July9 JavaScript

1 循环语句

1.1 for语句

for (expr1;expr2;expr3){

…

}

说明：

expr1，在循环之前无条件的执行一次，一般用于变量的初始化；

expr2，在循环前执行一次，以判断能否进入循环体；

expr3，在每次循环后来执行一次，用于步长的计算；

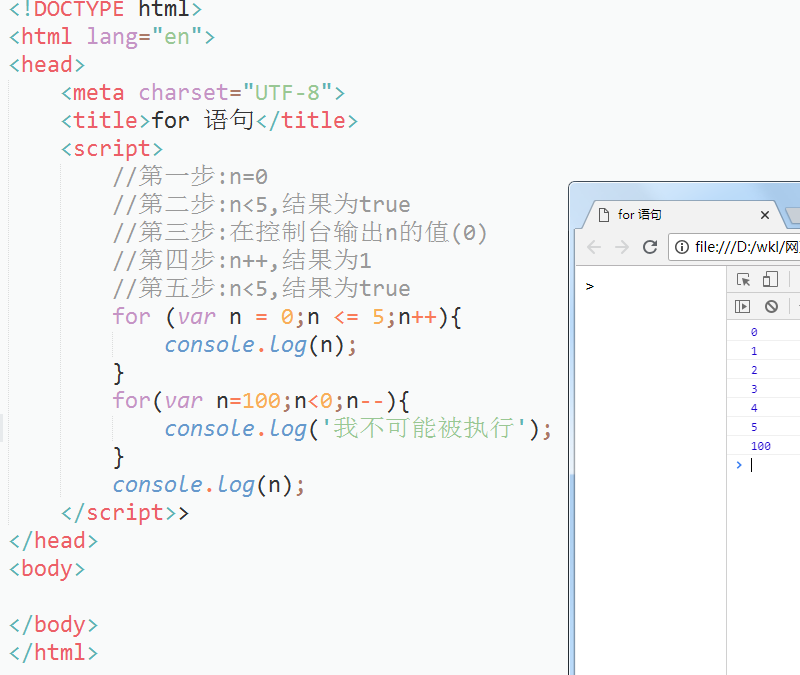
for语句的表达式可以省略其中任何一个或者全部；for（;;）

for(变量n=1;变量n<=5;n++){

…

…

}



1.2 while语句

while（条件表达式）{

…

…

}



for语句和while语句同属“当型循环（先判断后执行）”，所以可以互换；

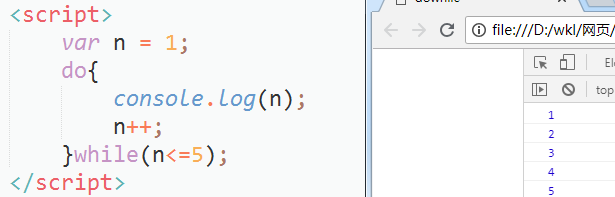
1.3 do…while循环

do{

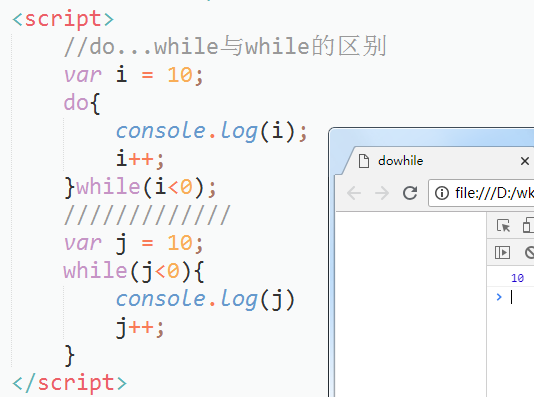
…

…

} while（条件表达式）；



do…while语句是先执行，后判断。而for和while是先判断，后执行



1.4 for…in语句

for(key in array){

…

…

}



2 break语句

结束本次的switch及循环语句

while(true){

if(xxxx){

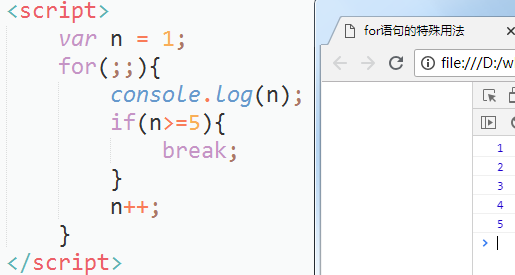
…

…

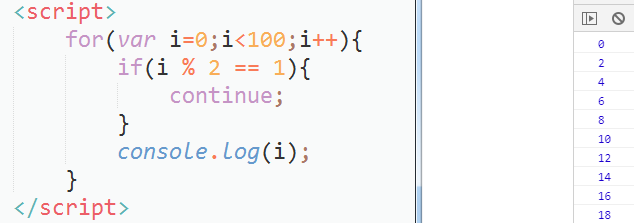
break;

