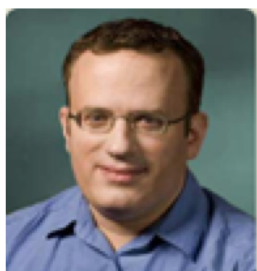


上周考试
新工具 HBuilder

1.1 JavaScript是什么?

- 1. JavaScript主要用于HTML的页面，嵌入在HTML的源码中。
- 2. JavaScript是因特网上最流行的脚本语言，它存在于全世界所有 Web 浏览器中，能够增强用户与 Web 站点和 Web 应用程序之间的交互。
- 3. JS是弱类型语言,没有类型声明,它的变量不必具有一个明确的类型。
- 5. JS也是解释性的语言。何为解释性语言?是在运行的时候将程序直接翻译成机器的语言
- 4. JS是脚本语言，换句话说，可以用来编程的并且直接执行源代码的语言,就是脚本语言。
- 6. JavaScript是一种基于对象(Object)和事件驱动(Event Driven)并具有安全性能的脚本语言，可广泛用于服务器、PC、笔记本电脑、平板电脑和智能手机等设备。
- 7. HTML5的出现更是突出了JavaScript的重要性，例如HTML5的绘图支持、本地存储、离线应用、客户端通信等，都大量使用了JavaScript。

网景公司在上个世纪的1995年，凭借其Navigator浏览器，成为Web时代开启时最著名的第一代互联网公司。由于网景公司希望能在静态HTML页面上添加一些动态效果，于是叫Brendan Eich这哥们在设计出了JavaScript语言。



为什么起名叫JavaScript? 原因是当时Java语言非常红火，所以网景公司希望借Java的名气来推广，但事实上JavaScript除了语法上有点像Java，其他部分基本上没啥关系。

1.2 JavaScript历史

JavaScript因为互联网而生，紧随着浏览器的出现而问世。回顾它的历史，就要从浏览器的历史讲起。

- 1990年底，欧洲核能研究组织（CERN）科学家Tim Berners-Lee（蒂姆·伯纳斯-李），互联网之父
- 1992年底，美国国家超级电脑应用中心（NCSA）开始开发一个独立的浏览器，叫做Mosaic（马赛克）。
- 1994年10月，NCSA的一个主要程序员MarcAndreessen（马克·爱德森）联合风险投资家Jim Clark（吉姆·克拉克），成立了Mosaic通信公司（Mosaic Communications），不久后改名为网景Netscape。这家公司的方向，就是在Mosaic的基础上，开发面向普通用户的新一代的浏览器 Netscape Navigator。
- 1994年12月，Navigator发布了1.0版，市场份额一举超过90%。
- 1995年，Netscape公司Brendan Eich开发这种网页脚本语言。1995年5月，Brendan Eich只用了10天，就设计完成了这种语言的第一版liveScript。它是一个大杂烩，语法有多个来源：
 - 基本语法：借鉴C语言和Java语言。
 - 数据结构：借鉴Java语言，包括将值分成原始值和对象两大类。
 - 函数的用法：借鉴Scheme语言和Awk语言，将函数当作第一等公民，并引入闭包。
 - 原型继承模型：借鉴Self语言（Smalltalk的一种变种）。
 - 字符串和数组处理：借鉴Python语言。
 - 正则表达式：借鉴Perl语言。
- 1995年12月4日，Netscape公司与Sun公司联合发布了JavaScript语言。
- 1996年3月，Navigator 2.0浏览器正式内置了JavaScript脚本语言。
- 1996年8月，微软模仿JavaScript开发了一种相近的语言，取名为JScript（JavaScript是Netscape的注册商标，微软不能用），首先内置于IE 3.0，Netscape公司面临丧失浏览器脚本语言的主导权的局面。
- 1997年 IE4与nn4平分天下。网景公司将JavaScript交给ECMA(European Computer Manufacturers Association)组织，"以此"来抵制微软的垄断。
- 1998年 ECMAScript 2.0
- 1999年 ECMAScript 3.0
- 2008年 ECMAScript4.0应为升级太大，废弃
- 2009年 ECMAScript5.0发布
- 2011年ECMAScript 5.1版发布，并且成为ISO国际标准（ISO/IEC 16262:2011）。到了2012年底，所有主要浏览器都支持ECMAScript 5.1版的全部功能。

2015年 6月 ECMAScript6.0 改名为 ECMAScript2015

为什么改名了呢?

