

Web APIs 第二天

Dom事件基础







1. 掌握事件绑定处理和事件对象,完成常见网页交互





- ◆ 事件监听(绑定)
- ◆ 事件类型
- ◆ 事件对象
- ◆ 环境对象
- ◆ 回调函数
- ◆ 综合案例







事件监听 (绑定)

- 事件监听
- 拓展阅读-事件监听版本



1.1 事件监听

目标:能够给 DOM元素添加事件监听

● 什么是事件?

事件是在编程时系统内发生的**动作**或者发生的事情 比如用户在网页上**单击**一个按钮

● 什么是事件监听?

件

就是让程序检测是否有事件产生,一旦有事件触发,就立即调用一个函数做出响应,也称为绑定事件或者注册事

比如鼠标经过显示下拉菜单,比如点击可以播放轮播图等等



1.1 事件监听

● 语法:

元素对象.addEventListener('事件类型',要执行的函数)

- 事件监听三要素:
 - ▶ **事件源:** 那个dom元素被事件触发了,要获取dom元素
 - ▶ 事件类型: 用什么方式触发,比如鼠标单击 click、鼠标经过 mouseover 等
 - ▶ 事件调用的函数: 要做什么事



1.1 事件监听

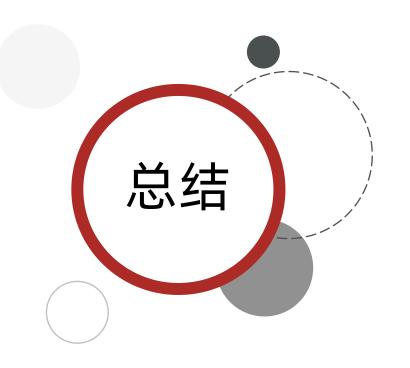
举例说明

```
<button>按钮</button>
<script>
  const btn = document.querySelector('.btn')
  // 修改元素样式
  btn.addEventListener('click', function () {
    alert('点击了~')
  })
  </script>
</action>
```

注意:

- 1. 事件类型要加引号
- 2. 函数是点击之后再去执行,每次点击都会执行一次





- 1. 什么是事件监听?
 - 就是让程序检测是否有事件产生,一旦有事件触发,就立即调用一个函数做出响应,也称为注册事件
- 2. 事件监听三要素是什么?
 - ▶ 事件源(谁被触发了)
 - ▶ 事件类型 (用什么方式触发,点击还是鼠标经过等)
 - ▶ 事件处理程序 (要做什么事情)

元素对象.addEventListener('事件类型', 要执行的函数)





京东点击关闭顶部广告

需求:点击关闭之后,顶部关闭

分析:

①:点击的是关闭按钮

②: 关闭的是父盒子

核心: 利用样式的显示和隐藏完成, display:none 隐藏元素 display:block 显示元素





小米 京东超级品牌日

全场至高优惠3000元 🙃







随机点名案例







• 随机点名案例

业务分析:

- ① 点击开始按钮随机抽取数组的一个数据,放到页面中
- ② 点击结束按钮删除数组当前抽取的一个数据
- ③ 当抽取到最后一个数据的时候,两个按钮同时禁用(写点开始里面,只剩最后一个数据不用抽了)

核心: 利用定时器快速展示, 停止定时器结束展示





事件监听 (绑定)

- 事件监听
- 拓展阅读-事件监听版本



1.2 事件监听版本

- DOM L0事件源.on事件 = function() { }
- DOM L2事件源.addEventListener(事件,事件处理函数)
- 区别:
 on方式会被覆盖, addEventListener方式可绑定多次, 拥有事件更多特性, 推荐使用



1.2 事件监听版本

● 发展史:

▶ DOM LO: 是 DOM 的发展的第一个版本; L: level

➤ DOM L1: DOM级别1于1998年10月1日成为W3C推荐标准

▶ DOM L2: 使用addEventListener注册事件

➤ DOM L3: DOM3级事件模块在DOM2级事件的基础上重新定义了这些事件,也添加了一些新事件类型





- ◆ 事件监听(绑定)
- ◆ 事件类型
- ◆ 事件对象
- ◆ 环境对象
- ◆ 回调函数
- ◆ 综合案例



2. 事件类型



鼠标触发

click 鼠标点击
mouseenter 鼠标经过
mouseleave 鼠标离开



表单获得光标

focus 获得焦点 blur 失去焦点



键盘触发

Keydown 键盘按下触发 Keyup 键盘抬起触发



表单输入触发

input 用户输入事件





轮播图点击切换

需求: 当点击左右的按钮,可以切换轮播图

分析:

