

HTML5程序设计基础

第六章 Web存储



主要内容

01

存储简介

02

LocalStorage使用方法

03

SessionStorage使用方法

04

Web Storage实例

01

存储简介



存储简介

- 在用户不登录帐户的情况下：
- 如何记录浏览历史？
 - 如何记录用户设置的首选项？
 - 如何将和用户关联的数据存储？

将数据存储在客户端（浏览器）上

最近浏览 [更多](#)



CITIZEN西铁城机械不
锈钢表带男士手表NH
¥3325.00



CITIZEN西铁城机械不
锈钢表带男士手表NH
¥3040.00



测试驱动的JavaScript
开发
¥54.10



代码质量
¥66.80

InfoQ读者们... QCon全球企业开发大会（北京站）2013 8折优惠开始抢购！团购5人以上优惠多多！

开放平台进行时、知名网站案例分析、架构设计模式与实践、大数据与NoSQL、云计算迷思、可编程的Web、Java .Net for Experts、前后端跨界之NodeJS、自动化运维、优秀测试实践分析、函数式编程、深入移动开发、机器学习与推荐算法、“硅谷”传奇、沙盘演练、用户体验与产品设计、构建高效能团队、持续集成与持续交付。

QCon

立即关闭

X

感谢通知

X

不再显示

X



大黄蜂 男女羽毛球鞋
¥158.00
月销 373 笔



正品羽毛球鞋 女鞋 特
¥798.00
月销 6 笔



假一罚三 蕉风正品 羽
¥260.00



特价 YONEX尤尼克斯
¥328.00
月销 258 笔



2011款英联0831D专
¥189.00
月销 3 笔

存储简介

🎨 关于 “存储在浏览器上的数据”，你想到了什么？

- Cookie
- Cookie 的工作原理？
- 使用 Cookie 做 Web 存储可以吗？

Cookie

Cookie

- 是 HTML4 中在客户端存储简单用户信息的一种方式，使用**文本**来存储信息。
- 程序使用 Cookie 时，服务器端就会发送 Cookie 到客户端，由 Web 服务器在 HTTP 响应消息文件中附带传递给浏览器一些数据。
- 下一次页面请求时，浏览器会把 Cookie 发送到服务器。
- **保存用户信息、用户设置、密码记忆。**

Cookie

包含的属性：

- **name**：表示 Cookie 的名称。
- **expires**：指定 Cookie 在删除之前要在用户的计算机上保留多长时间。默认 Cookie 只对当前浏览器会话有效。
- **path**：决定 Cookie 对于服务器上的其他网页的可用性。
- **domain**：设置相同域的多台服务器共享一个 Cookie。
- **secure**：表示 Cookie 只能通过使用 HTTPS 或其他安全协议的 Internet 连接来传输。

Cookie

Cookie的缺点

- 使用简单的文本存储数据，安全性差。保存在客户端浏览器，容易被黑客窃取。
- 存储容量有限，只有 4KB 的存储空间。
- 每次访问服务器，都会来回传送 Cookie，浪费带宽，尤其是使用移动设备时。
- 客户端操作 Cookie，非常复杂。

Cookie

HTML5 : Web存储

Cookie是不可以或缺的

- Cookie 的作用是与服务器进行交互，作为 HTTP 规范的一部分而存在，而 Web Storage 是为了在本地“存储”数据而生。

Web存储简介

HTML5 Web 存储的优点：

- 存储空间更大，浏览器至少提供 5M 的空间
- 数据仅仅存储在本地，不会自动向服务器发送
- 提供丰富的接口，方便的操作数据
- 独立的存储空间，每个域都有各自存储空间，不会造成混乱



Cookie



Web存储

域：文件的来源地址，由协议、域名、端口组成。例如：
`http://xxx.com:8010`

Web存储提供两种存储机制

LocalStorage

- 将数据保存在客户端本地的硬件设备中，即使浏览器被关闭了，该数据仍然存在，下次打开浏览器访问网站时仍然可以继续使用。

Session Storage

- 将数据保存在session对象中。session 是指用户在浏览某个网站时，从进入网站到浏览器关闭所经过的这段时间。session 对象可以用来保存在这段时间内所要求保存的任何数据。当用户关闭浏览器窗口后，数据会被删除。

sessionStorage为临时保存，而localStorage为永久保存。

浏览器支持

浏览器	说明
IE	8.0及以上的版本支持
Firefox	3.0及以上的版本支持
Opera	10.5及以上的版本支持
Chrome	3.0及以上的版本支持
Safari	4.0及以上的版本支持