因为这个版本引入的语法功能太多，而且制定过程当中，还有很多组织和个人不断提交新功能，不可能在一个版本里面包括所有将要引入的功能

任何人在任何时候，都可以向标准委员会提交新语法的提案，然后标准委员会每个月开一次会，评估这些提案是否可以接受，需要哪些改进。如果经过多次会议以后，一个提案足够成熟了，就可以正式进入标准了

标准在每年的6月份正式发布一次，作为当年的正式版本。接下来的时间，就在这个版本的基础上做改动，直到下一年的6月份，草案就自然变成了新一年的版本。这样一来，就不需要以前的版本号了，只要用年份标记就可以了。

ES6的第一个版本，在2015年6月发布了，正式名称就是《ECMAScript 2015标准》（简称ES2015）。2016年6月，小幅修订的《ECMAScript 2016标准》（简称ES2016）如期发布，这个版本可以看作是ES6.1版，因为两者的差异非常小（只新增了数组实例的includes方法和指数运算符），基本上是一个标准。根据计划，2017年6月将发布ES2017标准。

ES6既是一个历史名词，也是一个泛指，含义是5.1版以后的JavaScript的下一代标准，涵盖了ES2015、ES2016、ES2017等等，而ES2015则是正式名称，特指该年发布的正式版本的语言标准。本书中提到“ES6”的地方，一般是指ES2015标准，但有时也是泛指“下一代JavaScript语言”。

JS 是什么?

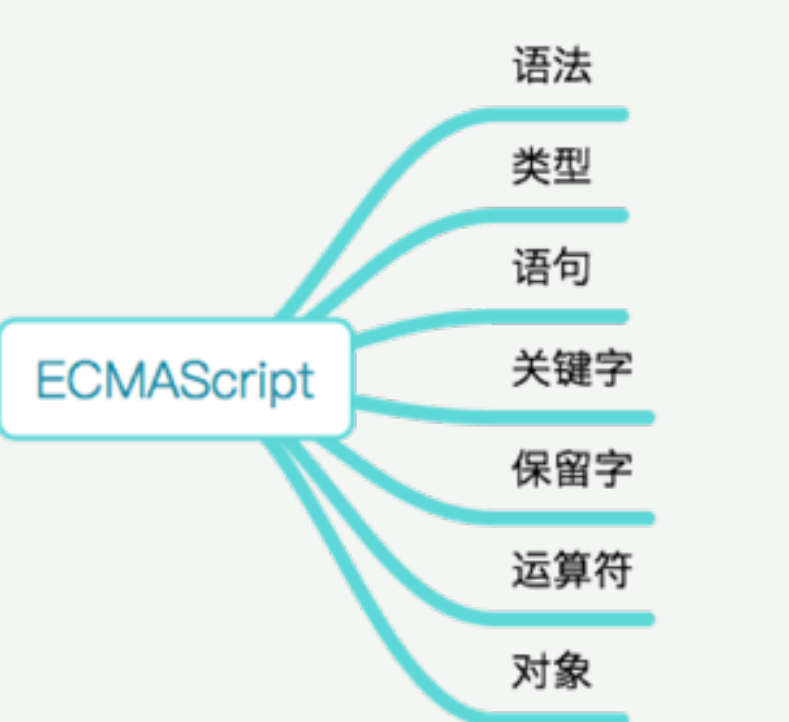
1.3 JavaScript主要特点

- 1) 简单性：它是基于Java基本语句和控制流之上的简单而紧凑的设计，但是相比Java来说，它的变量类型是采用弱类型，未采用严格的数据类型。
 - 比C语言好学
- 2) 安全性：JS不允许访问本地硬盘，不能将数据存入到服务器上，不允许对网络文档进行修改和删除，只能通过浏览器实现信息浏览或动态交互，从而有效的防止数据的丢失。
 - 不能将数据存入服务器
 - 不允许访问硬盘
 - 不允许对网络文档进行操作
 - 只能通过浏览器实现信息浏览或动态交互
- 3) 动态性：JS可以直接对用户或客户输入做出响应，无须经过Web程序。它对用户的响应采用以事件驱动的方式进行，即由某种操作动作引起相应的事件响应，如：点击鼠标、移动窗口、选择菜单等。
 - 直接对用户输入做出相应，无须经过web程序
 - 它对用户的响应采用事件驱动的方式进行
 - 点击鼠标
 - 移动窗口
 - 选择菜单
- 4) 跨平台性：JS依赖于浏览器本身，与操作环境无关。只要能运行浏览器的计算机，并安装了支持JS的浏览器就可以正确执行，从而实现了“编写一次，走遍天下”的梦想。
 - 依赖于浏览器
 - 与操作系统无关
 - 编写一次，随处运行