①:右侧按钮点击,变量++,如果大于等于8,则复原0

②: 左侧按钮点击,变量一,如果小于0,则复原最后一张

③: 鼠标经过暂停定时器

④: 鼠标离开开启定时器





2. 事件类型



鼠标触发

click 鼠标点击
mouseenter 鼠标经过
mouseleave 鼠标离开



表单获得光标

focus 获得焦点 blur 失去焦点



键盘触发

Keydown 键盘按下触发 Keyup 键盘抬起触发



表单输入触发

input 用户输入事件





小米搜索框案例

需求: 当表单得到焦点,显示下拉菜单,失去焦点隐藏下来菜单

分析:

①: 开始下拉菜单要进行隐藏

②: 表单获得焦点 focus,则显示下拉菜单,并且文本框变色(添加类)

③: 表单失去焦点,反向操作





2. 事件类型



鼠标触发

click 鼠标点击
mouseenter 鼠标经过
mouseleave 鼠标离开



表单获得光标

focus 获得焦点 blur 失去焦点



键盘触发

Keydown 键盘按下触发 Keyup 键盘抬起触发



表单输入触发

input 用户输入事件





评论字数统计

需求: 用户输入文字,可以计算用户输入的字数

分析:

①:判断用输入事件 input

②:不断取得文本框里面的字符长度,文本域.value.length

③: 把获得数字给下面文本框

发一条友善的评论 [

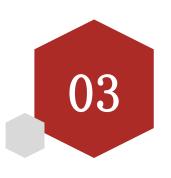
发布





- ◆ 事件监听(绑定)
- ◆ 事件类型
- ◆ 事件对象
- ◆ 环境对象
- ◆ 回调函数
- ◆ 综合案例





事件对象

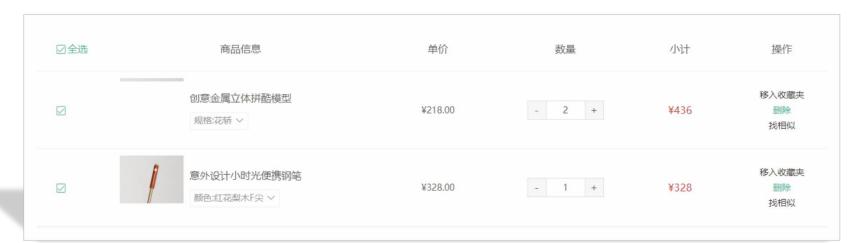
- 获取事件对象
- 事件对象常用属性



3.1 获取事件对象

目标: 能说出什么是事件对象

- 事件对象是什么
 - ▶ 也是个对象,这个对象里有事件触发时的相关信息
 - 例如: 鼠标点击事件中,事件对象就存了鼠标点在哪个位置等信息
- 使用场景
 - ▶ 可以判断用户按下哪个键,比如按下回车键可以发布新闻
 - ▶ 可以判断鼠标点击了哪个元素,从而做相应的操作







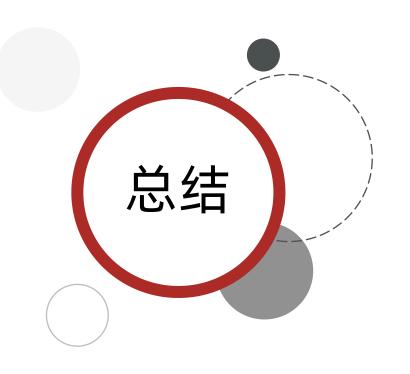
3.1 获取事件对象

- 语法:如何获取
 - > 在事件绑定的回调函数的第一个参数就是事件对象
 - ➤ 一般命名为event、ev、e

事件对象

元素.addEventListener('click', function (e) {

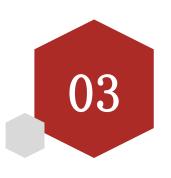




- 1. 事件对象是什么?
 - ▶ 也是个对象,这个对象里有事件触发时的相关信息
- 2. 事件对象在哪里?
 - ▶ 在事件绑定的回调函数的第一个参数就是事件对象

