# BOM

## 课程纲要

ECMAScript复习：

阶段一：BOM简介

阶段二：Window对象

## BOM 简介

### JavaScript的组成部分

ECMAScript 标准语法

BOM 浏览器对象模型

DOM 文档对象模型

### 什么是BOM ?

BOM是浏览器对象模型，是Bowser Object Model 的简称，哪浏览器对象模型又是什么概念？

浏览器：我们访问网站，必须通过浏览器这个平台来进行访问，我们之前所写的html、css以及一些js代码也必须通过浏览器来进行解析和运行。所以浏览器就是一个平台，类似一个我们的PC或手机的操作系统一样。

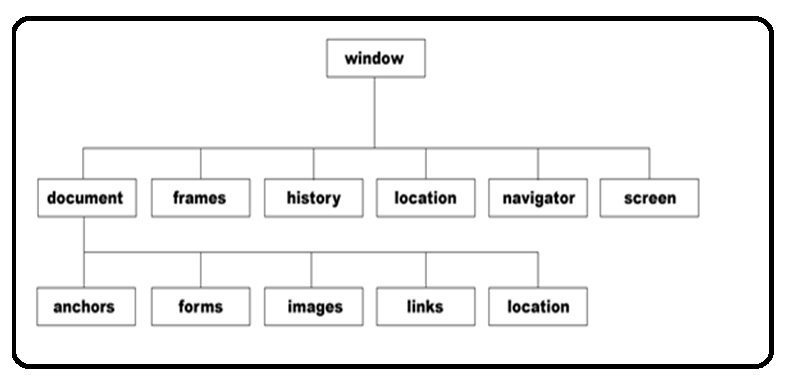
对象： 什么是对象？我们之前所接触的Date就是一个对象，Date对象通过提供给我们一些方法（函数、功能），可以方便的帮助我们操作时间。我们之前讲过，方法（函数、功能）就是一个工具，对象里有很多工具，所以对象我们可以理解为一个小的工具库！

模型：Model，有模式，模样的意思。

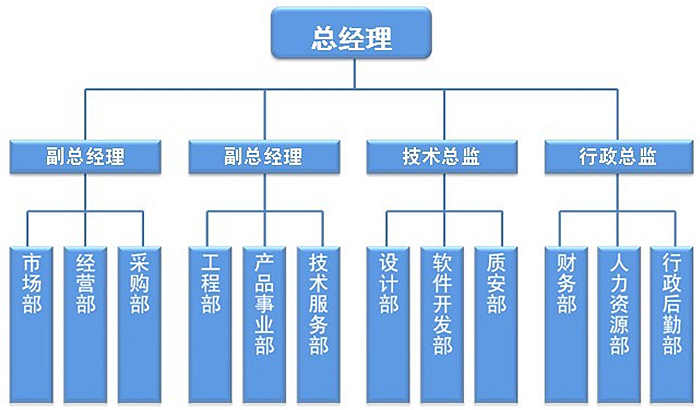
所以，通过以上的分析，BOM是什么呢？

就是 浏览器 工具库 说明书！

我们学习BOM就是学习 js 针对操作浏览器所提供的一些 工具



这种组织结构关系，可以理解为



## Window(类型)对象

在讲window对象之前，我们先看一下浏览器都有哪些功能?

Window 对象

--通过小写的 window 变量引用

--扮演者两个角色

--javascript访问浏览器的窗口的一个接口（入口）

--ECMAScript中的Global对象，全局对象。

什么是全局对象，全局对象就是顶层对象，顶层对象可以理解为CEO。其他的一切全局变量，都是window对象的下属。

注意对于全局变量，我们可以省略window

### Document(类型)对象

通过 document 变量引用，我们所有的网页内容都是通过这个Document对象来进行管理的。

比如我们之前学过的document.write();

后面的DOM课程，就是来学习Document这个对象所提供的一些工具。

### ~~frames~~

框架集，类似一个数组，可以通过frames[索引号]（frames[0]）或frames[框架名称]，来获取指定的框架。框架标签，工作当中基本上不用，所以，对于没用的东西，我们没必要浪费时间去学它。

### History(类型)对象

通过 history 变量可以引用，对于在当前窗口中，我们所访问的所有页面，都会被记录在History对象中。通过这个对象，我们可以很容易的跳转到我们之前所浏览的页面中

history.go(参数);

~~参数可以是整数数字也可以是字符串；~~

整数数字，可以是正数和负数，正数向前跳转，负数向后跳转；

~~字符串：可以使所浏览的url中包含的字符串，页面会跳转到距当前页面最近的历史记录页面中；~~

可以用另外两个方法来替 代go方法

history.back();

history.forward();

history.length; 可以获取浏览的历史记录的数量

### Location对象

通过变量 location 引用，location对象可以帮助我们操作url并提供了一下页面导航功能，关于url的操作呢，我们后面再讲。



https://www.baidu.com/s?ie=utf-8&f=8&rsv\_bp=0&rsv\_idx=1&tn=baidu&wd=location.tosource()&rsv\_pq=94ea40a40012ee13&rsv\_t=5bc3DyXXFKrQxqRFmyPzc6IXwfsOj6K%2FCWSb3Y3mVsyW1ncLTV77iqP5iTI&rsv\_enter=1&rsv\_sug3=20&rsv\_sug1=27&rsv\_sug7=100

* + href属性 控制浏览器地址栏的内容
  + reload()方法 刷新页面，有可能会使用缓存（暂存！！！！）
  + reload(true) 刷新页面，不使用缓存（绝对的！）,重新请求服务器。

### Navigator对象

通过navigator对象引用，Navigator对象可以帮助我们检测浏览器的信息以及浏览器所在的操作系统的信息。

* + appName 浏览器名称
  + appVersion 浏览器版本
  + platform操作系统
  + 最新的浏览器已经全面放弃以上这些属性
  + userAgent 用户代理信息，通过该属性可以获取浏览器及操作系统信息

### Screen对象

通过screen对象来引用，可以用它来获取当前屏幕的高度和宽度。

width:获取屏幕的宽度;

height;

h5:视口（view）

### setTimeout(函数，时间（毫秒数）)

超时调用，就是在指定过了多少时间后，执行我们的代码，只执行一次。

清除定时器

clearTimeout(id);

### setInterVal(函数，时间（毫秒数）));

间歇调用，每隔指定的时间，调用一次我们代码，重复调用。

清除定时器

clearInterval();

### close() //关闭当前页面

ie\chrome\[Safari](http://www.baidu.com/link?url=pHnrK5YbSxzvNrdJvpYzakXfcbBT2H4EexW07MnKqYd834IYne8UEW_TvObwz-UklnlcI-Nwc1x-UsdkYw9n7K" \t "_blank)均支持，火狐支持，不过它的浏览器配置参数默认禁用了这个方法，所以我们需要通过修改它的配置参数。

在浏览器地址中输入： about:config 找到

设置它的参数值为true

### alert(text)

弹出对话框

### confirm(text)

弹出对话框，但会有返回值，点击确定是true,点击取消返回false

### prompt(text,defaultText)

弹出对话框，显示提示信息，并提供用户输入的一个文本框，并且可以预先设定默认的输入内容

\*\*\*\*常见事件：

onload

页面加载完后，运行该事件所驱动的函数的代码

作业：随机点名（有能力的做一下）

作业：倒计时！（2016-5-18） // 剩余

剩余：2小时59分59秒

作业：用setTimeout模拟setInterVal 数数！