现在开始学习JavaScript

2017-11-26

用的书籍是《JavaScript编程精解》 马尔奇-哈弗贝克 机械工业出版社 2016年

isbn 978-7-111-518060-0

昨天稍微看了3个小时，把语言的一些基本东西看了下，例如

1. 值、类型和运算

和其他编程语言差不多，需要注意的是NaN之类的东西

1. 程序结构

和Java等类似，循环，变量，判断，还有讲了如果写JavaScript，最好使用log而不是alert

1. 函数

定义函数的两种方式

锁定义参数的作用域和Java等等不同，并不是以{}大括号为界，而是以函数为界，这个特性用作闭包等等技术(暂时未能完全看懂)

JavaScript函数的参数很不一样，可以传入多于或者少于定义的参数个数。

一般来说，可以将多个功能分成多个小(或者说是最小的)功能函数，然后调用，这样写出来的程序比较容易调试和理解，同时函数名字要体现一个函数一个功能，不要贪多将多个功能写在一个函数内。

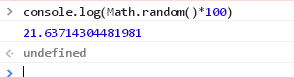
还有一些书写的建议，例如驼峰命名法等等。

1. 对象和数组

和其他编程语言差不多，暂时有点不一样的是arguments对象，这个显示了对象参数的个数

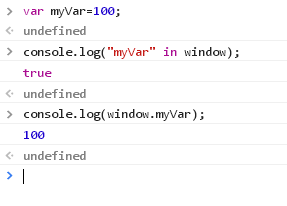
接下来需要学习的是Math对象以及其他。P51开始。

Math对象包含很多与数字相关的工具函数，一般用法是Math.sqrt(这个是求平方根的工具函数)现在试一试



这个函数是log出一个100以内的随机数

全局对象，在浏览器中，全局对象存储在window变量中，例如



接下来是这章的练习题

1. 特定范围数字求和

之前书中的例子，console.log(sum(range(1,10)));这个可以算出1-10之间所有数字之和。

现在需要写一个range函数，接受两个参数start和end，返回包含在内的所有数字。

写一个sum函数，接受一个数字数组，返回所有数字之和。检验之。

附加题：修改range函数，可以接受三个函数，第三个函数是步长。

现在先试试写range函数。它需要的是start和end之间的所有数字，应该可以通过for循环搞定。

试试这个Atom吧，之前没有试过。例如这样写？

function myRange(start,end){

var i=0;

var arr=new Array();

for(start;start<=end;start++,i++){

arr[i]=start;

}

return arr;

}

console.log(myRange(1,5));

汗啊，忘记了函数怎么定义了。

函数定义的方法有

1、

function myFunction(){

}

2、

var myFunction = function(){

}

好像不是这样搞……

Atom貌似不会用……

还是用Sublime吧

看来我不会写JavaScript，先复习一下。

继续，稍微复习了一下，算是可以写出function了，应该改成这样

function myRange(start,end){

var i=0;

var arr=new Array();

for(start;start<=end;start++,i++){

arr[i]=start;

}

return arr;

}

console.log(myRange(1,5));

成功写出range函数。

接下来是sum函数：接受一个数组，返回所有数字之和。很简单嘛~

function mySum(a){

var sum=0;

for(var i=0;i<a.length();i++){

sum+=a[i];

}

return sum;

}

噢，出现问题的原因应该是arr并没有被定义，因为它是myRange函数内的，所以出了myRange函数，它就没有了。

现在出现这个问题NaN，是为什么呢？是不是因为传入的不是数组？

？？？？

噢，原来是初始化问题。也就是说，每一个变量最好在定义的时候就初始化。

好的，mySum函数完成。

接下来是构建步长step，修改myRange。

好的，修改完成，修改后的myRange函数如下

function myRange(start,end,step){

var i=0;

var arr=new Array();

for(start;start<=end;start+=step,i++){

arr[i]=start;

}

return arr;

}

好像还有一点问题，不能倒过来输出，例如5,4,3,2,1这样。

问题应该是这里start<=end，这个条件不严谨。

function myRange(start,end,step){

var i=0;

var arr=new Array();

if(start<end){

for(start;start<=end;start+=step,i++){

arr[i]=start;

}

}

else{

for(start;start>=end;start+=step,i++){

arr[i]=start;

}

}

return arr;