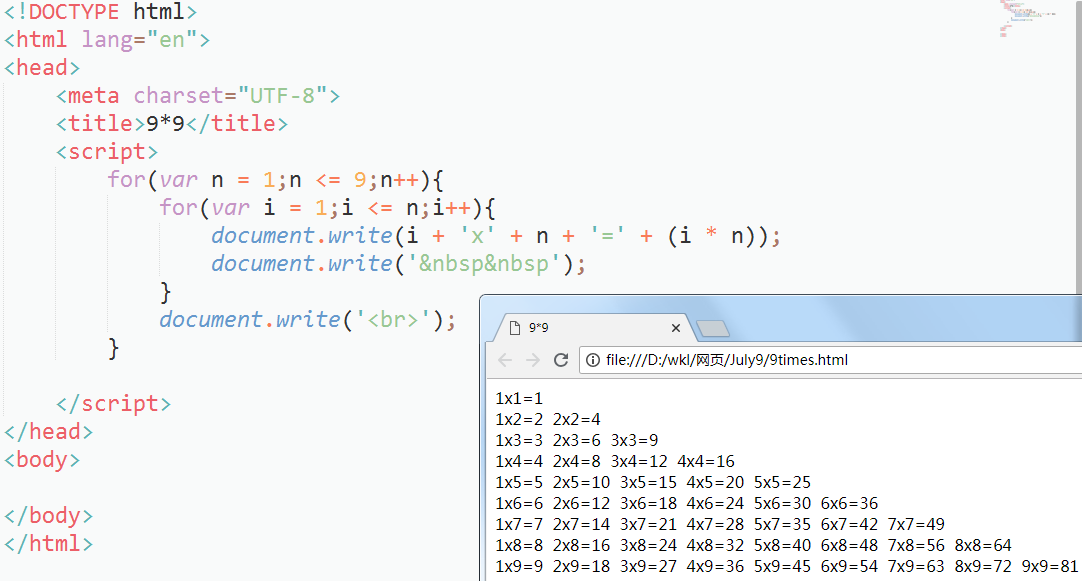
}

}



3 continue语句





4 内置对象

4.1 String对象

构建String对象

1.直接量方式

‘string’或”string”

2.构造函数方式

new String（’string’）

4.2 length

描述：

返回字符串的长度

语法：

int object.lenght

4.3 方法

toLowerCase()

描述：将字符转换小写字母



toUpperCase()

描述：将字符转换为大写字母

语法：string obj.toUpperCase(void)

substr()

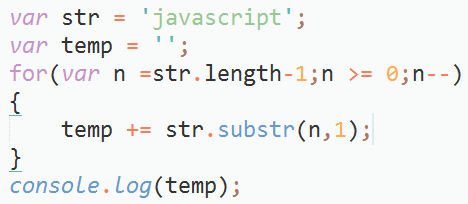
描述：截取字符串

语法：string object.substr(int start[,int length])

说明：字符串从0开始编号；



将字符串反转



substring（）

描述：截取字符串

语法：string object.substring（int str[,int end]）

说明：

1.返回的字符串包含起始位，不包含结束位

2.如果省略end参数，则返回到字符串结尾之间的字符；

indexOf（）

语法：int object.indexOf(string)

描述：返回字符串第一次出现的位置，如果没有出现，则返回“-1”

lastIndexOf（）

描述：返回字符串最后一次出现的位置，如果没有出现，则返回“-1”



UUID：通用唯一识别码，其形态为8-4-4-4-12，其作用是为保证文件名的唯一性

webp格式是Google退出的一种图像格式，其被Google Chrome支持，其特点是相比jpeg文件来说，自己数更少；

replace()方法

描述：字符串替换

语法：stringobject.replace(string search,string replament)

stringobject.replace(string RegExp,string replament)



4.4 新增方法

trim()

描述：删除第一个字符之前及最后一个字符的空白

语法：string object.trim(void)



startsWith()

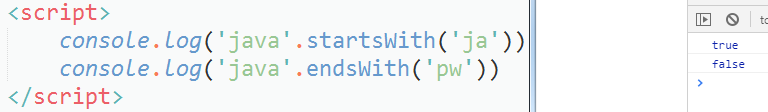
描述：检测字符是否以指定的字符开头

语法：bool object.startsWith(string)

endsWith()

描述：检测字符是否以指定的字符结尾

语法：bool object.endsWith(string)



