

- 十五、地理位置与本地存储
  - 地理位置
    - Navigator 对象
      - geolocation
  - 本地储存
    - Cookie
    - Storage
      - Storage API

# 十五、地理位置与本地存储

## 地理位置

### Navigator 对象

Navigator 对象包含有关浏览器的信息。

注释：没有应用于 navigator 对象的公开标准，不过所有浏览器都支持该对象。

Navigator 对象属性	
属性	描述
<a href="#">appName</a>	返回浏览器的代码名。
<a href="#">appMinorVersion</a>	返回浏览器的次级版本。
<a href="#">appName</a>	返回浏览器的名称。
<a href="#">appVersion</a>	返回浏览器的平台和版本信息。
<a href="#">browserLanguage</a>	返回当前浏览器的语言。
<a href="#">cookieEnabled</a>	返回指明浏览器中是否启用 cookie 的布尔值。
<a href="#">cpuClass</a>	返回浏览器系统的 CPU 等级。
<a href="#">onLine</a>	返回指明系统是否处于脱机模式的布尔值。
<a href="#">platform</a>	返回运行浏览器的操作系统平台。
<a href="#">systemLanguage</a>	返回 OS 使用的默认语言。
<a href="#">userAgent</a>	返回由客户机发送服务器的 user-agent 头部的值。
<a href="#">userLanguage</a>	返回 OS 的自然语言设置。

### geolocation

地理位置对象

navigator.geolocation

【单次定位请求】：getCurrentPosition(请求成功，请求失败，数据收集方式)

- 请求成功函数
  - 经度：coords.longitude
  - 纬度：coords.latitude
  - 准确度：coords.accuracy
  - 海拔：coords.altitude
  - 海拔准确度：coords.altitudeAccuracy
  - 行进方向：coords.heading
  - 地面速度：coords.speed
  - 时间戳：new Date(position.timestamp)
- 请求失败函数
  - 失败编号：code
  - 0：不包括其他错误编号中的错误
  - 1：用户拒绝浏览器获取位置信息：PERMISSION\_DENIED
  - 2：尝试获取用户信息，但失败了：POSITION\_UNAVAILABLE
  - 3：设置了timeout值，获取位置超时了：TIMEOUT
- 数据收集：json的形式
  - enableHighAccuracy：更精确的查找，默认false
  - timeout：获取位置允许最长时间，默认infinity
  - maximumAge：位置可以缓存的最大时间，默认0
- 多次定位请求【如果位置不停在改变，就需要多次请求】：watchPosition(像setInterval)
  - 移动设备有用，位置改变才会触发
  - 配置参数：frequency 更新的频率
- 关闭更新请求：clearWatch(像clearInterval)

以上的这些呢，都是使用的Google的原生的，所以，你可能需要翻墙。并且原生的有一些功能没有实现，所以。我们来使用高德地图或者是百度地图开发。

## 本地储存

### Cookie

数据存储到计算机中，通过浏览器控制添加与删除数据

- Cookie的特点
  - 存储限制：域名100个cookie,每组值大小4KB
  - 客户端、服务器端，都会请求服务器（头信息）
  - 页面间的cookie是共享
- cookie的作用
  - 页面用来保存信息
  - 同一个网站中所有页面共享一套cookie

- cookie的使用  
设置cookie 过期时间expires  
读取cookie 字符串分割  
删除cookie 已经过期

## Storage

- sessionStorage  
sessionStorage是用来做临时性存储的。  
sessionStorage的时效只存在于标签页存在的时间，一旦标签被关闭了，sessionStorage存储的数据也会被删除掉。
- localStorage  
永久存储（可以手动删除数据）
- Storage的特点  
存储量限制（5M）  
客户端完成，不会请求服务器处理  
sessionStorage数据是不共享、localStorage共享

## Storage API

- setItem():  
设置数据，key\value类型，类型都是字符串  
可以用获取属性的形式操作
- getItem():  
获取数据，通过key来获取到相应的value
- removeItem():  
删除数据，通过key来删除相应的value
- clear():  
删除全部存储的值
- 存储事件:  
当数据有修改或删除的情况下，就会触发storage事件  
在对数据进行改变的窗口对象上是不会触发的
- Key : 修改或删除的key值，如果调用clear(),key为null
- newValue : 新设置的值，如果调用
- removeStorage(),key为null
- oldValue : 调用改变前的value值
- storageArea : 当前的storage对象
- url : 触发该脚本变化的文档的url  
注：session同窗口才可以