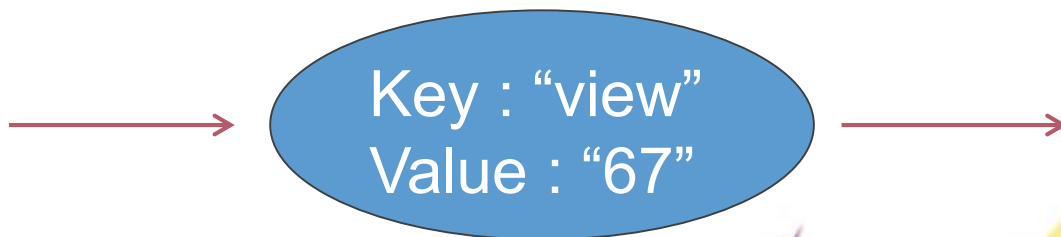
02

LocalStorage使用方法

Web Storage用法

数据的组织形式：

- 存储是基于**键/值对**的形式存储的，每个键值对称为一个**项 (item)**。
存储和检索数据都是通过指定的键名。
- 键名的类型是**字符串**类型。
- 值可以是包括字符串、布尔值、数值在内的**任意** JavaScript 支持的类型。

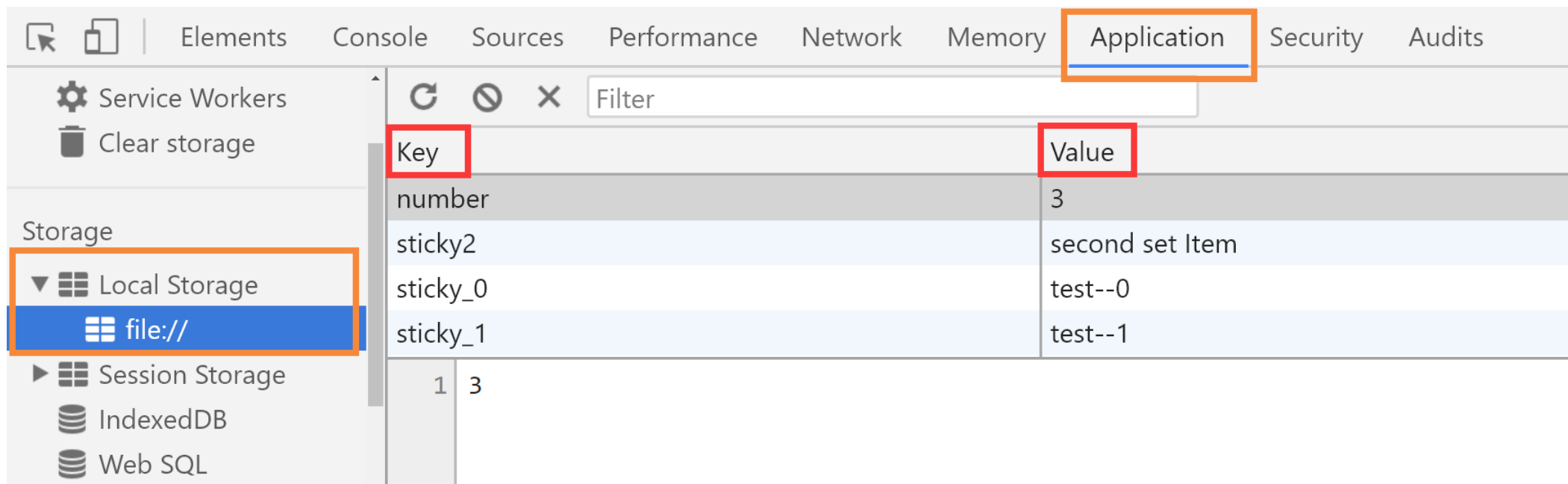


“键/值” 对



Web Storage用法

通过浏览器工具，查看数据存储情况



The screenshot shows the Chrome DevTools interface with the 'Application' tab selected. In the left sidebar, the 'Storage' section is expanded, and 'Local Storage' is selected, showing the file:// path. The main panel displays a table of stored data with 'Key' and 'Value' headers. The data includes a 'number' key with value '3', a 'sticky2' key with value 'second set Item', and two 'sticky' keys with values 'test--0' and 'test--1'.

