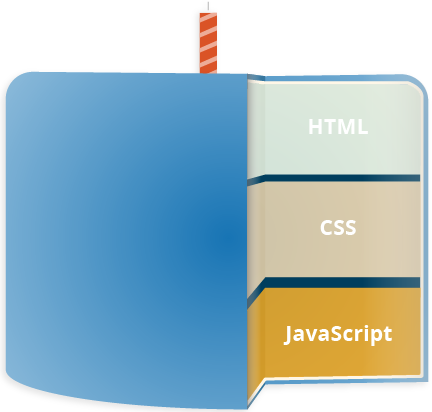
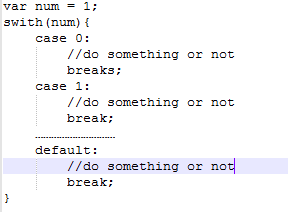
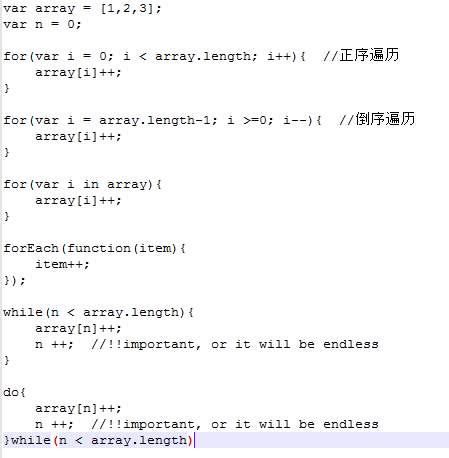
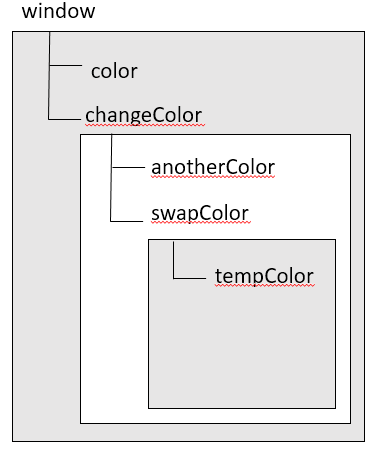
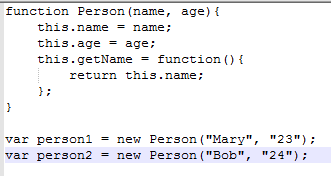
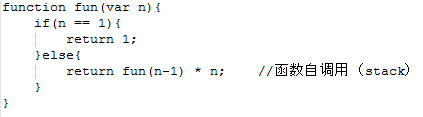
* 1995年诞生，Netscape开发并命名为LiveScript。后Netscape与Sun合作，改名为JavaScript。
* 1996年，微软在IE3中加入了名为JScript的JavaScript实现。
* 1997，欧洲计算机制造商协会（ECMA，European Computer Manufacturers Association）指定来自Netscape，Sun，微软及其他关注脚本语言发展的公司的程序员组成39号技术委员会（TC39，Techical Committee #39），最后完成了ECMA-262标准——用来定义名为ECMAScript（发音：“ek-ma-script”）的新脚本语言。
* 1998年，国际标准化组织和国际电工委员会（ISO/IEC，International Organization For Standardization andInternational Electrotechnical Committee）采用了ECMAScript作为标准。此后各浏览器开发商基于ECMAScript开发各自的JavaScript实现



* [HTML](https://developer.mozilla.org/en-US/docs/Glossary/HTML)是一种标记语言，用来结构化我们的网页内容和赋予内容含义，例如定义段落、标题、和数据表,或在页面中嵌入图片和视频。
* [CSS](https://developer.mozilla.org/en-US/docs/Glossary/CSS)是一种样式规则语言，我们将样式应用于我们的 HTML 内容， 例如设置背景颜色和字体，在多个列种布局我们的内容。
* [JavaScript](https://developer.mozilla.org/en-US/docs/Glossary/JavaScript)是一种编程语言，允许你创建动态更新的内容，控制多媒体，图像动画，和一些其他的东西。