}

使用一个if判断条件，算是满足要求了。

1. 逆转数组

数组有一个reverse方法，可以逆转数组中元素的次序。现在需要手写一个reverse方法：

reverseArray方法接受一个数组作为参数，逆转之。

reverseArrayInPlace与上一个函数功能相同，但是直接将数组作为参数进行修改其中数组中的元素次序。

不太明白，先去看看数组的reverse方法吧。

数组的reverse方法用于改变数组的次序，颠倒一次，不是创建新的数组，而且改变原来数组的次序，永久改变。

reverseArray方法，嗯，应该怎样写好呢？

传入一个数组，传出一个数组。

function reverseArray(a){

var trans = new Array();

for(var i=0;i<a.length;i++){

trans[i] = a[i];

}

for(var j=trans.length-1,i=0;j>=0;j--,i++){  
 a[i]=trans[j];

}

return a;

}

貌似可以了。

reverseArrayInPlace的意思应该是如果reverseArrayInPlace([1,2,3,4,5],2),结果应该是[1,2,5,4,3]

function reverseArrayInPlace(a,index){

var trans = new Array();

for(var i=index,j=0;i<a.length;i++,j++){

trans[j] = a[i];

}

//console.log(trans);

for(var j=trans.length-1,i=index;j>=0;j--,i++){  
 a[i]=trans[j];

}

return a;

}

哪里出问题呢？

数组多了两位，为何？？

trans[i] = a[i];

这里出问题了，应该改一改。

???

好了。任务完成。

另外，题目要求回想一下关于副作用和纯函数的对比，哪种写法应用场景更广。

已经不记得了，先看看吧。

函数可以分为两类，一种是调用后产生副作用，一种是产生返回值。

纯函数是一种只会产生值而不会影响它自己范围外任何事情的函数。也就是说，刚刚我写出的reverseArrayInPlace是属于哪个呢？有返回值，然后没有影响其他东西吗？好像是的。

如果像刚刚那样，在reverseArrayInPlace函数内写一个console.log，那就是有”输出”这一个额外的东西，就是副作用了。

哪种写法应用场景更广？我觉得是纯函数更广。因为它单元的，不影响其他东西，更可控。

1. 实现列表

列表(list)，一种数据结构，但是一种嵌套对象集合，第一个对象拥有第二个对象的引用，第二个对象拥有第三个对象的引用，如此类推。

var list = {

value:1,

rest:{

value:2,

rest:{

value:3,

rest:{

...

}

}

}

};

需要写一个函数arrayToList,给定参数(数组)时，建立一个列表

写一个函数listToArray,将列表转换成数组

写一个工具函数prepend，接受一个元素和列表，然后创建一个新列表，将元素添加到输入列表的开头

最后写一个nth，接受一个列表和书，返回列表中指定位置的元素，如果元素不存在，则返回undefined

附加题：递归版本的nth函数

哈哈哈，完全不懂

列表应该怎样表示出来？

先这样哈~~

既然不懂，那就百度咯

好像这个可以，仔细看看代码构成吧~

var arrList = [];

for(var i=0;i < 10; i++){

 arrList[i] = "data" + i;

}

alert(arrList);

var arrMap = [];

arrMap["name1"] = "data1";

arrMap["name2"] = "data2";

arrMap["name3"] = "data3";

for(o in arrMap){

  alert(arrMap[o]);

}

？？？

书中提示的信息是{value:0,result:list}这样，如果这样新建值，就会有一个独立的列表，包含列表末尾的三个元素？？

var list=[

{“value”:value,”Id”:id;next:”next”},

];

??

使用对象？？

var ele={

value:??,

next:??,

};

搞不懂……

2017-11-26 22时48分47秒

2017-11-28 11时1分16秒

复习了一下对象的概念，使用的是犀牛书

对象是JavaScript的基本数据类型，将多个值聚合在一起，可以通过名字来访问，是无序集合，每个属性值都是一个键值对，JavaScript中的对象属性是动态的，除了字符串，数字和布尔值，null和undefined之外，所有的值都是对象，每个对象拥有三个相关的对象特性：对象的原型，对象的类，对象的扩展标记。对象的常见用法是：创建，设置，查找，删除，检测和枚举。

继续学习数组 2017-11-28 11时17分28秒 犀牛书

1. 深度比较
2. 高阶函数

高阶函数是使用函数作为传入参数或者传出结果的函数，函数内又有函数。