元素.addEventListener('click', function (e) {





事件对象

- 获取事件对象
- 事件对象常用属性



3.2 获取事件对象

目标:能够使用常见事件对象属性

- 部分常用属性
 - > type
 - □ 获取当前的事件类型
 - > clientX/clientY
 - □ 获取光标相对于浏览器可见窗口左上角的位置
 - > offsetX/offsetY
 - □ 获取光标相对于当前DOM元素左上角的位置
 - > key
 - □ 用户按下的键盘键的值
 - 现在不提倡使用keyCode





评论回车发布

需求:按下回车键盘,可以发布信息

分析:

①: 用到按下键盘事件 keydown 或者 keyup 都可以

②: 如果用户按下的是回车键盘,则发布信息

③: 让留言信息模块显示,把拿到的数据渲染到对应标签内部







- ◆ 事件监听(绑定)
- ◆ 事件类型
- ◆ 事件对象
- ◆ 环境对象
- ◆ 回调函数
- ◆ 综合案例







3. 环境对象

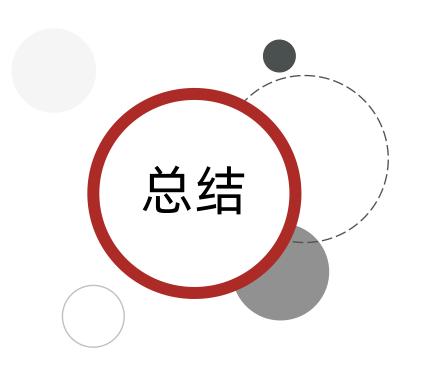
目标: 能够分析判断函数运行在不同环境中 this 所指代的对象

环境对象: 指的是函数内部特殊的变量 this,它代表着当前函数运行时所处的环境

作用:弄清楚this的指向,可以让我们代码更简洁

- ▶ 函数的调用方式不同, this 指代的对象也不同
- ▶ 【谁调用, this 就是谁】 是判断 this 指向的粗略规则
- ▶ 直接调用函数,其实相当于是 window.函数,所以 this 指代 window





- 1. 环境对象this是什么?
 - > 它代表着当前函数运行时所处的环境
- 2. 判断 this 指向的粗略规则是什么?
 - ▶ 【谁调用, this 就是谁】





- ◆ 事件监听(绑定)
- ◆ 事件类型
- ◆ 事件对象
- ◆ 环境对象
- ◆ 回调函数
- ◆ 综合案例







5. 回调函数

目标: 能够说出什么是回调函数

如果将函数 A 做为参数传递给函数 B 时,我们称函数 A 为回调函数

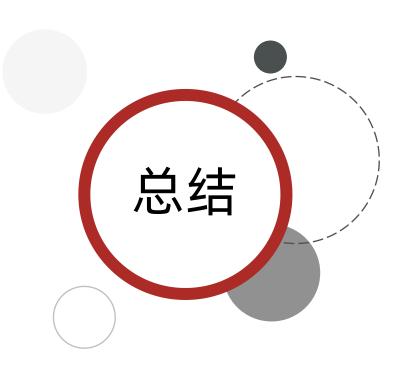
简单理解: 当一个函数当做参数来传递给另外一个函数的时候,这个函数就是回调函数

● 常见的使用场景:

```
function fn() {
    console.log('我是回调函数...')
}
// fn传递给了setInterval, fn就是回调函数
setInterval(fn, 1000)

box.addEventListener('click', function () {
    console.log('我也是回调函数')
})
```





1. 回调函数

- ▶ 把函数当做另外一个函数的参数传递,这个函数就叫回调函数
- ▶ 回调函数本质还是函数,只不过把它当成参数使用
- ▶ 使用匿名函数做为回调函数比较常见





- ◆ 事件监听(绑定)
- ◆ 事件类型
- ◆ 事件对象
- ◆ 环境对象
- ◆ 回调函数
- ◆ 综合案例





Tab栏切换

需求: 鼠标经过不同的选项卡, 底部可以显示 不同的内容







• Tab栏切换

分析:

- ①: 主要核心是类的切换, 设定一个当前类, 可以让当前元素高亮
- ②: 鼠标经过当前选项卡, 先移除其余元素身上的当前类, 而只给当前元素添加类,
- ③: 注意, 当前类只能有一个







- 1. 整理今天笔记, md或者Xmind都可以
- 2. 开始做测试题:
- 3. 练习、随机点名案例、评论发布案例
- 4. 练习tab栏切换案例 三遍
- 5. 作业

今天多一份拼搏,明日多一份欢笑



传智教育旗下高端IT教育品牌