

# 一、常见浏览器 JS 对象常见 API 及用法

## 什么是浏览器对象模型

BOM：Browser Object Model（浏览器对象模型），浏览器模型提供了独立于内容的、可以与浏览器窗口进行滑动的对象结构，就是浏览器提供的 API

其主要对象有：

1. window 对象——BOM 的核心，是 js 访问浏览器的接口，也是 ES 规定的 Global 对象
2. location 对象：提供当前窗口中的加载的文档有关的信息和一些导航功能。既是 window 对象属性，也是 document 的对象属性
3. navigation 对象：获取浏览器的系统信息
4. screen 对象：用来表示浏览器窗口外部的显示器的信息等
5. history 对象：保存用户上网的历史信息

## Window 对象

windows 对象是整个浏览器对象模型的核心，其扮演着既是接口又是全局对象的角色

alert()  
confirm()  
prompt()  
open()  
onerror()  
setTimeout()  
setInterval()

### ■ 窗口位置

screenLeft  
screenTop  
screenX  
screenY  
moveBy(x,y)  
moveTo(x,y)

### ■ 窗口大小

innerWidth  
innerHeight

outerWidth  
outerHeight

resizeTo(width, height)

resizeBy(width, height)

- 定时器

setTimeout

setInterval

## Location 对象

提供当前窗口中的加载的文档有关的信息和一些导航功能。既是 window 对象属性，也是 document 的对象属性

location 对象的主要属性：

hash

host

hostname

href

pathname

port

protocol

search

location 的应用场景： 讲课时补充

## Navigation 对象

navigation 接口表示用户代理的状态和标识，允许脚本查询它和注册自己进行一些活动

## History 对象

history 对象保存着用户上网的历史记录，从窗口被打开的那一刻算起，history 对象是用窗口的浏览历史用文档和文档状态列表的形式表示。

go()

back()

forword()

length

## 二、详解浏览器事件捕获，冒泡

浏览器事件模型中的过程主要分为三个阶段：捕获阶段、目标阶段、冒泡阶段。

这里上课时看图

## 第三个参数

这里要注意addEventListener的第三个参数, 如果为true, 就是代表在捕获阶段执行。如果为false, 就是在冒泡阶段进行  
上课时看一下代码

## 阻止事件传播

- e.stopPropagation()

大家经常听到的可能是阻止冒泡, 实际上这个方法不只能阻止冒泡, 还能阻止捕获阶段的传播。

- stopImmediatePropagation()

如果有多个相同类型事件的事件监听函数绑定到同一个元素, 当该类型的事件触发时, 它们会按照被添加的顺序执行。如果其中某个监听函数执行了 event.stopImmediatePropagation() 方法, 则当前元素剩下的监听函数将不会被执行。

上课时看一下代码

## 阻止默认行为

e.preventDefault()

e.preventDefault()可以阻止事件的默认行为发生, 默认行为是指: 点击a标签就转跳到其他页面、拖拽一个图片到浏览器会自动打开、点击表单的提交按钮会提交表单等等, 因为有的时候我们并不希望发生这些事情, 所以需要阻止默认行为

上课时看一下代码

## 兼容性

attachEvent——兼容: IE7、IE8; 不支持第三个参数来控制在哪阶段发生, 默认是绑定在冒泡阶段

addEventListener——兼容: firefox、chrome、IE、safari、opera;

## 绑定事件的运用, 以及封装一个多浏览器兼容的绑定事件函数

大家常见的一个面试题可能是ul + li, 点击每个li alert对应的索引, 这里就给大家来写一下看看

- 先来给每个li绑定事件
- 再来写一个事件委托的方式

上课时看一下代码

## 封装一个多浏览器兼容的绑定事件函数

上课时看一下代码

## 三、ajax 及 fetch API 详解

### 1. XMLHttpRequest

### 2. fetch

- 默认不带cookie
- 错误不会reject
- 不支持超时设置
- 需要借用AbortController中止fetch

## 常见的浏览器请求/响应头/错误码解析

### request header

:method:

:path:

:scheme:

accept:

accept-encoding:

cache-control:

cookie:

origin:

referer:

user-agent:

### response header

access-control-allow-credentials:

access-control-allow-origin:

content-encoding:

content-type:

date:

set-cookie:

set-cookie:

status:

## status

200 get 成功  
201 post 成功  
301 永久重定向  
302 临时重定向  
304 协商缓存 服务器文件未修改  
400 客户端请求有语法错误，不能被服务器识别  
403 服务器受到请求，但是拒绝提供服务，可能是跨域  
404 请求的资源不存在  
405 请求的method不允许  
500 服务器发生不可预期的错误

## 发送请求的示例，以及封装一个多浏览器兼容的请求函数

上课时看一下代码