JavaScript 中的所有事物都是对象：字符串、数值、数组、函数...

此外，JavaScript 允许自定义对象。

## **JavaScript 对象**

对象只是一种特殊的数据。对象拥有****属性****和****方法****。

## **访问对象的属性**

属性是与对象相关的值。

访问对象属性的语法是：

*objectName.propertyName*

这个例子使用了 String 对象的 length 属性来获得字符串的长度：

var message="Hello World!";  
var x=message.length;

## **访问对象的方法**

方法是能够在对象上执行的动作。

您可以通过以下语法来调用方法：

*objectName.methodName()*

## **创建直接的实例**

这个例子创建了对象的一个新实例，并向其添加了四个属性：

person=new Object();  
person.firstname="John";  
person.lastname="Doe";  
person.age=50;  
person.eyecolor="blue";

替代语法（使用对象 literals）：

## **实例**

person={firstname:"John",lastname:"Doe",age:50,eyecolor:"blue"};

## **使用对象构造器**

本例使用函数来构造对象：

## **实例**

function person(firstname,lastname,age,eyecolor)  
{  
    this.firstname=firstname;  
    this.lastname=lastname;  
    this.age=age;  
    this.eyecolor=eyecolor;  
}

在JavaScript中，this通常指向的是我们正在执行的函数本身，或者是指向该函数所属的对象（运行时）

## **创建 JavaScript 对象实例**

一旦您有了对象构造器，就可以创建新的对象实例，就像这样：

var myFather=new person("John","Doe",50,"blue");  
var myMother=new person("Sally","Rally",48,"green");

## **把属性添加到 JavaScript 对象**

您可以通过为对象赋值，向已有对象添加新属性：

假设 personObj 已存在 - 您可以为其添加这些新属性：firstname、lastname、age 以及 eyecolor：

person.firstname="John";  
person.lastname="Doe";  
person.age=30;  
person.eyecolor="blue";  
  
x=person.firstname;

T在以上代码执行后，x 的值将是：

John

## **把方法添加到 JavaScript 对象**

方法只不过是附加在对象上的函数。

在构造器函数内部定义对象的方法：

function person(firstname,lastname,age,eyecolor)  
{  
this.firstname=firstname;  
this.lastname=lastname;  
this.age=age;  
this.eyecolor=eyecolor;  
  
this.changeName=changeName;  
function changeName(name)  
{  
this.lastname=name;  
}  
}

## **JavaScript for...in 循环**

JavaScript for...in 语句循环遍历对象的属性。

### **语法**

for (variable in object)  
{  
  code to be executed  
}

****注意：****for...in 循环中的代码块将针对每个属性执行一次。

### **实例**

循环遍历对象的属性：

## **实例**

var person={fname:"John",lname:"Doe",age:25};   
  
for (x in person)  
  {  
  txt=txt + person[x];  
  }