

移动端事件

移动端

- Web 端
- APP 端
 - 原生 APP 安卓 IOS
 - 混合开发
 - RN和flutter
 - 内置页面 + 安卓/IOS 包壳
- 小程序

本节课目标

- 掌握 touchEvent
- 实现移动端幻灯片
- 初步认识 better-scroll

本节课知识点

- 移动端touch事件
 - touchstart
 - touchmove
 - touchend
 - touch 事件 和 mouse 事件的区别
 - 事件点透
 - mouse 事件的延迟问题
 - 阻止默认事件
 - 阻止 touchstart 事件带来的影响
 - 阻止 touchmove 事件带来的影响
- TouchEvent 对象详解
 - touches
 - targetTouches
 - changedTouches
- 案例：移动端滑屏切换的幻灯片
- orientationchange 监听横竖屏切换
- window.orientation 检测手机横竖屏
- devicemotion 监听手机加速度发生变化
 - acceleration 手机加速度检测
 - accelerationIncludingGravity 手机重力加速度检测
 - 案例: 方块移动

- IOS 和 安卓的兼容处理 `function getIos(){ var u = window.navigator.userAgent; return !!u.match(/\(i[^;]+;(U;)? CPU.+Mac OS X/); }`
- IOS 各个版本
 - 一般版本 https
 - 12.2 在 safari 设置
 - 13 1)

```
function permissionForShake() {
  // ios最新系统请求允许摇一摇 判断
  DeviceMotionEvent.requestPermission是否为函数
  if (typeof DeviceMotionEvent.requestPermission ===
  'function') {
    DeviceMotionEvent.requestPermission()
      .then(permissionState => {
        if (permissionState === 'granted') {
          // 权限允许
          window.addEventListener("devicemotion",
            () => {
              alert(2);
            },{once:true})
        }
      })
      .catch();
  } else {
    // 非ios系统或安卓
    window.addEventListener("devicemotion", () => {
      alert(3);
    },{once:true})
  }
}
```

2. 手动触发

- 扩展：函数防抖和函数节流
 - 函数防抖[debounce]
 - 希望函数只执行一次，哪怕我进行了多次调用

```
function debounce(fn,delay=200,isStart = false){
  if(typeof fn !== "function"){
    return console.error("请传入一个函数");
  }
  let timer = 0;
  let isFirst = true;
  return function(...arg){
    let _this = this;
```

```
        if(isFirst&&isStart){
            fn.apply(_this,arg);
            isFirst = false;
        }
        clearTimeout(timer);
        timer = setTimeout(() => {
            (!isStart)&&(fn.apply(_this,arg));
            isFirst = true;
        }, delay);
    }
}
```

- 函数节流[throttle]

- 让函数保持在一个可接受的固定频率执行

```
function throttle(fn,delay=200,start = true){
    if(typeof fn !== "function"){
        return console.error("请传入一个函数");
    }
    let timer = 0;
    return function(...arg){
        let _this = this;
        if(timer){
            return ;
        }
        start&&fn.apply(_this,arg);
        timer = setTimeout(() => {
            (!start)&&fn.apply(_this,arg);
            timer = 0;
        }, delay);
    }
}
```

- 案例：摇一摇功能实现

年后见小伙伴