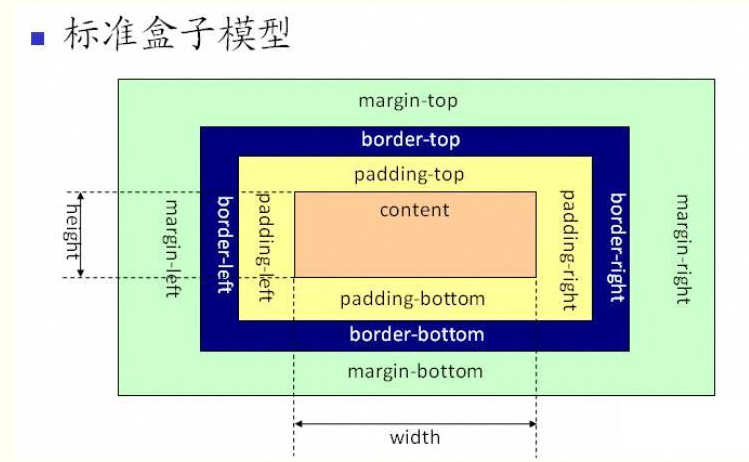
## 1盒模型

标准盒模型 ：content不包含padding，margin。

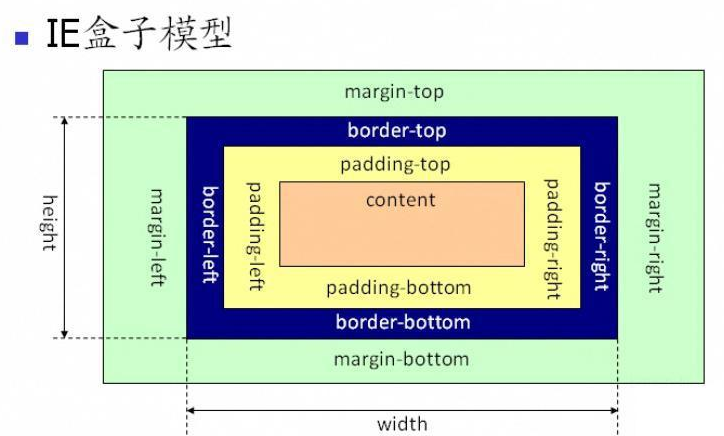
ie盒子模型：content 包含padding和border。

**因此计算盒子宽度**

标准 =（padding+margin）\*2+border+width



IE = margin\*2+width



## 2关于websocket

websocket一种基于tcp，全双工，一次握手，持久连接的协议

老的替代物有

Ajax long-polling技术（ajax长轮询）

flash技术。

在支持WebSocket的浏览器中，在创建socket之后。可以通过onopen，onmessage，onclose即onerror四个事件实现对socket进行响应

一个简单示例

var ws = new WebSocket(“ws://localhost:8080”);

ws.onopen = function()

{

  console.log(“open”);

  ws.send(“hello”);

};

## 3关于webworker

实例化 Worker 对象并传入要执行的 Javascript 文件名就可以创建一个新的 Web Worker：

1. var worker = new Worker('worker.js');

这段代码会导致浏览器下载 worker.js，但只有 Worker 接收到消息才会实际执行文件中的代码。要给 Worker 传递消息，可以使用 postMessage() 方法

主程序js和webworkerjs通信都是用postmessage（）;

* Webworker是单独作用域的，无法访问到window 对象
* document 对象
* parent 对象

## 4 JS基本数据类型和引用数据类型

基本数据类型：Number，String，Boolean，Null，Undefined

引用数据类型: array，function，object

## 5 html5语义化理解

方便人和虫的阅读

## 6 缓存

## Manifest 文件

manifest 文件是简单的文本文件，它告知浏览器被缓存的内容（以及不缓存的内容）。

manifest 文件可分为三个部分：

·        ***CACHE MANIFEST*** - 在此标题下列出的文件将在首次下载后进行缓存

·        ***NETWORK*** - 在此标题下列出的文件需要与服务器的连接，且不会被缓存

·        ***FALLBACK*** - 在此标题下列出的文件规定当页面无法访问时的回退页面（比如 404 页面）

Cache manifest：这下面部分会缓存

network：不会缓存

fallback：无法访问时回退页面

## 7cookie与webstorage

cookie缺陷：

1. 数据大小只有4kb左右
2. 安全性问题，在http请求中是明文传递的
3. 每次http请求都会带上
4. 不同浏览器还有数量限制

Webstorage

不是替代cookie，cookie是为了客户端与服务端通信，

有时候会被用来客户端本地存储，。webstorage为了解决这个问题

webstorage分，session storage和localstorage

一个是永久储存，sessionstorage是会话期间有效。

    sessionStorage：不在不同的浏览器窗口中共享，即使是同一个页面；

     localStorage：在所有同源窗口都是共享的；

     cookie：也是在所有同源窗口中共享的

拓展：webStorage只能操作**字符串对象**

## 8多页面通信

cookie，web worker，webstorage和windows.name等

## 9 键入url到页面渲染的过程。

1输入地址

2浏览器查找域名ip地址，（浏览器缓存，系统缓存，路由器缓存）

3浏览器向web服务器发送一个http请求。

4服务器重定向

5浏览器跟踪重定向

6服务器处理请求

7服务器返回http请求

8解析html文档，生成树

9渲染css，请求嵌入在html的资源

10浏览器发送异步请求

## 10重构、回流

浏览器的重构指的是改变每个元素外观时所触发的浏览器行为，比如颜色，背景等样式发生了改变而进行的重新构造新外观的过程。重构不会引发页面的重新布局，不一定伴随着回流，

回流指的是浏览器为了重新渲染页面的需要而进行的重新计算元素的几何大小和位置的，他的开销是非常大的，回流可以理解为渲染树需要重新进行计算，一般最好触发元素的重构，避免元素的回流；比如通过通过添加类来添加css样式，而不是直接在DOM上设置，当需要操作某一块元素时候，最好使其脱离文档流，这样就不会引起回流了，比如设置position：absolute或者fixed，或者display：none，等操作结束后在显示

## 11 JavaScript的闭包

闭包简单的说就是一个函数能访问外部函数的变量

## 12new操作符干了什么？

新建一个空对象，把this指向该对象

空对象的原型指向函数原型，

最后改变构造函数内部this指向

## 13call和apply和bind

改变函数内部this指向可以fn.call(obj);或者fn.apply(obj)

,然后立即执行，第一个参数都是对象，第二个apply为数组，

call为arg1，arg2，arg3这种。

另外bind只有一个参数，就是对象。

返回一个函数，

exp： var bar=fn.bind（obj）;