5 Math对象

Math对象是一个静态对象

5.1 属性

Math.PI 圆周率

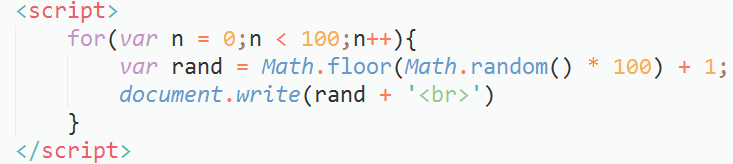
Math.SQRT2 2的平方根

5.2 方法

|  |  |  |
| --- | --- | --- |
| 方法 | 功能 | 语法 |
| Math.ceil() | 进一取整 | number Math.ceil(number) |
| Math.floor() | 向下取整 | number Math.floor(number) |
| Math.pow() | 幂运算 | number Math.pow(base,exp) |
| Math.sqrt() | 平方根 | number Math.sqrt(number) |
| Math.min() | 最小值 | number Math.min(num1，num2，…) |
| Math.max() | 最大值 | number Math.max(num1，num2，…) |

|  |  |  |
| --- | --- | --- |
| Math.random() | 产生介于0~1之间的随机数 | number Math.random(void) |
| Math.round() | 四舍五入（仅保留整数位） | number Math.round(num) |

求100内的随机数



6 Date对象

6.1 构建Date对象

new Date()

6.2 方法

|  |  |  |
| --- | --- | --- |
| 方法 | 功能 | 语法 |
| getYear()  getFullYear() | 获取年份（从公元1900年到现在经历的年份）  获取年份 | int object.getYear()  int object.getFullYear(void) |
| getMonth() | 获取月份（取值0~11） | int object getMonth(void) |
| getDate() | 获取日期（xx号） | int object getDate(void) |
| getDay() | 获取星期的第几天（0为星期日） | int object getDay(void) |
| getHours() | 获取小时数 | int object getHours() |
| getMinutes() | 获取分钟数 | int object getMinutes(void) |
| getSeconds() | 获取秒 | int object getSeconds(void) |
| getMilliseconds() | 获取毫秒 | int object get |
| getTime() | 返回从公元1970年到现在的毫秒 | int object getTime() |
| toUTCString()\  toGMTString() | 将日期转换成UTC/GMT格式 | string object.toUTCString()  string object toGMTString() |



6.3 新增方法

|  |  |  |
| --- | --- | --- |
| 方法 | 功能 | 语法 |
| Date.now() | 从公元1970年到现在的毫秒数 | int Date.now() |

说明：该方法为静态方法

7 Array对象

7.1 构建Array对象

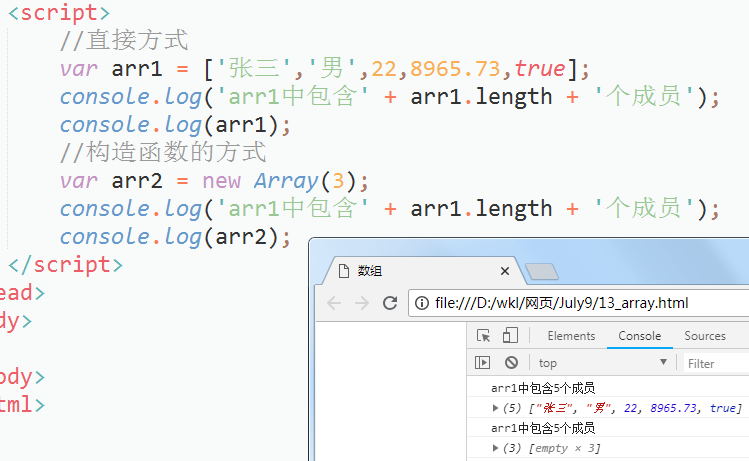
直接量方式

[value,…]

构造函数方式

new Array(length)

new Array(value,…)



7.2 属性

length

描述：获取数组成员的数量

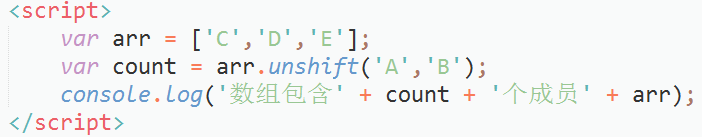
语法：int object.length

7.3 方法

unshift()

描述：在数组的开头添加一个或者多个成员，并返回数组的新长度

语法：int object.unshift(value,…)



shift()

描述：删除数组的第一个成员，并且返回该成员

语法：maxed object.shift()

push()

描述：在数组的末尾添加一个或者多个成员，并返回数组的新长度

语法：int object.push(value,…)

pop()

描述：删除数组的最后一个成员，并且返回该成员

语法：maxed object.pop()



join()

描述：将数组成员使用指定的分隔符，连接成字符串

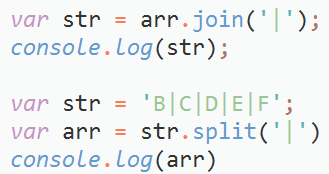
语法：string object.ioin(delimiter)

split()

描述：使用分隔符来拆分字符串

语法：array object.split(delimiter)

array object.split(RegExp)





isArray()

描述：判断是否为数组（静态方法）

语法：bool Array.isArray(obj);



7.4 数组遍历

forEach()语句

object.forEach(function(value[,index]){

…

…

})

