

Web APIs 第二天

DOM-事件基础



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



目录

Contents

- ◆ 事件
- ◆ 高阶函数
- ◆ 环境对象
- ◆ 编程思想
- ◆ 综合案例

学习目标

Learning Objectives

1. 能给元素添加事件
2. 利用事件做常见的网页交互效果



事件

- 事件监听
- 拓展阅读-事件监听版本
- 事件类型

1. 事件

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？

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

- 语法：

```
元素.addEventListener('事件', 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： 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. 事件

目标：能够给 DOM 元素添加事件监听

- 事件监听

举例说明

```
// 1. 获取元素
let btn = document.querySelector('button')
// 2. 事件监听（注册事件）
btn.addEventListener('click', function() {
  alert('被点击了')
})
```

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



总结

1. 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
- 比如点击按钮 click

2. 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

3. 事件监听三要素是什么?

- 事件源 (谁被触发了)
- 事件 (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.addEventListener('事件', 要执行的函数)
```

案例

淘宝点击关闭二维码

需求：点击关闭之后，淘宝二维码关闭

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



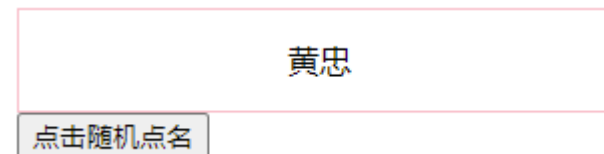
案例

随机点名

需求：点击按钮之后，随机显示一个名字，如果没有显示则禁用按钮

分析：

- ①：点击的是按钮
- ②：随机抽取一个名字
- ③：当名字抽取完毕，则利用 disabled 设置为 true





随机点名案例



案例

随机点名案例

需求：点击开始随机抽取，点击结束输出结果

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

核心：利用定时器快速展示，停止定时器结束展示



事件

- 事件监听
- 拓展阅读-事件监听版本
- 事件类型

1. 事件

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)

- 发展史:

- DOM L0 : 是 DOM 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

- 事件监听
- 拓展阅读-事件监听版本
- 事件类型

1. 事件

3. 事件类型

鼠标事件

鼠标触发

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

焦点事件

表单获得光标

focus 获得焦点
blur 失去焦点

键盘事件

键盘触发

Keydown 键盘按下触发
Keyup 键盘抬起触发

文本事件

表单输入触发

input 用户输入事件

案例

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

小米笔记本

全部商品

小米11

小米10S

小米笔记本

小米手机

黑鲨4

空调

案例

微博输入案例

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
- ③：把获得数字给下面文本框

有什么新鲜事想告诉大家？

我

I

1 / 200

发布

案例

全选文本框案例1

需求：用户点击全选，则下面复选框全部选择，取消全选则全部取消,文字对应变化

分析：

- ①：全选复选框点击，可以得到当前按钮的 checked
- ②：把下面所有的小复选框状态checked，改为和全选复选框一致
- ③：如果当前处于选中状态，则把文字改为取消， 否则反之

| <input checked="" type="checkbox"/> 取消 | 商品 | 商家 | 价格 |
|--|-------|----|--------|
| <input checked="" type="checkbox"/> | 小米手机 | 小米 | ¥ 1999 |
| <input checked="" type="checkbox"/> | 小米净水器 | 小米 | ¥ 4999 |
| <input checked="" type="checkbox"/> | 小米电视 | 小米 | ¥ 5999 |

案例

全选文本框案例2

需求：用户点击全选，则下面复选框全部选择，取消全选则全部取消,文字对应变化

分析：

- ①：遍历下面的所有的checkbox,添加点击事件
- ②：在事件内部,遍历所有的checkbox状态,只要有一个为false 就将全选状态设置为false ,把文字改为全选，并且直接return (退出循环)
- ③：在循环结束将全选的状态直接设置为true

| <input type="checkbox"/> 全选 | 商品 | 商家 | 价格 |
|-------------------------------------|-------|----|--------|
| <input checked="" type="checkbox"/> | 小米手机 | 小米 | ¥ 1999 |
| <input checked="" type="checkbox"/> | 小米净水器 | 小米 | ¥ 4999 |
| <input type="checkbox"/> | 小米电视 | 小米 | ¥ 5999 |

案例

购物车加减操作

需求：用户点击加号，则文本框+1，点击减号，则文本框-1，如果文本框为1，则禁用减号

分析：

- ①：给添加按钮注册点击事件, 获取表单的value, 然后自增
- ②：解除减号的disabled状态
- ③：给减号按钮添加点击事件, 获取表单的value, 然后自减
- ④：自减结束需要判断, 如果结果小于等于1 则添加上disabled状态





目录

Contents

- ◆ 事件
- ◆ 高阶函数
- ◆ 环境对象
- ◆ 编程思想
- ◆ 综合案例



高阶函数

1. 高阶函数

目标：掌握高阶函数用法，实现函数更多使用形式

高阶函数可以被简单理解为函数的高级应用，JavaScript 中函数可以被当成【值】来对待，基于这个特性实现函数的高级应用。

【值】就是 JavaScript 中的数据，如数值、字符串、布尔、对象等。

学习路径：

1. 函数表达式
2. 回调函数

1. 高阶函数

1.1 函数表达式

函数表达式和普通函数并无本质上的区别：

```
// 函数表达式与普通函数本质上是一样的
let counter = function (x, y) {
  return x + y
}
// 调用函数
let result = counter(5, 10)
console.log(result)
```

```
console.log(counter(5, 10))
let result = counter(5, 10)
```

