练习：

使用遍历数组元素实现翻转数组中的元素，不是使用reverse

[‘a’,’b’,’c’,’d’,]->[‘d’,’c’,’b’,’a’]

学习冒泡排序

//1.翻转数组元素

var arr=['a','b','c','d'];

//倒着获取每个元素，放入到新数组中

var newArr=[];

//遍历arr,倒着获取每个元素

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

/\*

0 3 4

1 2 4

2 1 4

3 0 4

长度-1-i

\*/

//console.log(arr[arr.length-1-i]);//长度-1-i

//把找到的元素依次放入到新数组中

newArr[i]=arr[arr.length-1-i];

}

//console.log(newArr);

复习

数组

存储数据

创建

数组字面量 [ 元素1,元素2....]

内置构造函数 new Array(元素1,元素2....);

数组元素访问 数组名称[0]

数组的长度 数组.length 在末尾添加元素 获取最后一个元素

数组的分类 索引数组 关联数组

数组的遍历

for(var key in 数组){ }

for(var i=0;i<数组.length;i++){ }

数组的API

toString()/join()/concat(arr2,arr3...)/slice(start,end)

splice(start,count,value1,..)reverse()/sort()

1. 数组API

push() 在数组的末尾添加元素，返回数组的长度

pop() 在数组的末尾删除元素，返回删除的元素

unshift() 在数组的开头添加元素，返回数组的长度

shift() 数组的开头删除元素，返回删除的元素

1. 二维数组

数组中的每个元素也是数组

|  |
| --- |
| var city=[[],[],[]...] |

访问二维数组中的元素 city[下标][下标]

1. 字符串操作

包装对象：目的就是让原始类型的数据也可以像引用类型数据一样，具有一组属性和方法。

JS中提供了3种包装对象：String,Number,Boolean

new String(1) 返回object

String(1) 返回string

用法上是一致的。

（1）转义字符-----

转换字符本身的意义

\n 将字符n 转成了换行符

\’ 将绰号转成普通的引号字符

\t 将字符t转成了制表符(tab键的效果)

练习：打印 welcomet to chi\na

console.log('welcomet to chi\\na');

1. 字符串API

length 获取字符串的长度

charAt() 查看下标对应的字符

charCodeAt() 查看某个字符的Unicode码

indexOf() 查找某个字符第一次出现的下标，找不到返回-1

练习：声明变量存储邮箱，检验邮箱是否为合法的合适（是否有@）；

如果是打印true,否则打印false

var email='747634280@qq.com';

//判断邮箱中是否含有@

/\*if(email.indexOf('@')==-1){

console.log(false)

}else{

console.log(true);

}\*/

toUpperCase() 将英文字母转为大写

toLowerCase() 将英文字母转为小写

练习：初始化4个英文字母(有大写小写)保存到变量中，循环弹出提示框，输入验证码(不区分大小写)，如果输入正确结束循环。、

06\_checkcode.html 06\_checkcode.js

//初始化4个英文字母

var code='ABEm';

//无限循环弹出提示框

do{

var str=prompt('input checkcode');

//比较和保存的4个英文字母是否一致

//都统一转成大写

if(str.toUpperCase()==code.toUpperCase()){

alert('right');

break;

}

}

while(true);

slice(start,end) 截取字符串，start是开始的下标，end是结尾的下标，不包含end本身

substr(start,count) 截取字符串，start是开始的下标，count是截取的长度

练习：声明变量保存邮箱，从中分别截取出用户名和域名

Tom123@sina.com

var email='tom123456@163.com';

//获取@的下标

var i=email.indexOf('@');

//console.log(i);//9

console.log(email.slice(0,i));//截取用户名//tom123456

console.log(email.slice(i+1));//截取域名//163.com

练习：声明变量保存省份证号，从中截取出出生的年月日，获取性别(倒数第2位)，打印1999年12月21日 女

440881199912213789

var sid='440881199912213789';

var year=sid.substr(6,4);

var month=sid.substr(10,2);

var date=sid.substr(12,2);

var sex=sid.substr(-2,1);

var s=sex%2==0 ? '女':'男';

console.log(year+'年'+month+'月'+date+'日 '+s)

split() 按照指定的字符将字符串分隔为数组

1. 匹配模式（掌握）

作用：用于查找，替换字符串

replace(value1,value2) 查找并替换。value1要查找的字符串，value2 要替换的字符串。

value1既可以使用字符串形式，也可以使用正则表达式形式 /tom/ig

g -> global 全局查找

i -> ignore 忽略大小写

match(value) 查找匹配的字符串，返回数组，

search(value) 查找匹配的第一个字符，返回下标，找不到返回-1

1. Math 对象

Math 对象不需要使用new来创建，可以直接使用。

Math .PI 获取圆周率

abs() 取某个数的绝对值

ceil() 向上取整

floor() 向下取整

round() 四舍五入取整

max() 取一组数字的最大值

min() 取一组数字的最小值

pow(x,y) 取x的y次幂

random() 取随机数 范围0~1 >=0 <1

练习：随机获取0~9之间的整数

var num=Math.random()\*10;

//向下取整

num=Math.floor(num);

console.log(num);

练习：

将一句英文的每个单词首字母大写，其余小写

hoW aRe YOU -> How Are You

创建数组，包含a~z和0~9，在数组中随机取4个字符，放到一个新数组中