ECMAScript是一个标准。因为网景开发了JavaScript，为了让JavaScript成为全球标准，几个公司联合ECMA（European Computer Manufacturers Association）组织定制了JavaScript语言的标准，被称为ECMAScript标准。

描述了改语言的语法和基本对象。

ECMAScript（核心）



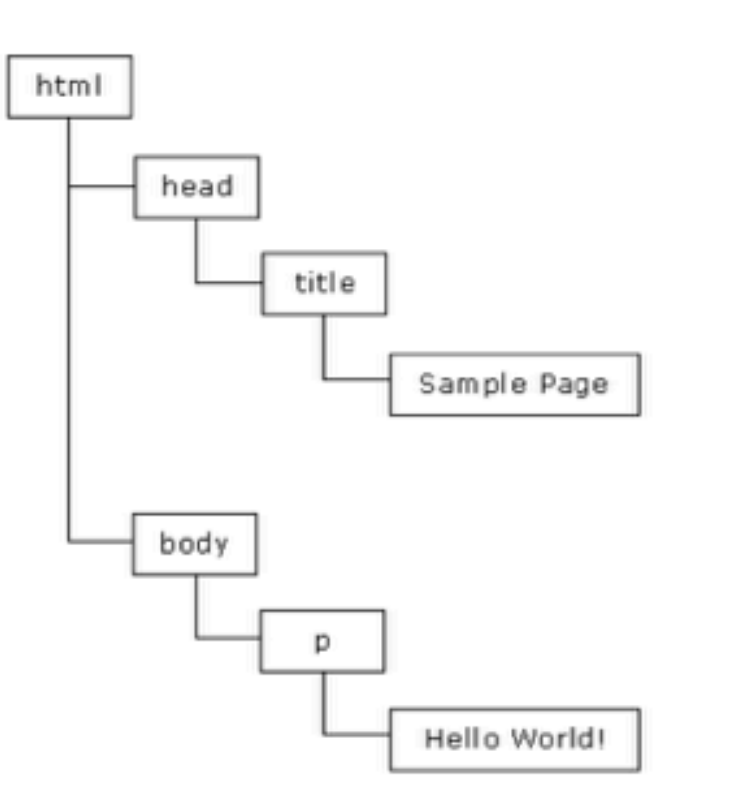
1.4 JavaScript理论体系

DOM（文档对象模型）

DOM(document object model 文档对象模型)。通过 DOM，可以访问所有的 HTML 元素。连同它们所包含的文本和属性，可以对其中的内容进行修改和删除，同时也可以创建新的元素。

DOM Document Object Model

```
<html>
<head>
  <title>Sample Page</title>
</head>
<body>
  <p>hello world!</p>
</body>
</html>
```