- 普通函数的声明与调用无顺序限制，推荐做法先声明再调用
- **函数表达式必须要先声明再调用**

1. 高阶函数

目标：掌握高阶函数用法，实现函数更多使用形式

高阶函数可以被简单理解为函数的高级应用，JavaScript 中函数可以被当成【值】来对待，基于这个特性实现函数的高级应用。

【值】就是 JavaScript 中的数据，如数值、字符串、布尔、对象等。

学习路径：

1. 函数表达式
2. **回调函数**

1. 高阶函数

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)
```

```
setInterval(fn, 1000)
```

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

1. 高阶函数

1. 目标：掌握高阶函数用法，实现函数更多使用形式



思考

1. 函数表达式

- 函数也是【数据】
- 把函数赋值给变量

2. 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



目录

Contents

- ◆ 事件
- ◆ 高阶函数
- ◆ 环境对象
- ◆ 编程思想
- ◆ 综合案例



环境变量

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



目录

Contents

- ◆ 事件
- ◆ 高阶函数
- ◆ 环境对象
- ◆ 编程思想
- ◆ 综合案例



编程思想

4. 编程思想

排他思想

当前元素为A状态,其他元素为B状态

使用:

1. 干掉所有人

使用for循环

2. 复活他自己

通过this或者下标找到自己或者对应的元素



目录

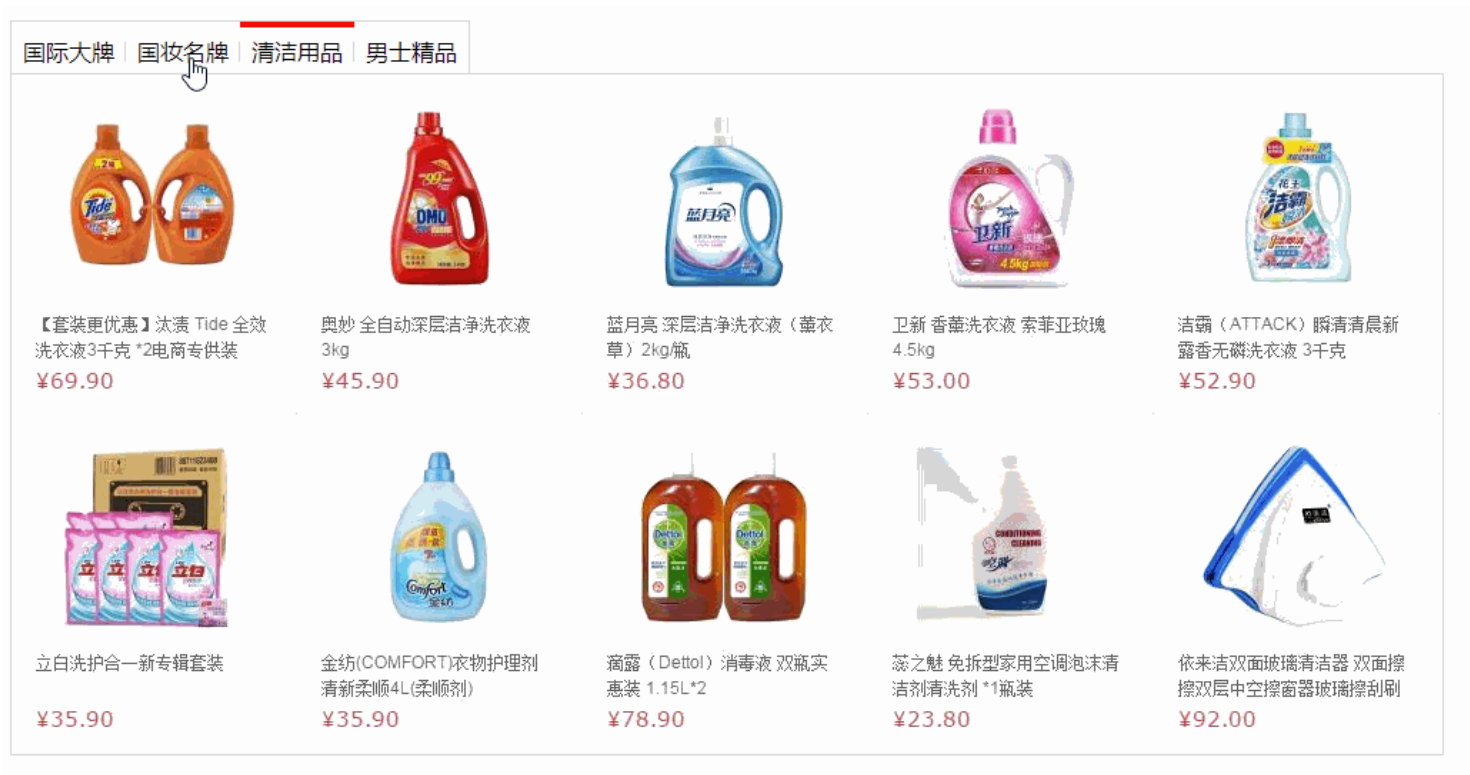
Contents

- ◆ 事件
- ◆ 高阶函数
- ◆ 环境对象
- ◆ 编程思想
- ◆ 综合案例

案例

Tab栏切换

需求：点击不同的选项卡，底部可以显示不同的内容





Tab栏切换

需求：点击不同的选项卡，底部可以显示 不同的内容

分析：

①：点击当前选项卡，当前添加类，其余的兄弟移除类， 排他思想

②：下面模块盒子全部隐藏，当前的模块显示



思考

1. 整理今天笔记
2. 练习 关闭二维码案例
3. 练习全选、购物车案例
4. 练习随机点名案例
5. 练习tab栏切换案例
6. 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