JavaScript基础

学习目标

1、了解什么是JavaScript？

2、了解JavaScript的发展历史

3、了解JavaScript的组成部分

4、在HTML中使用JavaScript

什么是JavaScript？

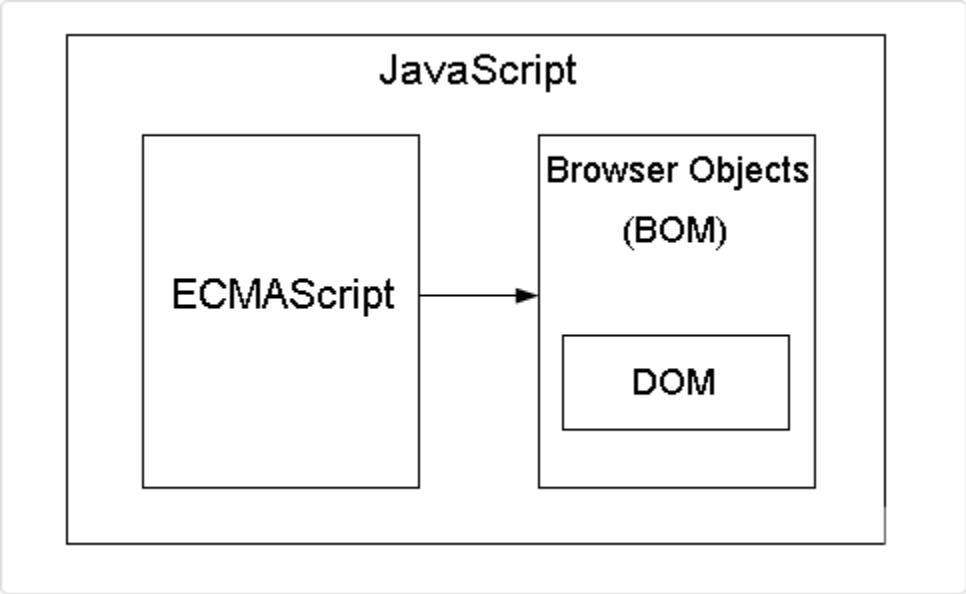
JavaScript是一种基于对象和事件驱动的客户端脚本语言，

最初的设计是为了检验HTML表单输入的正确性。

JavaScript的发展历史

JavaScript的组成

完整的JavaScript是由ECMAScript（语法）、Browser Objects（DOM、BOM）（特性）组成的。



在HTML中使用JavaScript

可以在head或body中使用<script>嵌入JavaScript脚本

JavaScript语法

学习目标

1、掌握JS的注释与分号

2、掌握JavaScript的语法

3、掌握标识符

4、掌握什么是变量

5、掌握变量的声明与赋值

JavaScript的注释与分号

// 单行注释

/**/ 多行注释

语句结束使用分号，如果省略，则由解析器确定语句的结尾。

JavaScript的语法

ECMAScript中的一切（变量、函数名和操作符）都区分大小写。

JavaScript的标识符

1、什么是标识符？

变量、函数、属性的名字，或者函数的参数。

2、标识符的命名规则：

- 由字母、数字、下划线（_）或美元符号（$）组 成
- 不能以数字开头
- 不能使用关键字、保留字等作为标识符。

什么是变量

ECMAScript的变量是松散类型

松散类型：可以用来保存任何类型的数据

换句话说，每个变量仅仅是一个用于保存值的占位符而已。

变量的声明与赋值

1、变量声明：

变量的声明要使用var操作符，

语法：var 变量名

2、变量赋值：

声明的同时赋值：var 变量名=值

先声明后赋值：　变量名=值

一次声明多个变量，用逗号隔开，如：
var id,sex,age,name=“marry”；

说明：

1、省略var声明的变量是全局变量

2、不推荐省略var操作符来定义全局变量

JavaScript数据类型

学习目标

1、掌握JavaScript的数据类型

2、掌握typeof操作符

3、掌握Undefined

4、掌握null

JavaScript的数据类型

ECMAScript中有5种简单数据类型（也称为基本数据类型）：Undefined、Null、Boolean、Number和String。

还有1种复杂数据类型：Object。

typeof

语法：typeof 变量 或 typeof(变量)

功能：检测变量类型

返回值：string类型，有可能是：

string、number、boolean、object、undefined、function

undefined

undefined类型只有一个值，即特殊的undefined。 说明：一般而言，不存在需要显式地把一个变量设置为undefined

值的情况。

null

1、null值表示一个空对象指针

2、如果定义的变量准备在将来用于保存对象，那么最好将变量初始化为null而不是其他值。

说明：undefined值是派生自null值的，所以undefined==null的返回结果是true。

JavaScript数据类型

学习目标

1、掌握Number

2、掌握isNaN()

3、掌握数值转换

1、Number()

2、parseInt()

3、parseFloat()

Number

Number：表示整数和浮点数

NaN：即非数值（Not a Number）是一个特殊的数值

说明：

1、任何涉及NaN的操作（例如NaN/10）都会返回NaN。

2、NaN与任何值都不相等，包括NaN本身。

isNaN()

