

JavaScript简介

- JavaScript 是 Sun Microsystems 和 Netscape 共同开发的一种非常重要的脚本语言。使用它可以创建动态的、交互式的 Internet 网页

什么是 JavaScript

- JavaScript 是一种基于对象的脚本语言，用于开发基于客户端和基于服务器的 Internet 应用程序
- 客户端应用程序在浏览器中运行
- 服务器应用程序则在 Web 服务器中运行

JavaScript 的作用和规则

- JavaScript 是一种可以嵌入到HTML网页中的编程语言
- 使用JavaScript可以：

- 提供用户交互
- 动态更改内容
- 验证数据

JavaScript 编程规则：

- 区分大小写
- 使用成对的符号
- 使用空格
- 使用注释

变 量

- 变量命名规则

- 变量名必须以字母或下划线开头
- 变量名可以包含数字
- 由于JavaScript区分大小写，所以可用的字母包括A到Z (大 写)和 a 到 z (小写)的字符

声明变量

变量是使用关键字 `var` 声明的

变量从它第一次被使用时，开始存在

可以在声明时初始化变量，也可以稍后在脚本中进行初始化

```
var A = 10;
```

也可以通过向变量赋值来声明变量 `A = 10;`

数据类型

- JavaScript 有四种基本的数据类型
 - 数字型(Number)
 - 逻辑型或布尔型(Boolean)
 - 字符串型(String)
 - 空型(null)
- JavaScript 中的基本类型属于弱类型

条件语句

- 条件语句用于判断条件，判断结果决定了要执行哪个语句或语句块
- 语法：
- If ... else
- 三种输写方式：
 - If (condition)
 - {statements }
 - If (condition)
 - {statements }
 - Else
 - {statements }
 - If (condition)
 - {statements }
 - Else If (condition)
 - {statements }
 - 其中 condition是条件， statements 是语句

条件语句

- 三元表达式
- 语法:
 - `(condition)?true val:false val`
 - 如果条件为true则将指定的值赋给变量;否则条件为false 将另一个值赋给变量
 - 例如:
 - `Status = (Age >= 18) ? “adult” : “minor”`
 - 如果Age大于或等于18,则将值adult赋给status,否则将值minor赋给它

循环语句

- 控制程序中重复执行的操作的结构称为循环
- For 循环
 - For 语句功能最强，也最复杂
 - For 循环重复执行语句块,直至测试条件为false
 - For 语句由三个用分号分隔的参数组成
 - For (initial-statement ; condition ; increment)
 - { statements; }
 - initial-statement(初始化语句) - 通常用于初始化数字计数器变量
 - Condition (循环条件) – 设置循环的条件
 - Increment (增量语句) – 更新循环计数器

循环语句

- For...in 语句

- For...in语句用于在对象的各个属性，或数组的各个元素之间循环
- For (variable in object)
- {statements;}
- 例:
- <html>

```
<head>
```

```
<script language = “javascript”>
```

```
color new Array(“红色”,“蓝色”,“绿色”);
```

```
for (var prop in color)
```

```
{   var record = “color”;
```

```
    record += prop + “=” + color[prop]+”<br>”
```

```
    document.write(record)
```

```
}
```

```
</script>
```

```
</head>
```

```
</html>
```

循环语句

- Do ... while
- Do...while循环语句用于运行语句块，直至条件变为false
 - Do
 - { statements; }
 - While (condition)
 - 其中 statements 是语句, condition 是条件

数组

- 数组用于存储具有相同数据类型的一组值，使用下标来区分各个值
- 在javascript中，数组的下标以零开始
- Javascript没有明确的数组数据类型，但却有内置的数组对象
- 创建数组
 - `arrayObjectName = new Array()`
 - 其中arrayObjectName 是对象名
给数组赋值
 - `arrayObjectName[0] = “Ryan Dias”`
 - `arrayObjectName[1] = “Graham Browne”`

数组

- 数组的属性
 - Length属性，用于返回数组中的元素个数，返回值为整型
- 使用方法
 - Sort 方法
 - Sort()方法用于对数组进行排序
 - Join()方法用于将数组元素合并为一个字符串，参数是分隔符
 - Reverse()方法用于反转数组元素：第一个数组元素变为最后一个，最后一个变为第一个

With 语句

- With 语句用于执行一组语句，所有这些语句都假定引用指定的对象
- 语法：
 - With (object)
 - { statements; }

例：

```
<html> <head>  
  <script language="javascript">  
    var a,b,c;      var r = 10;  
    with (Math)    {  
      a = PI*r*r;  
      b = r * cos(PI);  
      c = r * sin(PI2);  
    }  
  </script></head></html>
```

函数

- JavaScript 具有一些预定义的函数,可以用在脚本中
- Eval函数
 - Eval函数用于在不引用任何特定对象的情况下计算代码字符串
- isNaN函数
 - isNaN函数用于验证参数是否为数字,如果不是数字则返回true,若是数字则返回false
- parseInt 和 parseFloat函数
 - parseInt() 函数用于将字符串转换为整型
 - parseFloat() 函数用于将字符串转换为浮点数

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/018036141052006123>