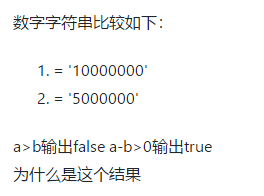
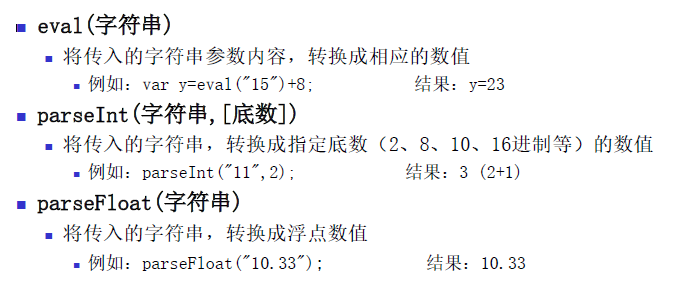
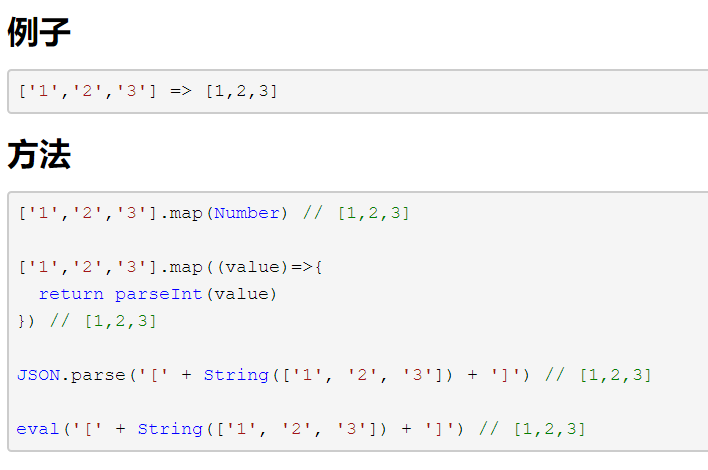
1. **数字字符串的比较**

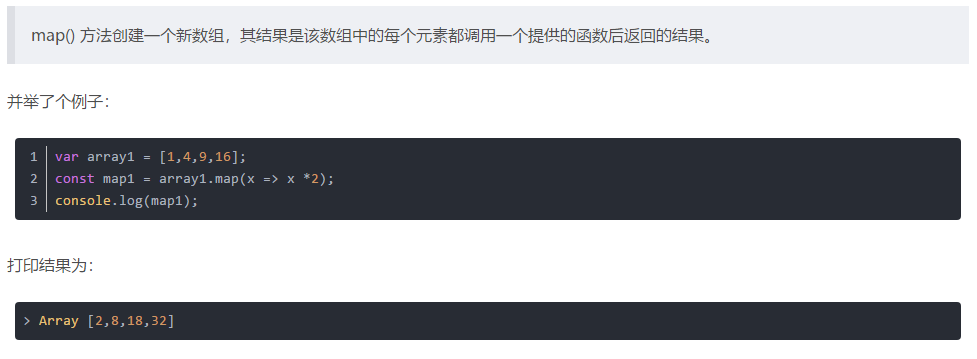


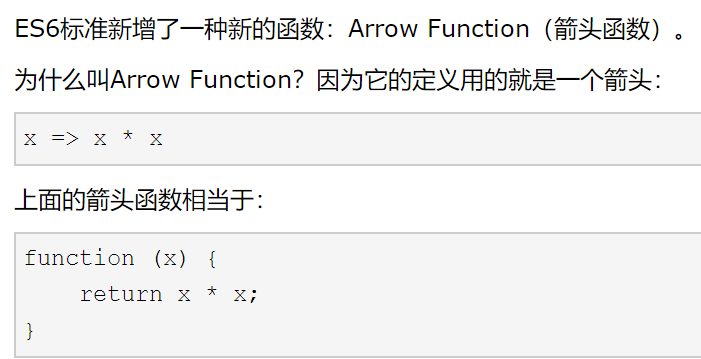
这是JavaScript中隐式转换的关系，首先在比较的时候，二者是按照string的形式进行比较的，那么比较第一位的时候，1 < 5,直接返回false,而 a - b会把两个数转成number类型进行比较，转换完之后输出就是true。可以仔细看看隐式转换这一块。



**字符串数字转化为数字数组：**

1. **map( ) 方法，“=>”箭头函数**





1. **Js数组常用方法**
2. **.push(“ddd”) 数组末尾添加字符串**
3. **.pop() 删除数组末尾的元素**
4. **.shift() 删除数组头部的元素**
5. **.indexOf(“ddd”) 找出某个元素在数组中的索引**
6. **.splice()**

