

day1-javascript入门

笔记本： 凡尘：二阶段

创建时间： 2017/6/13 15:09

作者： 437389128@qq.com

更新时间： 2017/9/24 21:59

JS--基础

网页和网站的区别（扩展）

网页：是网站的一部分

网站：是多个网页的集合

JAVASCRIPT概括

1、js是怎么来的？

js创始人：Brendan Eich（布兰登·艾奇）



为什么推出javascript?

早期的网速比较慢，注册一个网站要等待很长时间，而且验证是放在服务器端的，如果用户部分信息未填写或者填写格式有误，则会刷新页面，用户需要重新填写所有信息。于是Brendan Eich 提出表单的普通验证应该由浏览器验证，于是花10天设计了JavaScript。

2、什么是javascript?

- a、由网景公司（Netscape）开发
- b、这是一个跨平台的脚本语言
- c、应用在网页中，用来操作浏览器及网页内容
- d、javascript最初的设计是为了检验HTML表单输入的正确性

什么是跨平台？

平台：window linux（谐音里纽克斯） unix（谐音邮尼克斯）.....

跨平台：在任何平台上都可以运行就叫做跨平台

什么是脚本语言？

脚本语言 VS 编程语言

javascript：脚本语言-----》标记语言和编程语言之间的一种语言

编程语言： 脚本语言 汇编语言 机器语言 高级语言。。。。。。

JS的三大组成（重）

网页的三大组成部分： 结构层 表现层 行为层----》三大部分

1、ECMAScript

ECMAScript:他是一种标准 一种规范 就当于造汽车一样必须有发动机，有轮胎。。。一些必要的东西

1997年06月第一套（ECMAScript）发布

2、BOM(browser) object Model

浏览器对象模型

用来获取或设置浏览器的信息；

3、DOM（document）

文档对象模型

js和H5的关系

JavaScript和Java什么关系？

CPU、内存、硬盘关系（扩展）

编写JS及如何运行JS

1、<div onclick="alert(1)"></div>

2、写在<script></script>

```
<script>
  alert(1)
</script>
```

3、写在js文件

```
<script src="js/index.js"></script>
```

JS注释

第一个程序的编写

什么是变量

所谓的变量，就是内存中的一块空间。

变量来源于数学，是计算机语言中能储存计算机结果或能表示值抽象概念。变量可以通过变量名访问。一般情况下变量通常是可变的

就像数学一样，javascript变量用于保存数值或者表达式

变量可以把程序中的每一段数据都赋值给一个简短、易记忆的名字

内存，即内部的存储器

硬盘，外部存储器

硬盘的数据，是可以永久保存的。

内存的数据，不是永久的，断电以后，清空。

硬盘读取数据的速度，是非常慢的。

计算机里所有的程序，在运行的时候，都是放在内存当中，而不是硬盘当中。

内存里这么多的数据，如何管理？？

谁来负责给新的数据分配空间？ 谁来负责将没有用的数据清理掉？？

操作系统

所谓的定义变量，即是在内存中开辟了一块空间，用来存储数据。

这块空间会有一个名字作为标记，叫做变量名

我们使用 = 这个符号，来给变量赋值

```
var username = "赵亚华";
```

```
var age = 30;
```

```
var school = "千锋";
```

```
document.write("你好，我叫"+username+"，年龄:"+age+"，学校："+school);
```

变量的命名规则

规则：数字、字母、下划线、\$符号，其中不能以数字开头

声明：不可以使用关键字和保留字

建议：见名知意

命名方法：驼峰命名法、常规命名(下连接)、匈牙利命名

驼峰命名法：classNo = 1609

常规命名： class_no = 1609

关键字和保留字

1、关键字

关键字：系统已经使用了（比如电话号码 110 120等）

2、保留字

保留字：现在没有使用，以后可能会使用

js中常见的数据类型

JavaScript 拥有动态类型

JavaScript 拥有动态类型。这意味着相同的变量可用作不同的类型：

```
var x  
var x = 6;  
var x = "Bill";
```

基本数据类型：

复杂数据类型：

扩展：

在内存中的存储方式：

1、基本数据类型把数据名和值直接存储在栈当中

2、复杂数据类型在栈中存储数据名和一个堆的地址，在堆中存储属性及值。访问时先从栈获取地址，在到堆中拿出相应的值

总结：基本数据类型复制的是值，复杂数据类型复制的是引用地址

为什么划分数据类型？

计算机为了更方便的对内存进行管理，对不同的数据，做了类型上的划分。

如何查看一个变量的数据类型？？

使用**typeof**命令

例如：

var x = 5;

var res = **typeof** x;

