

🎯 5. 相机动画(.position和.lookAt())

下面给大家讲解，通过相机对象 `Camera` 的 `.position` 属性和 `.lookAt()` 方法，实现一段相机动画。

相机运动动画

改变相机的位置 `.position`，三维场景在canvas画布上呈现不同的效果，如果连续改变相机的位置 `.position`，就可以获得一个动画效果。

课件案例源码是一个工厂模型，相机在空中俯视工厂，如果在渲染循环中不停地改变相机位置，这时候产生的视觉效果，就好比你在天上运动，看地面的效果。

```
// 渲染循环
function render() {
  camera.position.z -= 0.3; // 相机直线运动动画
  renderer.render(scene, camera);
  requestAnimationFrame(render);
}
render();
```

相机圆周运动

在渲染循环中，改变相机位置，在XOZ平面上绕着y轴圆周运动。

```
// 渲染循环
let angle = 0; // 用于圆周运动计算的角度值
const R = 100; // 相机圆周运动的半径
function render() {
  angle += 0.01;
  // 相机y坐标不变，在XOZ平面上做圆周运动
  camera.position.x = R * Math.cos(angle);
  camera.position.z = R * Math.sin(angle);
  renderer.render(scene, camera);
  requestAnimationFrame(render);
}
```

```
}  
render();
```

执行 `lookAt()` 计算相机视线方向

改变 `.position` 属性后，如果不执行 `.lookAt()` 方法，相机的观察方向默认不变。

如果你希望相机圆周运动的同时，改变相机视线方向，保持相机镜头始终指向坐标原点或其它位置，需要每次改变 `.position` 属性后，重新执行一遍 `.lookAt()` 方法

```
function render() {  
  angle += 0.01;  
  camera.position.x = R * Math.cos(angle);  
  camera.position.z = R * Math.sin(angle);  
  // .position改变，重新执行lookAt(0,0,0)计算相机视线方向  
  camera.lookAt(0,0,0);  
  requestAnimationFrame(render);  
}  
render();
```

← 4. 地图案例(包围盒、正投影)

6. 不同方向的投影视图 →