**arrayObject.splice(index,howmany,item1,.....,itemX)**

**删除从索引处开始的零个或多个元素，并添加新元素。**

1. **.slice() 复制一个数组**

**arrayObject.slice(start,end)**

**注意：新数组包括start，不包括end**

1. **JS基础内容**
2. **数据类型：**

**原始类型：number，string，boolean，null，undefined.**

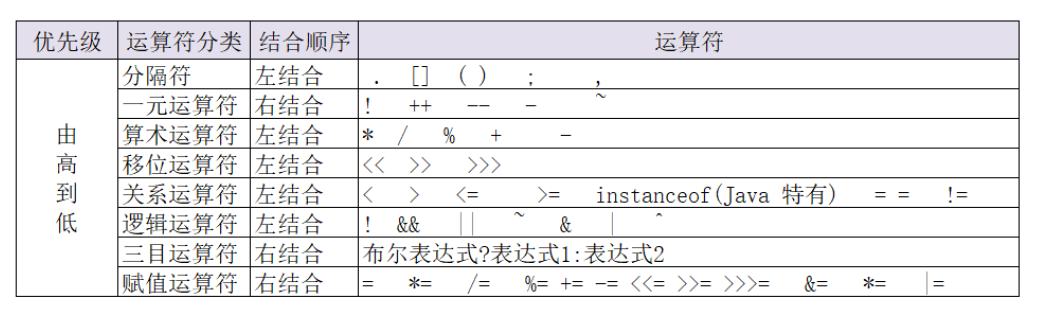
**对象类型：function，array，date，math**

**已经存在的数据vs自己创造的数据**

1. **数据类型转换：参照上节**
2. **特殊字符、转义字符：” \n ”、”\\”、” ” ”**
3. **运算符的优先级：**

**括号 > 自加自减非 > 加减乘除 > 移位 > 比较 >**

**逻辑 > ？布尔表达式 > 赋值**



1. **=== 与 == ，===即判断数值又判断类型**
2. **判断语句： if（条件）{}**

**else if条件）{}**

**else（条件）{}**

**多分支语句：switch（变量）{**

**case 值1：执行代码；break;**

**case 值2：执行代码；break;**

**case 值3：执行代码；break;**

**……**

**default: 执行代码；**

**}**

**for循环： for(初始值；循环条件；迭代){**

**执行代码；**

**}**

**while循环： while（表达式条件）{**

**执行代码；**

**表达式变更；**

**}**

**do{**

**执行代码；**

**表达式变更；**

**}while(表达式条件)**

**break: 跳出整个循环**

**continue: 跳出当前循环**

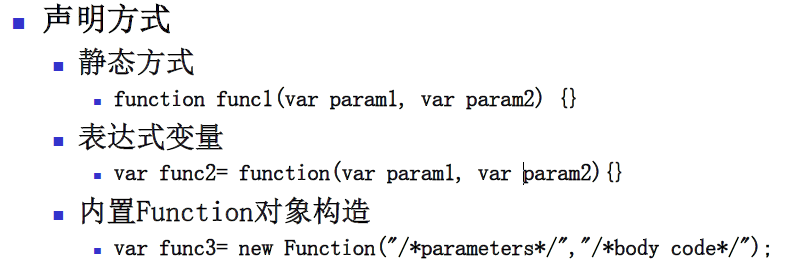
1. **JS在html的位置：**

**1. <head>中，不是立即执行的函数或W3C标准函数**

**2. <body>中，函数或立即执行的语句**

**3. </body>后，立即执行的函数**

**（8） 函数：**

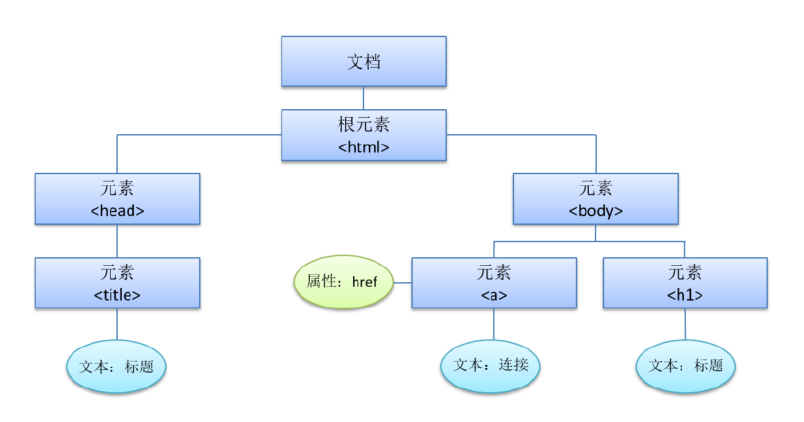


**闭包：能够读取其他函数内部变量的函数**

1. **对象：类比python**
2. **DOM：（DocumentObjectModel）文档对象模型**

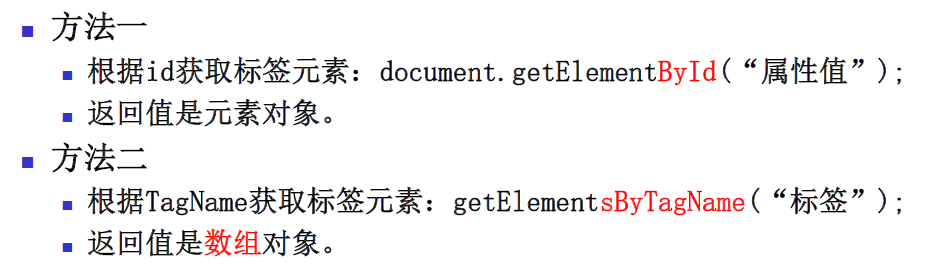
**作用：操作页面元素。**

**节点层次：**

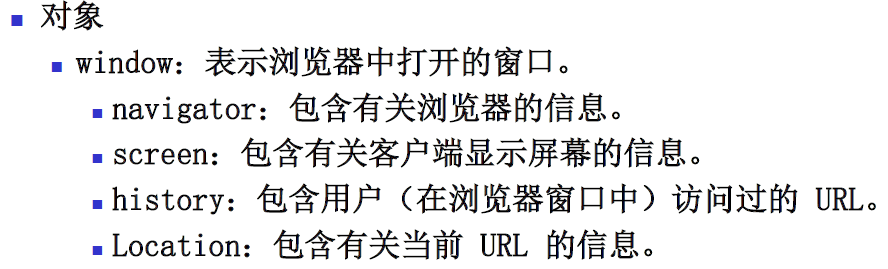


**DOM事件：用户触发页面某个HTML元素的属性时启动一段JavaScript。**

**获取标签元素的方法：**



1. **BOM：(BrowserObjectModal),浏览器对象模型**

**作用：控制浏览器的行为**