(); //物体初始位置
```

js

```
// 渲染循环
function render() {
    const spt = clock.getDelta();//两帧渲染时间间隔(秒)
    t += spt;
    // 在t时间内，以速度v运动的位移量
    const dis = v.clone().multiplyScalar(t);
    // 网格模型初始位置加上t时间段内运动的位移量
    const newPos = pos0.clone().add(dis);
    mesh.position.copy(newPos);
    renderer.render(scene, camera);
    requestAnimationFrame(render);
}
render();
```

速度 x 间隔时间，然后累加计算位移

```
const v = new THREE.Vector3(10,0,10);//物体运动速度
const clock = new THREE.Clock();//时钟对象
// 渲染循环
function render() {
    const spt = clock.getDelta();//两帧渲染时间间隔(秒)
    // 在spt时间内，以速度v运动的位移量
    const dis = v.clone().multiplyScalar(spt);
    // 网格模型当前的位置加上spt时间段内运动的位移量
    mesh.position.add(dis);
    renderer.render(scene, camera);
    requestAnimationFrame(render);
}
render();
```

js