1. **学习资料及方法**
   1. <http://www.w3school.com.cn/index.html>
   2. <https://developer.mozilla.org/zh-CN/docs/Web/JavaScript>
2. **变量**
   1. var a = 1, b;  //声明a,b 及初始化   
      b = 2;  //给b赋值
   2. null VS undefined<http://www.ruanyifeng.com/blog/2014/03/undefined-vs-null.html>
      1. null表示没有对象，即此处不该有值，用法：
         * 作为函数的参数， 表示该函数的参数不是对象；
         * 作为对象原型链的终点
      2. undefined表示没有初始值，即此处应该有值，但是还没有定义，用法：
         * 变量被声明了，但是还没有赋值；
         * 调用函数的时候，应该提供的参数没有提供，则该参数被赋值为undefined；
         * 对象的属性在被赋值之前默认值为undefined；
         * 函数的默认返回值为undefined
      3. 在计算时null转换为0，undefined转换为NaN（Not a Number）：  
         var  a = null, b = undefined;  
         Number(null); //0  
         a + 5; // 5  
         Number(undefined); //NaN  
         b + 5; //NaN
   3. [], {
   4. “name”: "jason"
   5. }
   6. 常量：不变化的量（如pi=3.1415926，pi是变量-用来存储数值的容器，3.1415926是常量）  
                  true/false, null……
3. **运算符 （从高到底）**

|  |  |  |  |
| --- | --- | --- | --- |
| 优先级 | 运算类型 | 关联性 | 运算符 |
| 20 | [圆括号](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Grouping) | n/a | ( … ) |
| 19 | [成员访问](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Property_Accessors#点符号表示法) | 从左到右 | … . … |
|  | [需计算的成员访问](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Property_Accessors#括号表示法) | 从左到右 | … [ … ] |
|  | [new](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/new) (带参数列表) | n/a | new … ( … ) |
|  | [函数调用](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Guide/Functions) | 从左到右 | … ( … ) |
| 18 | [new](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/new) (无参数列表) | 从右到左 | new … |
| 17 | [后置递增](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Increment)(运算符在后) | n/a | … ++ |
|  | [后置递减](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Decrement)(运算符在后) | n/a | … -- |
| 16 | [逻辑非](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Logical_Operators#Logical_NOT) | 从右到左 | ! … |
|  | [按位非](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators#Bitwise_NOT) | 从右到左 | ~ … |
|  | [一元加法](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Unary_plus) | 从右到左 | + … |
|  | [一元减法](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Unary_negation) | 从右到左 | - … |
|  | [前置递增](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Increment) | 从右到左 | ++ … |
|  | [前置递减](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Decrement) | 从右到左 | -- … |
|  | [typeof](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/typeof) | 从右到左 | typeof … |
|  | [void](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/void) | 从右到左 | void … |
|  | [delete](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/delete) | 从右到左 | delete … |
| 15 | [幂](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Exponentiation) | 从右到左 | … \*\* … |
| 14 | [乘法](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Multiplication) | 从左到右 | … \* … |
|  | [除法](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Division) | 从左到右 | … / … |
|  | [取模](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Remainder) | 从左到右 | … % … |
| 13 | [加法](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Addition) | 从左到右 | … + … |
|  | [减法](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Arithmetic_Operators#Subtraction) | 从左到右 | … - … |
| 12 | [按位左移](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators) | 从左到右 | … << … |
|  | [按位右移](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators) | 从左到右 | … >> … |
|  | [无符号右移](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators) | 从左到右 | … >>> … |
| 11 | [小于](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Less_than_operator) | 从左到右 | … < … |
|  | [小于等于](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Less_than__or_equal_operator) | 从左到右 | … <= … |
|  | [大于](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Greater_than_operator) | 从左到右 | … > … |
|  | [大于等于](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Greater_than_or_equal_operator) | 从左到右 | … >= … |
|  | [in](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/in) | 从左到右 | … in … |
|  | [instanceof](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/instanceof) | 从左到右 | … instanceof … |
| 10 | [等号](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Equality) | 从左到右 | … == … |
|  | [非等号](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Inequality) | 从左到右 | … != … |
|  | [全等号](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Identity) | 从左到右 | … === … |
|  | [非全等号](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comparison_Operators#Nonidentity) | 从左到右 | … !== … |
| 9 | [按位与](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators#Bitwise_AND) | 从左到右 | … & … |
| 8 | [按位异或](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators#Bitwise_XOR) | 从左到右 | … ^ … |
| 7 | [按位或](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Bitwise_Operators#Bitwise_OR) | 从左到右 | … | … |
| 6 | [逻辑与](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Logical_Operators#Logical_AND) | 从左到右 | … && … |
| 5 | [逻辑或](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Logical_Operators#Logical_OR) | 从左到右 | … || … |
| 4 | [条件运算符](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Conditional_Operator) | 从右到左 | … ? … : … |
| 3 | [赋值](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Assignment_Operators) | 从右到左 | … = … |
|  |  |  | … += … |
|  |  |  | … -= … |
|  |  |  | … \*= … |
|  |  |  | … /= … |
|  |  |  | … %= … |
|  |  |  | … <<= … |
|  |  |  | … >>= … |
|  |  |  | … >>>= … |
|  |  |  | … &= … |
|  |  |  | … ^= … |
|  |  |  | … |= … |
| 2 | [yield](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/yield) | 从右到左 | yield … |
|  | [yield\*](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/yield*) | 从右到左 | yield\* … |
| 1 | [展开运算符](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Spread_operator) | n/a | ... … |
| 0 | [逗号](https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Comma_Operator) | 从左到右 | … , … |

