**对象**

JS中的对象是一组属性和方法的集合。

JS中的对象分为三类：1）原生对象；2）宿主对象；3）自定义对象。

1.对象直接量(自变量)

1)使用对象直接量创建对象：

使用大括号{}创建对象；

多个属性之间用逗号隔开；

属性值与属性名之间用冒号隔开；

如果对象中出现了空格，-，属性名必须添加引号，其他情况引号可以不加。

例：var phone={

color:'white',

'o-s':'Android' //特殊符号加引号

}

2)单独访问对象直接量：

例：console.log(phone.color);

console.log(phone[‘o-s’]); // [ ]形式访问加引号的属性

3)添加或者修改对象中的属性

phone.size=’5.5’; 或 phone[‘size’]=’5.5’;

2.内置构造函数

1)使用内置构造函数创建对象—工厂函数

new Object()

函数封装：把重复的代码放到函数中 对象封装：把重复的代码放到对象中

例：function c(name,color){

var o=new Object();

o.name=name;

o.color=color;

return o;

}

console.log(c('tom','yellow'));

console.log(c('jerry','grey'));

3.遍历属性 for in

例：var dog={

name:'大黄',

color:'yellow',

type:'中华田园犬',

age:2

}

for(var key in dog){

console.log(key); //name color type age

console.log(key+’:’+dog[key]); // name:大黄 color:yellow type:中华田园犬 age:2

}

注：for(var key in 对象){

//key表示要遍历的所有属性

//如果要获取属性值只能使用—对象[key]

}

4.对象中的方法

例：var c={

r:10,

len:function(){

console.log('周长='+2\*3.14\*this.r);

},

s:function(){

console.log('面积='+3.14\*this.r\*this.r);

}

}

c.len();

c.s();

5.检查对象中是否含有某个属性

例：var person={

name:'Tom',

age:20,

gender:'男'

}

console.log(person.hasOwnProperty('name')); //1.内置的预定义成员方法hasOwnProperty()

console.log('gender' in person); //2.使用in关键字

console.log(person.age1===undefined); //3.和undefined进行对比

if(person.age){ //4.在条件语句中判断（undefined\null\0\空字符串都会自动转换为false）

console.log('age属性存在');

}

6.数据在内存中的存储方式 栈内存 & 堆内存

原始数据类型存在栈内存中，存储的就是数据的值

引用数据类型存在堆内存中，存的是对象的地址

例：var i=123, //栈内存

obj={age:12}; //堆内存

function change(num){

num--;

obj.age--;

}

change(i);

console.log(i,obj.age); //123 11