

事件核心





描述	属性/方法	效果		
事件监听	元素.addEventListener()	事件监听,事件绑定,事件注册		
	鼠标事件	click 鼠标点击 mouseenter 鼠标进入 mouseleave 鼠标离开		
事件类型	焦点事件	focus 获得焦点 blur 失去焦点		
	键盘事件	keydown 键盘按下 keyup 键盘抬起		
	文本事件 input	当表单value 被修改时触发		
事件对象	e.key	判断用户按下哪个键		
环境对象	this	谁调用,指向谁		





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
- ◆ 事件类型
- ◆ 事件对象
- ◆ 拓展知识
- ◆ 综合案例





# 事件监听

- 事件监听
- 回调函数
- 拓展阅读-事件监听版本



### 1.1 事件

- 以前写的代码都是自动执行的,我们希望一段代码在某个特定的时机才去执行,比如
- ▶ 点击按钮可以弹出警示框
- 比如鼠标经过显示下拉菜单等等。

#### 事件

- 事件是程序在运行的时候,发生的特定动作或者特定的事情
- 比如点击按钮、
- 比如鼠标经过菜单等等

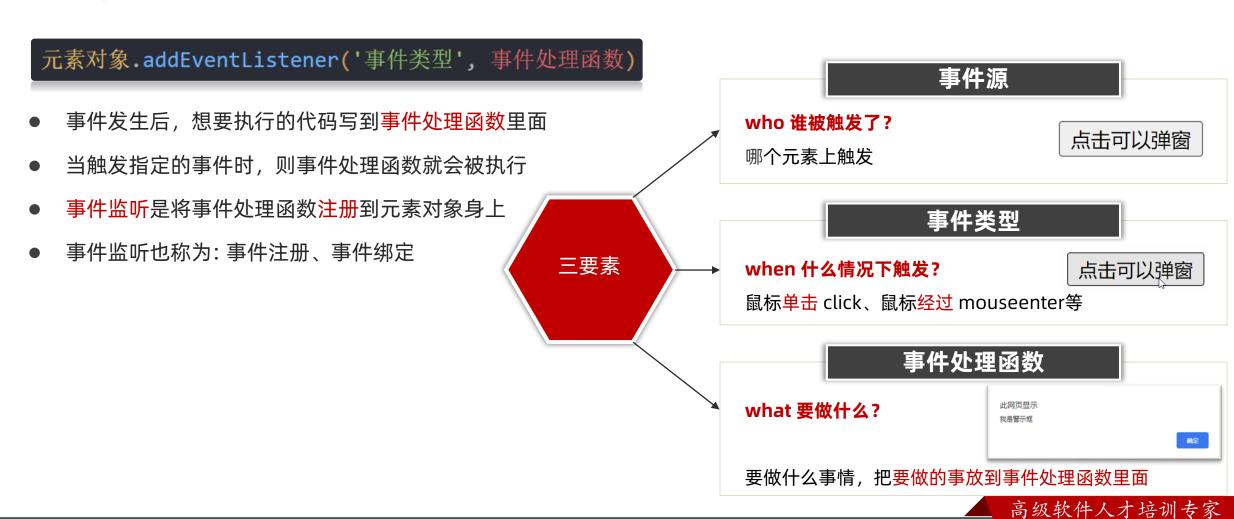
通常,当事件发生时,可以做些事情 比如点击按钮,可以 <mark>弹出警示框</mark> 比如鼠标经过某个盒子,可以 显示下拉菜单

那我们怎么实现事件触发时 JavaScript 可以执行一些代码呢?



#### 1.1 事件监听

● 语法:



事件处理函数



#### 1.1 事件监听

● 语法:

事件源

元素对象.addEventListener('事件类型',事件处理函数)

事件类型

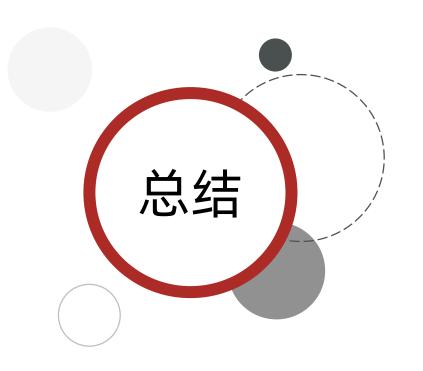
#### 举例说明

```
<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>
```