1. **条件语句**
   1. if(){}
   2. If()else{}
   3. If(){}else if{}else{}
   4. Switch （括号里必须是变量）  
      
2. **循环结构**
   1. 将一个数组里的所有元素都+1 （6种方法）  
      
   2. for，forEach - 遍历数组  
      for，while，do-while - 可自定义循环的间隔（i+=2, i+=3……）
   3. while - 先判断，后执行  
      do-while - 先执行循环体，后判断，常用于循环体中的代码至少必须执行一次的情况
3. **break/continue**
4. **函数：被重复执行的代码块**
   1. 关键字：function
   2. 函数在执行完return语句后立刻退出（即return后的语句都不会被执行）
   3. 有返回值：return a; / 无返回值：return;
   4. 参数
      1. 参数不能同名
      2. 传不传参，传参个数都没关系 - arguments对象，类数组访问（arguments[0]）
      3. 按值传递 - 将函数外的变量复制给函数内的一个局部变量  
         值传递 vs 引用传递
   5. 不能重载（overwrite），即函数名不能重复， 因为没有签名（参数的类型和数量）
   6. eval(string)函数可计算并执行string中的JavaScript代码  
      eval("x=10;y=20;document.write(x\*y)")   //输出200
5. **变量作用域**
   1. 执行环境，变量对象（保存环境中定义的所有变量和函数）
   2. 全局变量 - 最外围的执行环境中定义的变量
   3. 局部变量- 每个函数都有自己的执行环境中定义的变量
   4. 局部变量 > 全局变量  
        
      
6. **对象：无序属性的集合，其属性可以包含基本值、对象或者函数**
   1. new - 类的实例化
   2. 构造函数  
      
      1. 首字母大写 - 与普通函数区分
      2. 调用构造函数实际执行的步骤：
         * 创建一个新对象；
         * 把构造函数的作用域赋给新对象（this指向这个对象）；
         * 执行构造函数中的代码；
         * 返回这个新对象
7. **递归**  
   计算1\*2\*3……\*n:  
   
8. **注释**
   1. 单行注释： //
   2. 多汗注释： /\*……\*/
9. **处理错误/异常**  
   try{  
          //运行代码  
         alllert("Error occurs!");  //正确应该是：alert("Error occurs!");  
         throw "Error!";  //自定义错误  
   }catch(error){  
         //处理错误  
   }finally{  
         //无论有无异常都会执行  
   }