11对象

是应用类型数据

（1）对象：是一组属性和方法（功能）的集合。

哪些是对象？

一个手机：属性：品牌、尺寸。方法：打电话，发短信

一个人：属性、年龄、姓名。方法：工作，说话

万物皆对象

（2）JS中的对象

内置对象：JS提供的

宿主对象：根据JS不同的执行环境来划分

自定义对象：自己创建的对象

（3）自定义对象

对象字面量

内置构造函数

自定义构造函数

（4）使用对象字面量创建对象

使用大括号{}创建对象

属性名和属性值之间用冒号隔开

多组属性之间用逗号隔开

属性名中引号可加可不加，如果含有特殊字符必须加引号

{属性名：属性值…}

练习：创建一个人对象，添加姓名，性别，年龄

练习；创建一个员工对象，包含编号、姓名、性别、生日、工资、所属部门编号

（5）访问对象的属性

对象属性名

对象['属性名']

可以添加之前不存在的属性；

如果属性不存在，属性值为udefined.

/\*var pople={

eid:1,

name:'hh',

sec:'nv',

birthday:'1996',

salary:20000,

'dept-id':20

}

//console.log(pople)

//访问对象中的属性值，获取对象中的属性值

console.log(pople.name)

console.log(pople['dept-id'])

//访问对象中不存在的属性返回undefined;

console.log(pople.size)

//修改对象中的属性值

pople.salary=10000;

pople['birthday']='2019-12'

console.log(pople)\*/

练习：创建一个电脑对象，包含的属性有屏幕尺寸，品牌，颜色，内存大小，获取品牌的属性值，修改颜色，添加产地

var computer={

size:5.5,

brand:'vivo',

color:'blue',

memory:'16G',

}

console.log(computer);

console.log(computer.brand);

computer.color='red'

computer.madein='成都'

console.log(computer)

（6）、使用内置构造函数创建对象

new Object(); 创建一个空的对象

需要单独添加每一个属性

对象['属性名']=‘属性值’

对象.属性名=‘属性值’

//使用内置构造函数创建对象

/\*var car=new Object();

car.id=123;

car.brand='BMW';

car['color']='black';

console.log(car)8\*/

练习：使用内置构造来创建用户对象，包含编号，用户名，密码，邮箱，手机

（7）、遍历对象的属性

访问所有的属性

for(var key in 对象){

key，//对象中所有属性名

对象[key] //通过属性名找对应的属性值

}

/\*var person={

name:'tom',

age:18,

sex:'男'

};

for(var key in person){

//key对象中每一个属性名

console.log(key,person[key])

}\*/

练习：创建对象，包含若干个成绩属性，遍历对象中属性获取总成绩

提升声明sum用于保存总和

var total={

//name:'mak',

//age:19,

//sex:'女',

math:99,

china:75,

english:78,

history:89

}

var sum=0;

for(var key in total){

//console.log(key,total[key])

//把每一个属性值加到sum中

sum+=total[key];

}

console.log(sum)

（8）、对象中的方法

方法也称为成员方法，对应的是一个匿名函数

var person={

name:'tom',

say:function(){

this.name//this指当前所在的对象

}

}；

person.say()//调用方法

创建一个圆对象，包含半径、圆周率属性；添加获取面积和获取周长的方法，分别返回对应的值。

尝试可以构造函数创建对象，并添加属性和方法。。

（9）检测对象中是否含有某个属性

对象属性名===undefined,如果是true——>不存在，false——>存在。

比较一个属性的值是否为undefined

console.log(person.sex===undefined);

对象.hasOwnProperty('属性名');如果是true——>存在，false——>不存在

//查看是否有自己的某个属性

console.log(person.hasOwnProperty('name'));

‘属性名’in 对象；如果是true-->存在，false🡪不存在

//查看某个属性是否存在与对象中

console.log('phone'in person);

/\*var book={

id:1002,

title:'javascript高级程序设计',

price:79

}

//把book变量中存储的数据赋给book2变量

//把现在book存储的地址赋给了book2

//现在

var book2=book;

//修改你book中属性

book.price=86

console.log(book2)

book2.title='javascript权威指南';

console.log(book2)

//fn提升到最前边

//var fn

//function fn(){}//把一个函数赋给fn

console.log(fn)

var fn=1;//再次赋盖了fn中的值

function fn(){

}

console.log(fn);

12、数组

数组是由多个元素组成的集合，每个元素就是一个数据。

（1）创建数组

①数组字面量

['元素1'，'元素2'，…]

练习：创建数组，包含多个商品的名称；创建数组，包含多个城市名称

var mp=['牛奶','鸡蛋','牛肉','西红柿'];

//console.log(mp);

var ct=['成都','都江堰','乐山','德阳','重庆'];

//console.log(ct);

②访问数组中的元素

数组[下标] 下标是从0开始，第一个元素的下标就是0

//通过下标来获取元素

console.log(ct[4]);

ct[5]='布鲁塞尔';//添加元素

ct[4]='米兰'//修改其中的元素

console.log(ct)

③内置构造函数

new Array(元素1，元素2，…)

new Array(3) 初始化元素个数为3，也可以添加更多元素

练习：创建数组，保存若干个课程名称

练习：创建数组，初始化长度为5，添加篮球场上的五个位置

var cal=new Array('数学','语文','英语','物理','化学')

console.log(cal)

var bas=new Array(5);

bas[0]='中场';

bas[1]='后场';

bas[2]='前锋';

bas[3]='后卫';

bas[4]='主攻';

console.log(bas)

var car=new Array('奔驰','宝马','奥迪');

var car2=new Array(3);

car2[0]='凯迪拉克';

car2[1]='沃尔沃';

car2[2]='捷豹';

car2[3]='路虎';

console.log(car2)

