

# JavaScript学习

---

- JavaScript概述
  - JavaScript介绍
    - JavaScript是一门脚本语言，是一门解释型语言，是一门弱类型语言是一门基于对象的语言是一门动态的语言
    - 组成
      - ECMAScript标准——基础语法和基本对象
      - DOM Document Object Model 文档对象模型：描述了处理网页内容的方法和接口
      - BOM Browser Object Model 浏览器对象模型
    - 用处
      - js最初的目的，解决用户和服务端之间的交互问题，现在js可以做特效，游戏，移动，服务器
  - 基本概念
    - 语法
      - 区分大小写
      - 标识符
      - 注释
      - 严格模式
      - 语句
    - 数据类型
      - typeof操作符
      - Undefined
      - Null
      - Number
      - String
      - Object
  - 应用场景
    - 浏览器：大部分公司的前端都在使用javascript来完成浏览器客户端的开发
    - 服务端：有些有大前端的公司开始使用node.js做web服务端开发
    - 其他：JavaScript还可以运行在App，微信小程序，桌面应用，浏览器插件开发，游戏VR，AR
  - 使用
    - JavaScript可以放置在HTML文档的头部
    - 可以放在外部文件中
      - 特点：分离HTML和代码
      - 使HTML和JavaScript更容易阅读和维护
      - 缓存的JavaScript文件可以加快页面加载速度
  - 输出

- 通过不同的方式来输出数据
- 使用document.write()方法将内容写到HTML文档中
- 使用window.alert()弹出警告
- 使用innerHTML写入到HTML元素
- 使用console.log ( ) 写入到浏览器的控制台

## • JavaScript的基本概念

### • 命名规则

- 由字母，数字，下划线，¥符号组成，不能以数字开头
- 不能是关键字和保留字
- 区分大小写
- 不能是算数和运算符

### • 规范

- 变量名必须有意义
- 遵守驼峰式命名法
- 尽量不要用\$作为变量名

### • JavaScript数据类型

- 字符串 ( string )
- 数组 ( number )
- 布尔 ( boolean )
- 空值 ( null )
- 未定义 ( undefined )

## • JavaScript函数

### • 函数定义

- JavaScript函数是用于执行特定任务代码块

### • 使用函数的有点

- 定义代码一次，并使用多次，使用不同的参数多次使用相同的代码，以产生不同的结果

### • 函数参数

- 函数定义时，参数没有指定数据类型
- 函数对隐藏参数没有进行检测
- 函数对隐藏参数的个数没有进行检测

### • 变量作用域

- 对象和函数同样也是变量
- 作用域为可访问变量，对象，函数的集合
- 函数作用域，作用域在函数内修改

### • 变量的生命周期

- JavaScript变量生命周期在他声明时初始化
- 局部变量在函数执行完毕后销毁
- 全局变量在页面关闭后销毁
- 函数参数只在函数内起作用，是局部变量

- JavaScript中的弹出框
- Alert警告框
  - 当弹出警告框时用户必须单击确定以继续，警报功能采用单个参数，即弹出框中显示的文本
  - alert ( "你好，这是一个警告框，必须点击确定" )
- Prompt提示框
  - 通常使用Prompt提示来让用户输入一个值
  - 当弹出提示框时，输入输入值，用户点击确定，该框返回一个输入值，若果单击取消，该框将返回null
  - prompt方法有两个参数
- Confirm确认框
  - 经常使用confirm让用户验证或者接受某些内容，当弹出确认框时，用户必须单击确定，或取消，继续
  - 若果用户单击取消，该框返回true若果用户单击取消，该框将返回false
- JavaScript中的对象
  - 对象的方法
    - 对象的方法是一个包含属性定义的属性
      - 对象名.方法名 ( )
    - 使用String对象的toUpperCase ( ) 方法来讲文本转换为大写
- JavaScript数组
  - Array，对象数组对象是使用单独的变量名来存储一系列的值
  - 数组可以用一个变量名存储所有的值，并且可以用变量名访问任何一个值
  - 数组中的每个元素都有自己的ID以便他可以很容易的被访问到
  - 数组方法
    - push ( ) 添加数组元素最简单的方法，可以使用push ( ) 方法在数组的末尾添加一个或者多个元素
    - pop ( ) 改变数组中数据的另一种方法
    - shift ( ) 移除第一个元素
    - unshift()方法。在数组的头部添加元素
    - concat ( ) 方法允许连接数组并创建一个全新的数组
    - forEach ( ) 方法用于调用数组的每个元素，并将元素传递给回调函数
    - indexOf ( ) 方法可以返回数组中某个指定元素位置，该方法从头到尾的检索数组，看他是否，看他是否含有对应的元素，返回第一次出现该元素的位置，若果数组中没有找到该元素则返回-1
    - slice ( ) 方法可以从已有的数组中返回选定的元素，可以提取字符串的某个部分，并以新的字符串返回被提取的部分
- JavaScript的核心对象
  - JSON
    - JSON是用于存储和传输数据的格式
    - JSON通常用于服务端向网页传递数据
    - JSON是一种轻量级的数据交互格式

- JSON的语法格式
- 数据为键值对
- 数据由逗号分隔
- 大括号保存对象
- 方括号保存数组

- DOM

- HTML DOM(文档对象模型)

- 当网页被加载时，浏览器，会创建页面的文档对象模型
    - 通过可编程的对象模型，JavaScript获得了足够的能力来创建动态的HTML
      - JavaScript能够改变页面中的所有HTML元素
      - 能够改变页面中的素有HTML属性
      - 能够改变页面中的所有CSS样式
      - 能够对页面中的所有事件作出反应

- 使用DOM

- DOM中的每个元素都有一组属性和方法，他们提供有关他们在DOM中的关系的信息
      - element.childNodes返回一个元素的子节点和数组
      - element.firstChild返回元素的第一个子节点
      - element.lastChild返回元素的最后一个子节点
      - element.hasChildNodes若果元素有任何的子节点，则返回true，否则返回false
      - element.previousSibling返回在同一树级别的上一个节点
      - element.parentNode返回元素的父节点

- 正则表达式

- 使用字符串的方法

- 在JavaScript中正则表达式通常用于两个字符串方法：search ( ) 和replace ( )
    - search ( ) 方法用于检索字符串中指定的子字符串，或者检索与正则表达式相匹配的子字符串，并返回子字符串的起始位置
    - replace ( ) 方法用于在子字符串中用一些字符替换另外一些字符，或者替换一个与正则表达式匹配的字符串的起始位置

- 使用RegExp对象

- 使用test ( )

- test方法是一个正则表达式的方法
      - test方法用于检测一个字符串是否匹配某个模式，若果字符串中含有匹配的文本则返回true。否则返回false

- 使用

- exec ( ) exec方法是一个正则表达式方法
      - exec方法用于检索一个字符串中的正则表达式的匹配，该方法返回一个数组，其中存放匹配的结果，若果找不到匹配，则返回值为null

以上内容整理于 [幕布](#)