Key	Value
number	3
sticky2	second set Item
sticky_0	test--0
sticky_1	test--1

Web Storage API

🎨 浏览器提供 **localStorage 对象**，其中包含一些操作数据的方法

➤ 检测浏览器是否支持

```
if( window.localStorage ) { ... }
```

➤ 存储数据项

```
localStorage.setItem(key, value)
```

```
localStorage[ 'key' ] = value
```

➤ 读取数据项

```
localStorage.getItem(key)
```

```
localStorage[ 'key' ]
```

demo6-1.html

Web Storage API

🎨 浏览器提供 **localStorage 对象**，其中包含一些操作数据的方法

➤ 删除数据项

`localStorage.removeItem(key)`

➤ 清除全部数据

`localStorage.clear()`

➤ 获得键名

`localStorage.key(index)`

➤ 获得数据项的个数

`localStorage.length`

demo6-2.html

demo6-3.html

Web Storage 用法

❖ 存储数据的数据类型

- 仅能存储字符串类型的数据

❖ 对象如何存储？

- 转换为字符串 —— JSON API
- `JSON.stringify()`
- `JSON.parse()`

```
{  
  "sticky1" : " test1" ,  
  "sticky2" : "test2" ,  
  ...  
}
```

练习

数据操作

添加指定数据：

key: value:

读取指定数据：

key: value:

删除指定数据：

key:

删除所有数据：

Elements Console Sources Application >>

Storage

▼ Local Storage

file://

▶ Session Storage

IndexedDB

Web SQL

▶ Cookies

↻ ⛔ ✕ Filter

Key	Value
-----	-------

03

SessionStorage使用方法

SessionStorage用法

LocalStorage

- 对象名称：localStorage
- 数据有效期：总是有效

SessionStorage

- 对象名称：sessionStorage
- 数据有效期：会话周期（浏览器关闭则失效）

SessionStorage用法

- ❖ 用户浏览这个网站所花费的时间就是 session 的生命周期。
- ❖ session 对象可以用来保存在这段时间内所要求保存的任何数据。
它的作用域是窗口级别的。
- ❖ sessionStorage方法针对一个 session 进行数据存储。当用户关闭浏览器窗口后，数据会被删除。

SessionStorage用法

🎨 检测浏览器是否支持

➤ `if(window.sessionStorage) { ... }`

🎨 session 对象方法：

- 保存数据：`sessionStorage.setItem(Key, value)`
- 读取数据：`sessionStorage.getItem(Key)`
- 移除数据：`sessionStorage.removeItem(key)`
- 清除所有数据：`sessionStorage.clear()`

设计计数器

- 🎨 设计计数器，思考使用 localStorage 和 sessionStorage 之间的区别？
- 🎨 sessionStorage 可以作为会话计数器，localStorage 则可以作为 Web 应用访问计数器。

04

Web Storage实例

综合实例--简单的Web留言本

- ❖ 使用 HTML5 存储技术实现一个简单的 Web 留言本。
- ❖ 使用文本域输入数据，单击“添加”按钮时，会将数据保存到 localStorage 中，在表单下面放置一个 P 元素来显示保存后的数据。

简单Web留言本

HTML5 Web Storage

添加

全部清除

明日科技	Mon, 05 Dec 2016 09:06:37 GMT
HTML5 Web Storage	Mon, 05 Dec 2016 09:08:22 GMT

综合实例--简单的Web留言本

🎨 第一步：编写显示页面用的HTML代码部分。

➤ 点击“添加按钮”来保存数据，点击“全部清除”按钮来消除全部数据。

```
<h1>简单Web留言本</h1>
<textarea id="memo" cols="60" rows="10"></textarea><br>
<input type="button" value="添加" onclick="saveStorage('memo'); "/>
<input type="button" value="全部清除" onclick="clearStorage('msg'); "/>
<hr/>
<p id="msg"></p>
```

综合实例--简单的Web留言本

❖ 第二步：编写点击“添加”按钮时调用的 saveStorage 函数。

- 在这个函数中使用 “new Date().getTime()” 语句得到了当前的日期和时间戳，然后调用 localStorage.setItem 方法，将得到的时间戳作为键值，并将文本框中的数据作为键名进行保存。
- 保存后，调用脚本中的 loadStorage 函数在页面上重新显示保存后的数据。

综合实例--简单的Web留言本

第三步：获取保存数据。

- 使用 `loadStorage.length` 属性获取保存数据的条数，循环，从 0 开始将该变量作为 `index` 参数传入 `loadStorage.key (index)` 属性，每次循环时该变量加 1，以此取得保存在 `loadStorage` 中的所有数据。

第四步：编写“全部清除”按钮。

- 调用 `clearStorage` 函数 将所有保存在 `localStorage` 中的数据全部清除。

练习

页面换肤



LocalStorage将数据保存在客户端本地的硬件设备中，即使浏览器被关闭了，该数据仍然存在，下次打开浏览器访问网站时仍然可以继续使用。

THANKYOU