④数组的长度

数组.length 获取数组中元素的个数

作用：在数组的末尾添加新的元素

数组[数组.length]=值；

var bas=new Array(5);

bas[0]='中场';

bas[1]='后场';

bas[2]='前锋';

bas[3]='后卫';

bas[4]='主攻';

bas[bas.length]='进攻后卫'

//console.log(bas)

//获取数组的长度或者个数

console.log(bas.length)

练习：创建一个空数组，使用数组长度添加若干个国家名称

var cat=new Array(6);

cat[cat.length]='巴黎';

cat[cat.length]='里约';

cat[cat.length]='伦敦';

cat[cat.length]='米兰';

cat[cat.length]='埃塞俄比亚';

cat[cat.length]='西班牙';

console.log(cat)

(2)数组的分类

数组分为索引数组和关联数组

索引数组以0以上的整数作为下标

关联数组义字符串作为下标，只能单独的添加元素

（3）遍历数组

①for--in

for(var key in 数组){

key 要遍历的下标

数组[key] 下标对应的元素

}

既可以遍历关联数组，也可以遍历索引数组

练习；创建数组，包含有多个学生的成绩，获取总成绩

var sum=0

var studes=[65,98,78,85,99,88,48]

//for (var key in studes )

//{sum+=studes[key];

//}

//console.log(sum)

②循环

for(var i=0;i<=数组.length;i++){

I 代表下标 0-数组长度

数组[i] 代表下标对应的元素

}

只能遍历索引数组

//使用for循环获取下标0-4，使用下标获取元素

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

//i代表下标

//studes[i]下标对应的元素

console.log(i,studes[i])

sum+=studes[i]

}

console.log(sum)

练习：创建数组，包含多个姓名，把tom全部改成'汤姆'。

参加数组

遍历得到每一个元素

判断进这个元素是否为Tom汤姆，如果是，重新赋值为汤姆在遍历之后打印数组

var name=['tom','make','jieke','linda'];

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

{

if (name[i]=='tom')

{name[i]='汤姆'

}

}console.log(name)

//创建数组，包含多个姓名，查询Tom出现的次数

//创建数组

//遍历数组

//初始化变量的次数为0

//判断每个元素是否为tom,如果是次数加1

/\*var name=['tom','mak','jiexi','tom','timu','linaad']

for (var i=0,str=0;i<=name.length ;i++ )

{if (name[i]=='tom')

//如果每个元素为tom，则次数加一

{str++

}

}console.log(str)\*/

//练习：创建数组，包含多个数字，获取这组数字地最大值

//创建数组

////遍历数组元素

//声明变量用于保存最大值，默认把第一个元素放进去

//使用之前声明的变量和每个元素比价，如果变量小于任何一个元素，咋把该元素放入到变量中

var a=[25,56,98,99,102,015,250,126,380,666]

var max=a[0]

for (var i=1;i<=a.length ;i++ )

{//用max和数组每个元素比较

//如果小于任意一个元素，则把该元素放入到max

if(max<a[i])

{max=a[i]

}

}console.log(max)

练习创建函数getAvg，传递一个参数（一组工资），返回工资的平均值

实参就是一个数组，包含多个工资；

函数体中，计算元素的和，求平均

平均=总和/元素长度

(4)数组API(方法)

API应用程序编程接口，预定义好的一些方法或者函数

toString() 将数组中的元素按照逗号组合成字符串

join('-')将数组中的元素按照指定字符组合成字符串，默认是逗号

concat(arr2,arr3) 拼接多个数组

slice(start，end) 截取数组中的元素,start开始的下标，end结束的下标，不包含end本身，如果是负数表示倒数；返回一个数组

splice(start,count,value1,value2…)删除数组中的元素，start开始的下标，count删除的数量，value删除后插入的元素；返回删除的元素，原数组会发生变化

reverse()翻转数组中的元素

sort()对数组中的元素进行排序，默认按照unicode码从小到大【】排序

sort(function(a,b){a-b})从小到大排序

sort(function(a,b){b-a})从大到小排序

/\*var arr=['tom','jerry','king'];

var arr2=['david','lucy']

var arr3=['smith','white']

//数组转为字符串

console.log(arr.toString() );

console.log(arr.join('-'))

//拼接多个数组中的元素

console.log(arr.concat(arr2,arr3))\*/

/\*var arr=['java','html','css','ajax','vue','react','angular'];

//截取

console.log(arr.slice(1,4))

console.log(arr.slice(-3,-1))\*/

练习创建数组a-g,每个字母是一个元素，分贝截取cd,f，b,把截取的结果拼接成一个新的数组

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

var a=arr.slice(2,4)

var b=arr.slice(5,6)

var c=arr.slice(1,2)

console.log(a.concat(b,c))

//删除数组中的元素

/\*var arr=['html','css','js','nodejs','ajax','vue','react','angular'];

//console.log(arr.splice(2,2))

//console.log(arr.splice(-3,1))

console.log(arr.splice(-3,0,'ionic','h5','c++'))

console.log(arr)\*/

//翻转数组中的元素

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

console.log(arr.reverse())

练习创建数组a-h，每个字母是一个元素，删除d,e替换f为m,在下标为1的位置插入字母z

var arr=['a','b','c','d','e','f','g','h']

console.log(arr.splice(3,2))

console.log(arr.splice(-3,1,'m'))

console.log(arr.splice(1,0,'z'))

console.log(arr)