#### 注意:

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





- 1. 我们为什么需要事件?
  - ▶ 我们希望一段代码在某个特定的时机才去执行的时候
- 2. 事件监听的语法是什么?
  - ▶ 将事件处理函数注册到元素对象身上,当触发指定的事件时,则事件处理函数就会被执行
  - ▶ 也称为事件注册、事件绑定
- 3. 事件监听三要素是什么?
  - ▶ 事件源(谁被触发了)
  - 事件类型 (什么情况下触发,点击还是鼠标经过等)
  - 事件处理函数 (要做什么事情)

元素对象.addEventListener('事件类型',事件处理函数)



# 1 案例

### 点击关闭王者荣耀登录窗口

需求:点击关闭按钮之后,关掉登录提示盒子

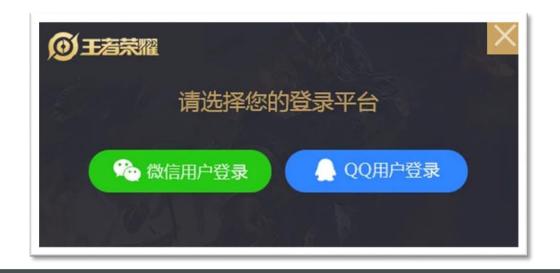
分析:

①:事件源:关闭按钮 a链接

②:事件类型:鼠标点击 click

③:事件处理程序: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







# 事件监听

- 事件监听
- 回调函数
- 拓展阅读-事件监听版本



#### 1.2 回调函数

回调函数: 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(回头调用的函数)

作用:完成某些特定任务

● 常见的使用场景:

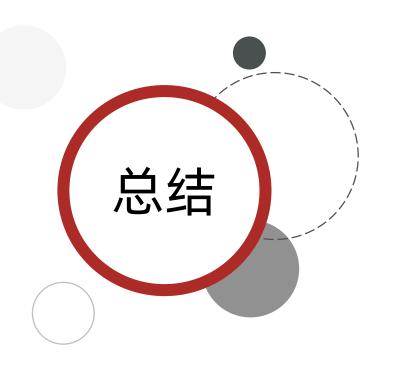
setInterval(function(){}, 1000)

box.addEventListener('click', function() {})

```
setInterval(function(){
    // 执行特定任务,比如可以切换轮播图片
}, 1000)
```

```
box.addEventListener('click', function() {
    // 执行特定任务,比如可以弹出警示框
})
```





#### 1. 回调函数

- ▶ 把函数当做参数传递给另外一个函数,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
- > 作用是完成某些特定任务

```
setInterval(function () {
  console.log('我是回调函数')
}, 1000)
```

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





# 事件监听

- 事件监听
- 回调函数
- 拓展阅读-事件监听版本



## 1.3 事件监听版本

L: level 标准、层次





#### ● 区别:

on 方式同名事件会被覆盖,addEventListener则不会,同时拥有事件更多特性,推荐使用



#### 1.3 事件监听版本

#### ● 发展史:

▶ DOM LO: 是 DOM 的发展的第一个版本, 使用 on.事件类型 来注册事件

➤ 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

```
btn.addEventListener('click', function() {
    alert('我是弹框')
})
```

```
btn.onclick = function() {
    alert('我是弹框')
}
```