请问变量res的内容是什么？

number ----> 整数、浮点数

string 字符串

boolean 布尔类型

undefined 对于JS来讲，一个变量如果没有赋值，不能确定类型，类型就是undefined

object 对象类型

function 函数类型

null的类型也是object

使用一个没有被var定义过的变量，会报错！

javascript转换成false的值有哪些？

null和undefined的区别

1、都是用于描述变量没有值

2、undefined表示变量定义了但没有赋值

null表示变量的值为空（一般给对象赋初值）

javascript中的运算符

算术运算符：

赋值运算符：

关系运算符：

逻辑运算符：

自增自减运算符：

+号的大坑

var result = '你' + '好' ;

var result = 1+'1' ;

var result = '1'+true;

```
var result = '1'+undefined;
```

```
var result = '1'+null
```

任何与字符串相加的内容，都会与字符串进行连接

- 1、进行运算（+号前后都是数字）
- 2、如果+号前后有一个不是数字，那么进行拼接（连接）
- 3、任何与字符串相加的内容，都会变成字符串

```
var num1 = 10,
    num2 = 20,
    str = '15';

document.write('<br>');
document.write(num1 + num2 + str); //3015

document.write('<br>');
document.write(num1 + str + num2); //101520

document.write('<br>');
document.write(str + num1 + num2); //151020

document.write('<br>');
document.write(num1 + (num2 + str) ); //102015

document.write('<br>');
document.write(str + true); //151020
```

隐式类型转换&&强制类型转换

类型转换：当有两个不同类型的变量进行运算时，即有可能出现类型转换
哪些符号可以帮我们做类型转换呢？

常见的隐式类型转换：

常见的强制类型转换：

- 1 Number()函数 只针对 纯数字内容的字符串转换有效
- 2 parseInt()函数 可以忽略数字后面的字符
- 3 parseFloat()函数 可以带小数点的

NaN

- 1、什么是NaN
 - a、Not a Number 不是一个数字
 - b、NaN仅代表不是数字，自身和自身都不相等（NaN != NaN）

- 2、怎么判断是不是NaN？

isNaN(a) 判断a是不是NaN
如果 a 是 NaN，返回true，否则返回false

isNaN()不靠谱，建议使用Number.isNaN()

- 3、NaN属于什么数据类型？
数字类型 number

练习二

- 为抵抗洪水，战士连续作战89小时，编程计算共多少天零多少小时？

```
var time = 89;
document.write('共'+parseInt(time/24)+'天'+time%24)
```

- 小明要到美国旅游，可是那里的温度是以华氏度为单位记录的。它需要一个程序将华氏温度（80度）转换为摄氏度，并以华氏度和摄氏度为单位分别显示该温度。

提示：摄氏度与华氏度的转换公式为：摄氏度 = 5/9.0*(华氏度-32)

```
var a = 80;
alert(parseInt(5/9*(a-32)))
```

关系运算符： > < >= <= == !=

=== 绝对相等

!= 绝对不相等

```
var a = 100;

var b = "100";

//alert(a == b); //将左右两边的变量做类型转换，然后再比较大小
alert(a === b); //不会对左右两边做类型转换
```

逻辑运算符 && || !

```
var res = 1==1 && 3>5; //false;
var res = 1==1 || 3>5; //true;
var res = !(3>5); //true
```

自增自减：

```
//alert(a++); // 1 先使用，再加1
//alert(++a); // 3 先加1，再使用
var a = 3;
a+ a++ + ++a+ a-- + --a+ a++
```

补充

- 1、可以通过id名直接获取某个元素
- 2、更改元素的样式 元素.style.样式名 = 样式值
- 3、获取文本框的值 var a = 文本框.value 给文本框赋值 文本框.value = 10;
- 4、给元素添加点击事件


```
元素.onclick = function(){
    //点击事件时做的事情
}
```

练习

- 1、假设在公司写1行代码可以得到1元钱，1小时自己可以写80行代码，公司规定迟到一次扣除50元，每天上班7个小时，上个月自己迟到了2次，上个月共发工资12100元，问上个月自己工作多少天？
(百度 向上取整)
- 2、实现两个文本框的和
- 3、点击按钮字体变大

预习

- 0、逻辑运算
- 1、if(){else{}
- 2、switch(){ break
- 3、警告框 alert()
- 4、判断任意数字是不是偶数
比如： var a = 19; 弹出a不是偶数