

## 🟡 11. 鼠标滑动改变视角(指针锁定模式)

前面改变相机左右和上下视角，用的是鼠标拖动方式，本节课给大家演示一个游戏、元宇宙项目中，常用的不用拖动，鼠标直接滑动改变相机玩家角色或相机视角。

### 知识点回顾

你可以把原来鼠标左键拖动改变视角的代码 `if (leftButtonBool) {}`，if条件去掉，只剩下鼠标滑动改变视角，你会发现并不是很好控制，有个问题，鼠标离开网页后，无法在旋转玩家视角，那么怎么解决？请看下面关于**指针锁定模式**的介绍。

```
js
let leftButtonBool = false; //记录鼠标左键状态
document.addEventListener('mousedown', (event) => {
  leftButtonBool = true;
});
document.addEventListener('mouseup', () => {
  leftButtonBool = false;
});

document.addEventListener('mousemove', (event) => {
  if (leftButtonBool) { //根据左键按下拖动才起作用
    player.rotation.y -= event.movementX / 600;
    cameraGroup.rotation.x -= event.movementY / 600;
  }
});
```

### 体验案例源码

你可以打开本节课案例源码，然后鼠标点击页面随便一个位置，就会进入指针锁定模式，人话说，这时候，你发现鼠标箭头不见了，鼠标可以上下左右无限滑动，如果你按下键盘左上角Esc按键，鼠标指针箭头恢复到原来状态。

请求指针锁定 `requestPointerLock`

鼠标滑动时候，会受到浏览器网页页面窗口范围限制，不能无限制移动，这时候可以通过执行 `requestPointerLock()` 请求指针锁定。

```
// 当鼠标左键按下后进入指针锁定模式(鼠标无限滑动)
addEventListener( 'mousedown', () => {
    document.body.requestPointerLock();//body页面指针锁定
});
```

js

你执行 `document.body.requestPointerLock();` 以后，意味着

`document.pointerLockElement` 属性，会拥有一个值 `document.body`

```
if(document.pointerLockElement == document.body){
    // 指针锁定模式下，才能执行的代码
}
```

js

## 在指针锁定模式下，改变玩家人姿态角度

进入指针模式后，才能根据鼠标位置控制人旋转

通过 `document.pointerLockElement` 判断web页面是否进入指针锁定模式。

鼠标点击页面进入指针锁定模式的时候，点击位置默认鼠标的坐标为原点，左右方向是x坐标 `.movementX` (单位像素)，上下方面是y坐标 `.movementY` 。

```
// 人和相机初始姿态正前方：沿着z轴正半轴方向
//鼠标左右移动，人绕y轴旋转
addEventListener('mousemove', (event) => {
    // 进入指针模式后，才能根据鼠标位置控制人旋转
    if (document.pointerLockElement == document.body) {
        // 鼠标左右滑动，让人左右转向(绕y轴旋转)，相机会父对象人绕左右转向
        //加减法根据左右方向对应关系设置，缩放倍数根据，相应敏感度设置
        person.rotation.y -= event.movementX / 500;
    }
});
```

js

## 退出指针锁定模式

执行 `document.exitPointerLock();` 可以退出指针锁定,或者键盘Esc退出指针锁定模式，就像你玩游戏一样。

---

← 10. 玩家角色左右运动(叉乘)

12. 快捷键切换第一、第三人称 →