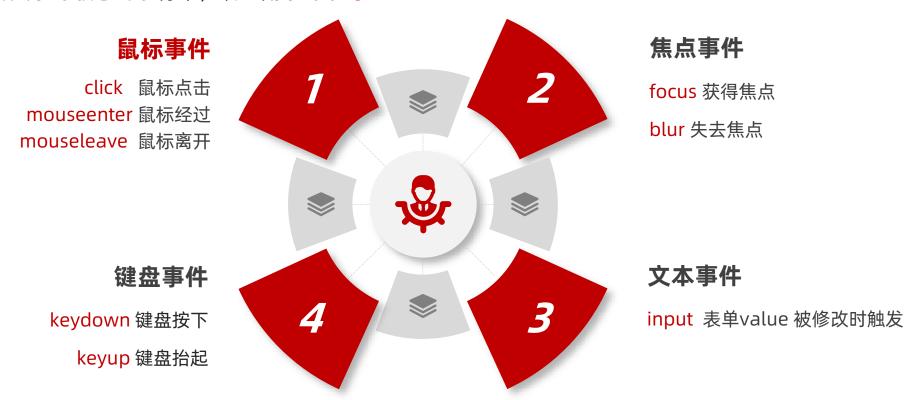


- ◆ 事件监听
- ◆ 事件类型
- ◆ 事件对象
- ◆ 拓展知识
- ◆ 综合案例



# 2. 事件类型

事件类型的大小写敏感的字符串,统一用小写字母







## 轮播图点击切换

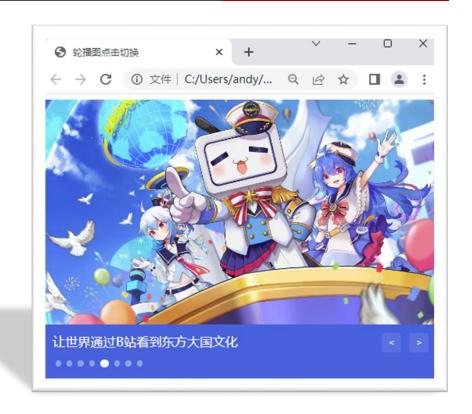
#### 需求分析:

①:右侧按钮点击,顺序切换图片

②: 左侧按钮点击, 倒序切换图片

③:自动播放图片

④:鼠标经过/离开可以继续/停止播放图片





# **宝** 案例

# 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++,无缝衔接顺序播放

②:左侧按钮点击,变量--,无缝衔接倒序播放

③: 开启定时器,自动播放

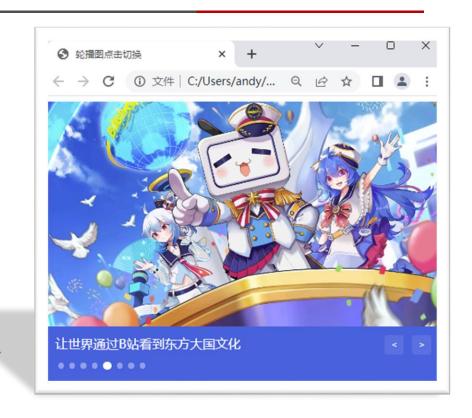
对象.click() 方法 是程序可以模拟用户点击事件,自动执行

不是所有的事件都能这样写

常见的有click()、blur()、focus()

④: 鼠标经过/离开可以开启/暂停播放

⑤: 抽取公共 toggle 函数,用于更换图片、文字等





<b>国</b> 案例
-------------

# 轮播图点击切换版

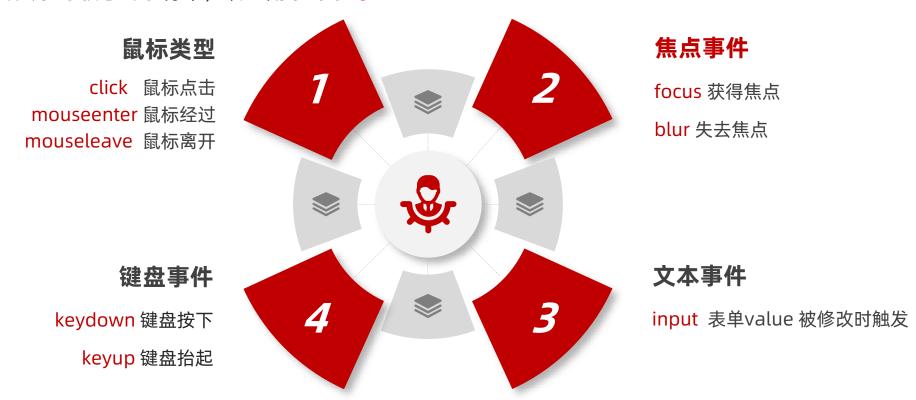
# 小技巧

7	6	5	4	3	2	1	0	-1	i
+	+	+	+	+	+	+	+	+	+
8	8	8	8	8	8	8	8	8	数组长度
15	14	13	12	11	10	9	8	7	
%	%	%	%	%	%	%	%	%	%
8	8	8	8	8	8	8	8	8	数组长度
7	6	5	4	3	2	1	0	7	•



# 2. 事件类型

事件类型的大小写敏感的字符串,统一用小写字母







### 小米搜索框案例

需求: 当表单得到焦点,显示下拉菜单,失去焦点隐藏下拉菜单

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,并且搜索框变色(添加类)