语法：isNaN(n)

功能：检测n是否是“非数值”

返回值：boolean

参数：参数n可以是任何类型

说明：isNaN()在接收到一个值之后，会尝试将这个值转换为数值。

某些不是数值的值会直接转换为数值。

数值转换

有3个函数可以把非数值转换为数值：Number()、parseInt()和parseFloat()。其中Number()可以用于任何数据类型，而parseInt()和parseFloat()则专门用于把字符串转换成数值。

parseInt()

parseInt() ：会忽略字符串前面的空格，直至找到第一个非空格字符。

说明：

1、parseInt()：转换空字符串返回NaN。

2、parseInt()这个函数提供第二个参数：转换时使用的基数（即多少进制

parseFloat()

parseFloat：从第一个字符开始解析每个字符，直至遇见一个无效的浮点数字符为止

说明：

除了第一个小数点有效外，parseFloat()与parseInt()的第二个区别在于它始终都会忽略前导的零。

注：如果字符串中包含有效的十六进制格式，parseInt('0xf')将'0x'转换为相同大小的十进制数值

而parseFloat('0xf')只会输出0

JavaScript数据类型

学习目标

1、掌握String

2、掌握字符串转换

1、String()

2、toString()

3、掌握Boolean

4、掌握类型转换

String

String类型用于表示由零或多个16位Unicode字符组成的字符序列，即字符串。字符串可以由双引号（” ）或单引号（’ ）表示。

toString()与String()

语法：str.toString()

功能：将str转换为字符串

返回值：str的一个副本

参数：str是要转换的内容，可以是数值、布尔值、对象和字符串。

说明： 在不知道要转换的值是不是null或undefined的情况下，还可以使用String()函数，它能够将任何类型的值转换为字符串。

Boolean

用于表示真假的类型，即true表示真，false表示假

类型转换

1、除0之外的所有数字，转换为布尔型都为true

2、除”“之外的所有字符，转换为布尔型都为true

3、null和undefined转换为布尔型为false

JavaScript操作符

学习目标

1、掌握什么是表达式

2、掌握JavaScript操作符的分类

3、掌握算数操作符

什么是表达式

将同类型的数据（如常量、变量、函数等），用运算符号按一定的规则连接起来的、有意义的式子称为表达式。

操作符的分类

1、算数操作符
2、逻辑操作符
3、赋值操作符
4、比较操作符
5、三元操作符

算数操作符

+：加

-：减

*：乘

/：除

%：取余

递增和递减

1、递增

++a与a++都是对a进行递增的操作

区别：

++a先返回递增之后的a的值

a++先返回a的原值，再返回递增之后的值

2、递减同理

JavaScript操作符

学习目标

1、掌握赋值操作符

2、掌握比较操作符

3、掌握三元操作符

赋值操作符

简单赋值：=

复合赋值：+=、-=、*=、/=、%=

比较操作符

>、<、>=、<=、==、===、!=、!==

==：相等，只比较值是否相等

===：全等，比较值的同时比较数据类型是否相等

!=：不相等，比较值是否不相等

!==：不全等，比较值的同时比较数据类型是否不相等

返回值：boolean型

三元操作符

语法：

条件？执行代码1：执行代码2

说明：

可代替简单的if语句，

如果条件成立，执行代码1，否则执行代码2

JavaScript操作符

学习目标

1、掌握逻辑操作符的分类

2、掌握逻辑与

逻辑操作符

逻辑操作符：

&&：与

||：或

！：非

逻辑与

&& 与（只要有一个条件不成立，返回false）

说明：在有一个操作数不是布尔值的情况，逻辑与操作就不一定返回值，此时它遵循下列规则：

1、如果第一个操作数隐式类型转换后为true，则返回第二个操作数

2、如果第一个操作数隐式类型转换后为false，则返回第一个操作数

3、如果有一个操作数是null，则返回null

4、如果有一个操作数是NaN，则返回NaN

5、如果有一个操作数是undefined，则返回undefined

说明：
1、2在两个操作数情况下。
3、4、5在前面的操作数隐式类型转换后为ture 的情况时

JavaScript操作符

学习目标

1、掌握逻辑或

2、掌握逻辑非

逻辑或

|| 或（只要有一个条件成立，返回true）

说明：在有一个操作数不是布尔值的情况，逻辑与操作就不一定返回 值，此时它遵循下列规则：

1、如果第一个操作数隐式类型转换后为true，则返回第一个操作数

2、如果第一个操作数隐式类型转换后为false，则返回第二个操作数

3、如果两个操作数是null，则返回null

4、如果两个操作数是NaN，则返回NaN

5、如果两个操作数是undefined，则返回undefined

说明：规则是两个操作数的情况。

逻辑非

! 非

说明：

1、无论操作数是什么数据类型，逻辑非都会返回一个布尔值

2、 ! ! 同时使用两个逻辑非操作符时：

第一个逻辑非操作会基于无论什么操作数返回一个布尔值，

第二个逻辑非则对该布尔值求反。