Javascrpit：



Javascript变量：

构造变量名称（唯一标识符）的通用规则是：

* 名称可包含字母、数字、下划线和美元符号
* 名称必须以字母开头
* 名称也可以 $ 和 \_ 开头（但是在本教程中我们不会这么做）
* 名称对大小写敏感（y 和 Y 是不同的变量）
* 保留字（比如 JavaScript 的关键词）无法用作变量名称

**重复声明 JavaScript 变量**

如果再次声明某个 JavaScript 变量，将不会丢它的值。

在这两条语句执行后，变量 carName 的值仍然是 "porsche"：

**实例**

var carName = "porsche";

var carName;

javescript数据类型：

<https://www.w3school.com.cn/js/js_datatypes.asp>

要看

**JavaScript 对象**

**typeof 运算符**

**Undefined**

**空值**

空值与 undefined 不是一回事。

空的字符串变量既有值也有类型。

**实例**

var car = ""; // 值是 ""，类型是 "string"

**Null**

**Undefined 与 Null 的区别**

**复杂数据**

typeof 运算符可返回以下两种类型之一：

* function
* object

typeof 运算符把对象、数组或 null 返回 object。

typeof 运算符不会把函数返回 object。

**实例**

typeof {name:'Bill', age:62} // 返回 "object"

typeof [1,2,3,4] // 返回 "object" (并非 "array"，参见下面的注释)

typeof null // 返回 "object"

typeof function myFunc(){} // 返回 "function"

[亲自试一试](https://www.w3school.com.cn/tiy/t.asp?f=js_datatypes_typeof_complexdata)

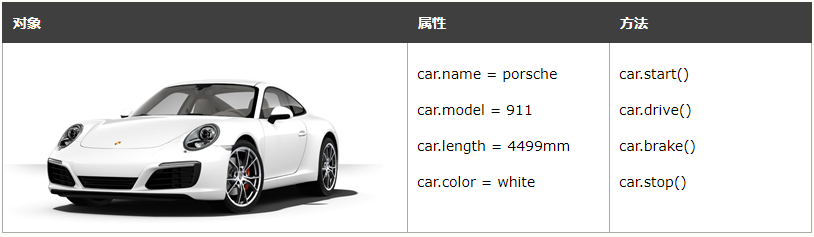
typeof 运算符把数组返回为 "object"，因为在 JavaScript 中数组即对象。

Javescript对象：

**真实生活中的对象、属性和方法**

在真实生活中，汽车是一个**对象**。

汽车有诸如车重和颜色等**属性**，也有诸如启动和停止的**方法**：



所有汽车都拥有同样的**属性**，但属性值因车而异。

所有汽车都拥有相同的**方法**，但是方法会在不同时间被执行。

This：

var person = {

firstName: "Bill",

lastName : "Gates",

id : 678,

fullName : function() {

return this.firstName + " " + this.lastName;

}

};

**访问对象属性**

您能够以两种方式访问属性：

*objectName*.*propertyName*

或者

*objectName*["*propertyName*"]



Javascript类型转换：

Date().toString() // 返回 "Wed Jun 10 2020 16:48:58 GMT+0800 (中国标准时间)"

**一元 + 运算符**

**一元的 + 运算符**可用于把变量转换为数字：

**实例**

var y = "5"; // y 是字符串

var x = + y; // x 是数字

javascript正则表达式：

代码：

<!DOCTYPE html>

<html>

<body>

<h2>JavaScript 正则表达式</h2>

<p>检索下面段落中的一个 "1"：</p>

<p id="p01">The best things in life are free!</p>

<p id="demo"></p>

<script>

bext = document.getElementById("p01").innerHTML;

document.getElementById("demo").innerHTML = /1/.test(bext);

</script>

</body>

</html>

结果：

**Javascript 正则表达式**

检索下面段落中的一个 "1"：

The best things in life are free!

false

输入用法：

使用 window.alert() 写入警告框

* 使用 document.write() 写入 HTML 输出
* 使用 innerHTML 写入 HTML 元素
* 使用 console.log() 写入浏览器控制台

**JavaScript 作用域**

<https://www.w3school.com.cn/js/js_scope.asp>重要，细看

js hoisting

**JavaScript 初始化不会被提升**

JavaScript 只提升声明，而非初始化。

<https://www.w3school.com.cn/js/js_hoisting.asp>

**JavaScript Use Strict**

Javascript this关键字：

<https://www.w3school.com.cn/js/js_this.asp>

**JavaScript 样式指南和代码约定】**

<https://www.w3school.com.cn/js/js_conventions.asp>

**JavaScript 最佳实践**

<https://www.w3school.com.cn/js/js_best_practices.asp>

**JavaScript 表单**

<https://www.w3school.com.cn/js/js_validation.asp>

**JavaScript 验证 API**

<https://www.w3school.com.cn/js/js_validation_api.asp>