③:表单失去焦点 blur,则隐藏下拉菜单,搜索框复原原来颜色

# 小米笔记本 全部商品 小米11 小米10S 小米笔记本

黑鲨4

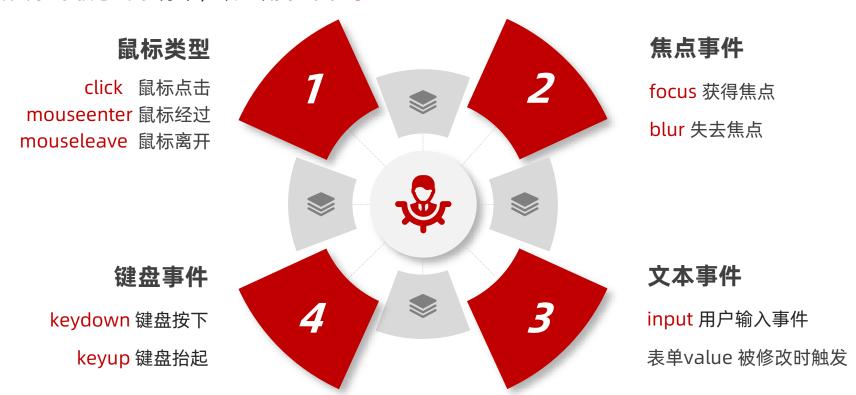
空调

小米手机



# 2. 事件类型

事件类型的大小写敏感的字符串,统一用小写字母





# 2. 事件类型

● keydown / keyup 和 input 事件区别

事件	触发时机	得到表单值		
keydown	按下键盘时触发	不带最后一次按键值 ab		
keyup	<mark>弹起</mark> 键盘时触发	输入内容 abc		
input	表单value发生变化时触发	输入内容 abc		

- 三者的执行顺序:
- keydown → input → keyup





### 统计用户输入字数案例

需求: 用户输入文字, 可以计算用户输入的字数

功能:

①:统计字数盒子,可以根据文本域获得失去焦点,实现显示和隐藏

②:根据用户输入的字数,写到统计字数盒子里面

发一条友善的评论 🍸

发布





### 统计用户输入字数案例

需求: 用户输入文字, 可以计算用户输入的字数

分析:

①:文本域获得焦点则显示统计 total 盒子,失去焦点则隐藏统计 total 盒子

②:文本域输入内容,使用input事件,不断取得字符长度(文本域.value.length)

③:把获得字符长度赋值给 total 字数统计盒子

发一条友善的评论 🍸

发布





- ◆ 事件监听
- ◆ 事件类型
- ◆ 事件对象
- ◆ 拓展知识
- ◆ 综合案例





# 事件对象

- · 获取事件对象
- 事件对象常用属性



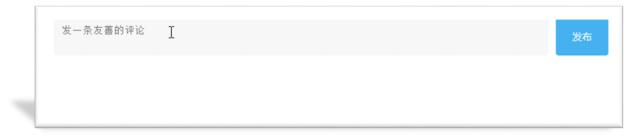
### 3. 事件对象(重要)

- 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,包含属性和方法
  - ▶ 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

事件对象是浏览器在触发时所创建的对象,这个对象中封装了事件相关的各种信息,通过事件对象可以获取到事件的详细信息

#### ● 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. 事件对象

- 语法:
  - > 注册事件中,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

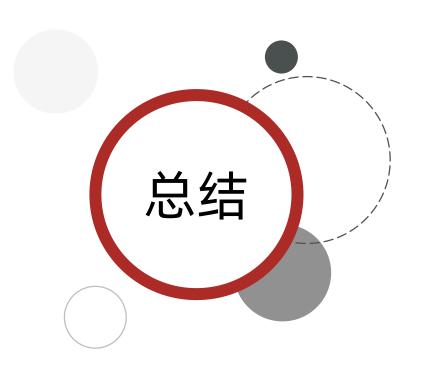
元素.addEventListener('click', function (e) {



# 3. 事件对象-常见属性

属性名	类型	说明
altKey	boolean	事件发生时,是否按下alt按键
ctrlKey	boolean	事件发生时,是否按下ctrl按键
shiftKey	boolean	事件发生时,是否按下shift按键
offsetX	number	事件发生时,鼠标相对于事件源的x坐标
offsetY	number	事件发生时,鼠标相对于事件源的y坐标
target	object	事件源对象
pageX	number	事件发生时,鼠标相对于网页的x坐标
pageY	number	事件发生时,鼠标相对于网页的y坐标
clientX	number	事件发生时,鼠标相对于视口的x坐标
clientY	number	事件发生时,鼠标相对于视口的y坐标
key	string	如果是键盘相关事件,则事件对象中包含该属性,表示键盘事件发生时,按下的是什么键。'Enter' 回车键