BOM（浏览器对象模型）

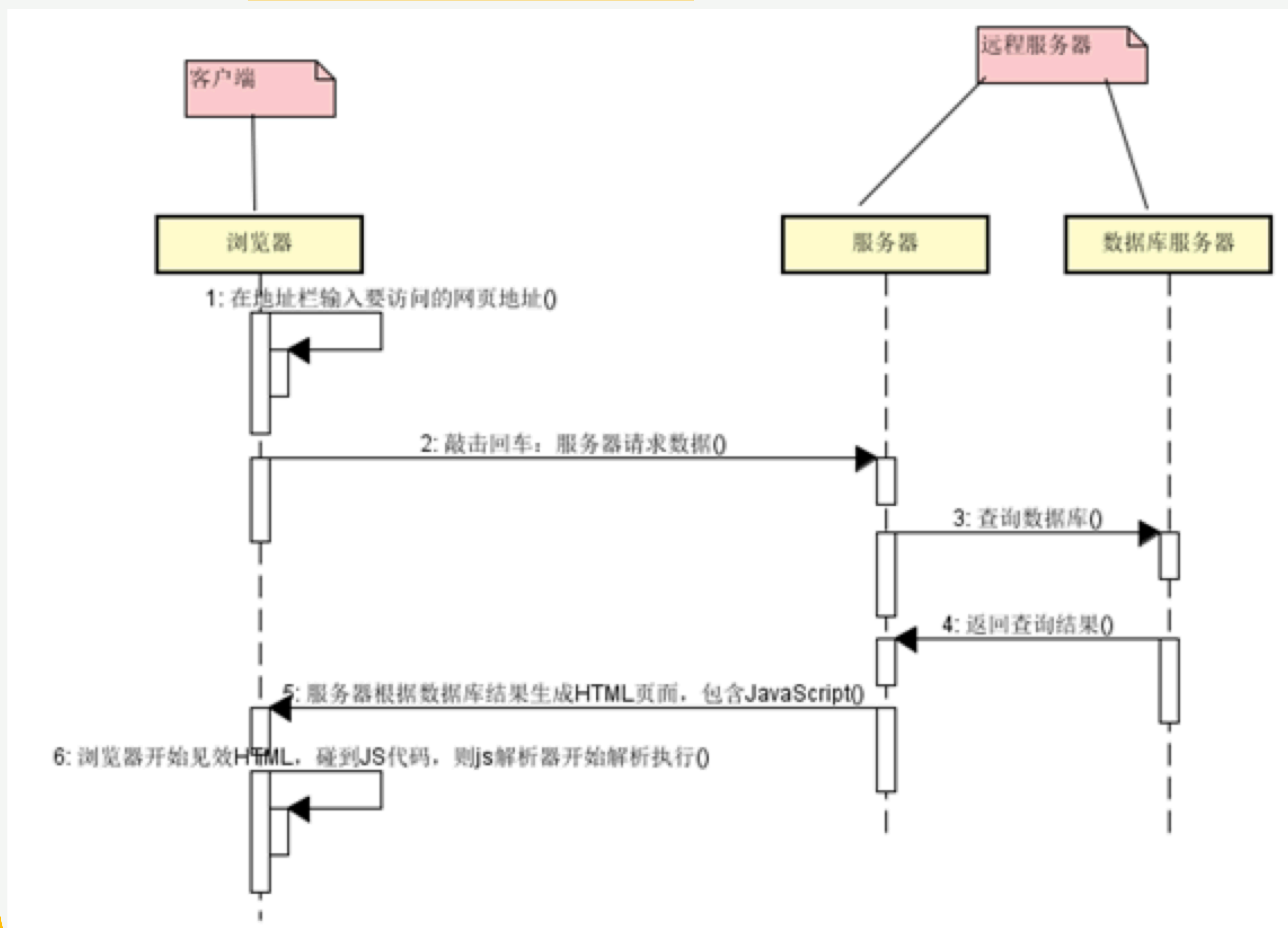
BOM(browser object model 浏览器对象模型)。“描述了与浏览器进行交互的方法和接口。BOM提供了独立于内容而与浏览器窗口进行交互的对象，例如可以移动，调整浏览器大小的window对象，可以用于导航的location对象与history对象，可以获取浏览器的，操作系统与用户屏幕信息的navigator与screen对象，可以使用document作为访问HTML文档的入口，管理框架的frames对象等。

BOM Browser Object Model

- 弹出新的浏览器窗口
- 移动、关闭、调整浏览器窗口
- Web浏览器详细信息的定位对象
- 用户屏幕分辨率详细信息的屏幕对象
- 对cookies的支持
- 像XMLHttpRequest和IE的scriptableObject这样的自定义对象

由于没有 BOM 标准可以遵循,因此每个浏览器都有自己的实现

1.5 JavaScript工作原理



如何去引入 JS

- 内部引用 JS 推荐书写位置:body的下面
- 外部链接引用 JS 中间不能书写内容
- 标签内部使用 JS script 双标签

注释

- 单行注释 //
- 多行注释 /* */

书写的时候的注意事项

JS 中提示框的三种类型

- 警告框 alert
- 确认框 confirm
- 提示框 prompt

JS 的调试常用方法

- alert
- console
 - log
 - info
 - warn
 - error
- document.write
- document.title

JS 的书写语法规则

- JS 执行顺序 从上至下, 从左至右
- JS 中严格区分大小写的 JS 对大小写敏感
- JS 中忽略空白符和换行符
- JS 中可以通过 \ 来实现代码的折行
- JS 中使用 ; 作为语句的结尾, 可以省略, 但是不推荐省略

JS 的变量

变量的声明规范

标识符

- 首字母必须是字母、下划线、或者美元符 (\$) 不能使用中文
- 其他字符可以是数字、字母、下划线、美元符
- JS 变量的命名要严格区分大小写
- JS 的命名不能使用关键字和保留字
- 习惯
 - 驼峰命名
 - 下滑线

关键字

break	do	instanceof	typeof
case	else	new	var
catch	finally	return	void
continue	for	switch	while
delete	function	this	with
default	if	throw	
delete	in	try	

保留字

abstract	enum	int	short
boolean	export	interface	static
byte	extends	let	super
char	final	new	synchronized
class	float	package	throw
const	goto	private	transient
delete	implements	protected	volatile
double	import	public	

第 5 版在非严格模式下运行的保留字缩减为下列这些:

class	enum	extends	super
const	export	import	

在严格模式下, 第 5 版还对以下保留字增加了限制:

implements	package	private	static
interface	protected	public	