- 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<mark>第一个参数</mark>就是事件对象

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

- 3. 想要得到用户按下了键盘哪个键怎么来实现?
  - ➤ 事件对象.key 属性



# **国** 案例

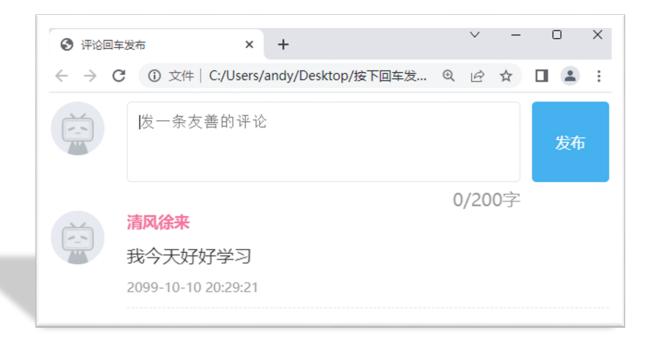
## 评论回车发布

需求:按下回车键,可以发布评论

功能:

①:按下回车,可以显示评论信息,并且评论内容显示到对应位置

②:输入完毕,文本域清空内容,并且字数复原为0





# **国** 案例

### 评论回车发布

需求:按下回车键盘,可以发布信息

分析:

①:通过键盘事件中的事件对象判断回车键,则显示评论模块,把内容填入

②:清空文本域就是修改文本域值为空字符串,同时把字数通过 innerText 复原为 0

③:点击发布按钮也可以发布评论,此处 回车可以采用 btn.click() 方法更简单







- ◆ 事件监听
- ◆ 事件类型
- ◆ 事件对象
- ◆ 拓展知识
- ◆ 综合案例



### 4.1 环境对象-this

环境对象:指的是函数内部特殊的 this,它指向一个对象,并且受当前环境影响

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



## 4.2 排他思想

- 是一种思路,目的是突出显示某个元素
- 比如,有多个元素,当鼠标经过时,只有<mark>当前元素</mark>会添加高亮样式,其余的元素移除样式



口诀: 注意顺序

①:排除其他人

②:保留我自己



描述	属性/方法	效果		
事件监听	元素.addEventListener()	事件监听,事件绑定,事件注册		
	鼠标事件	click 鼠标点击 mouseenter 鼠标进入 mouseleave 鼠标离开		
事件类型	焦点事件	focus 获得焦点 blur 失去焦点		
	键盘事件	keydown 键盘按下 keyup 键盘抬起		
	文本事件 input	当表单value 被修改时触发		
事件对象	e.key	判断用户按下哪个键		
环境对象	this	谁调用,指向谁		





- ◆ 事件监听
- ◆ 事件类型
- ◆ 事件对象
- ◆ 拓展知识
- ◆ 综合案例



# 1 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

业务:

①:顶部鼠标经过谁,谁会高亮显示

②:底部跟随切换不同内容







### Tab栏切换

业务1分析:鼠标经过谁,谁会高亮显示

①: 给5个链接注册鼠标经过事件(循环注册)

②:鼠标经过当前链接a,先移除其余元素身上的 active 类,而给当前元素添加 active 类(排他思想)





# 1 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

业务:

①:顶部鼠标经过谁,谁会高亮显示

②:底部跟随切换不同内容







## Tab栏切换

业务2分析:底部跟随切换不同内容

①:显示隐藏底部盒子通过 active 来控制显示隐藏

②: 底部其余 item盒子移除 active 类

③:先把5个item获取过来得到伪数组,然后利用 items[i] 选取对应盒子,然后添加 active





# 轮播图点击切换

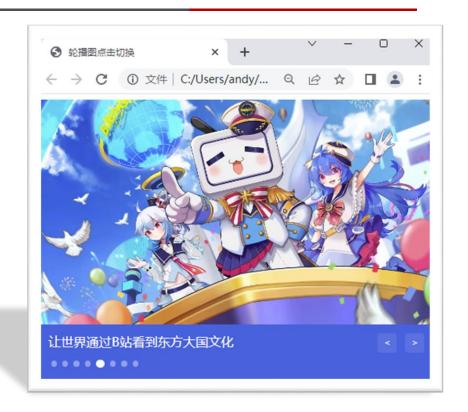
新增需求: 当点击小圆点, 可以切换轮播图

分析:

①:利用循环给所有小圆点注册点击事件

②:把当前点击的索引号赋值给信号量 i

③:调用公共 toggle 函数





传智教育旗下高端IT教育品牌