# JavaScript简介及变量

是什么 为什么 怎么玩

## Javascript简介

### 1、是什么

#### 概念

1995年 美国 布莱登·艾奇 在网景公司 用了10天写的一门语言；

Javascript是一门：动态的 弱类型的 解释型 脚本语言（面向对象的语言）

动态：程序执行的时候才会确定数据类型

弱类型：数据类型不固定，可以随时改变；

解释型：相对编译型的来说；

编译型语言：程序在运行之前需要整体先编译（翻译）

解释型语言：程序运行的时候，不会编译，拿一行解释执行一行；

脚本：一般都是可以嵌在其它编程语言当中执行；

#### Java和JavaScript的区别(了解)

Java和Javascript没有任何关系；各自是各自

起初livescript 后来借着 sun公司的Java火起来；

### 2、为什么要有JavaScript

#### 开始能干啥

做表单验证

#### 现在能干啥

表单验证

用户交互

游戏

后端开发

html负责结构

css 负责样式

js 负责交互

### 3、怎么做

#### JavaScript的组成部分

ECMAscript 负责js的语法部分 （ES5 后期去学ES67）

DOM document object model 文档对象模型 （操作元素）

BOM browser object model 浏览器对象模型 （操作浏览器）

#### js的体验，第一个js程序，js的语句特点

alert函数 以弹出框的形式输出内容

console.log函数 以日志的形式在控制台输出内容

document.write函数 把内容作为网页的内容进行展示

#### js代码的特点

js语言执行的过程是从上到下，一行一行进行执行

js语句的特点，每一行我们都会有一个分号进行结尾 ，分号不是必须的,写不写都行，

//单行注释

/\* 多行注释 \*/

#### js的三种书写位置及优缺点

行内js 局限性很大，只能针对事件进行添加， 用的很少；它的代码分离性最差，可读性不强；

内嵌js 在body的最下面 script标签内去写我们的js代码，教学和项目用的最多，前期

外链js 在外部js文件当中去写js代码，最后通过script标签src引入到html当中，项目最终都会把文件进行分离；

注意：在外链的script标签之间不能去写代码

## 变量及作用

### 1、是什么

变量：可以变化的量

常量：不可以改变的量

### 2、为什么要有变量

什么是程序？什么是进程？

程序：代码的集合，一般指的是文件，程序是静态的

进程：运行的程序，就被称作进程，进程是动态的

程序其实就是一个输入 运算 输出的过程，那么我们要对一个数据进行运算，然后进行输出，这个数据就需要有一个东西去保存，这个东西，就是我们所说的变量，本质上是一个容器（内存空间）；

cpu（寄存器） 最快

内存 断电消失 第二快

硬盘 永久存储 最后

### 3、怎么去玩

#### 定义变量

1）定义变量赋值

2）定义变量初始化

3）不带var的变量 不带var的变量 必须赋值否则报错；

4）内存的结构图（字节、位、地址）

var a = 10; 本质就是在内存当中开辟一块内存，起名字叫a,然后把10存储到a这块内存当中；

#### 变量的命名规范（组成，大驼峰，小驼峰，下划线）

1）标识符：

数字、字母、下划线和$符组成；

不能以数字开头

不能和关键字及保留字同名；

2）起名字的约定成俗：

大驼峰 var PersonName; 每个单词首字母都大写，大驼峰在前端当中用的不多，只是在特定的情况下才用；

小驼峰 var personName; 第一个单词首字母不会大写，其余的首字母都大写；这是我们前端主打用法；

下划线 var person\_name; 一般在后端用的比较多，前端用的不是很多；

变量名字：只能用英文 不许拼音~ 汉语；

3）关键字和保留字

javascript的关键字

break case catch continue default delete do

else finally for function if in instanceof

new return switch this throw try typeof

var void while with

javascript的保留字

abstract boolean byte char class const debugger double enum export extends final float goto implements import int interface long native package private protected public short static super synchronized throws transient volatile

4）小案例：

两种方式交换两个变量的值

## 数据类型

### 1、基本数据类型和对象(复杂，引用)数据类型

#### 基本数据类型

数字Number 字符串String 布尔Boolean undefined null

#### 对象数据类型（对象类型，后续）

### 2、基本数据使用场景

数值类型 整数 小数 科学记数法 2进制(0b) 8进制(0) 16进制(0x)

字符串 单引号或者双引号包含 空字符串 和 空白字符串 引号嵌套

布尔 true或者false

undefined 定义的变量没有赋值

null 定义的变量赋值为null(一般是对对象进行初始化使用或者是删除一个对象的时候用到)

### 3、数据类型的判断

#### typeof判断

typeof 是用来对数据类型做判断的；结果是数据类型名的小写字符串形式；

#### typeof的结果是什么类型

字符串类型

typeof 基本数据 整体的值都是字符串；

'number'

'string'

'boolean'

'undefined'

'object'

#### typeof的判断基本数据类型的特殊情况

null最终会判断为object（设计缺陷）

因为null只在对象初始化和删除用到所以类型归为object；

## 运算符和表达式

### 1、什么是运算符 什么是表达式\*\*\*\*\*\*\*\*\*\*\*

运算符：就是参与运算的符号；

表达式：由变量或者常量与运算符组成的式子；表达式是有值的；

var a = 10;

var b = 20;

a = a + b //表达式，表达式是有值得

### 2、运算符和表达式分类

#### 1）算术运算符和算术表达式

###### + - \* / %

%:求余或者取模 /：求商 %拿余数 5/2 = 2.5 5%2 = 1

%:作用很大，后期用的比较多：

1、判断一个数能否被另一个数整除

2、可以求出一个数字每个位上的数字

3、可以求一个范围内的数

//任何的一个数字对5取余，肯定落在0 - 4之间

###### ++ -- //自增自减运算符； \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

a 和 a++ 是两个东西 a代表变量 a++代表表达式 他们两个都是有值的；

++在后，先赋值（先把a这个变量的值赋值给表达式整体a++）后++ 然后偷摸让a变量内部的值加1；

++在前，先让a+1 再把a的值赋值给++a这个表达式；

###### 案例：

var a = 2;

var b = 3;

var c = a + b++;// a 2 b++ 3 c 5 b 4

c= ++a + b++; // ++a 3 b++ 4 c 7 a 3 b 5

c = a + b--;// a 3 b-- 5 c 8 b 4

a 3 b 4 c 8

作业：

运算符： 计算前++和后++的表达式的值

var a = 7;

var b = 9;

var c = a + b++;

c = ++a + b++;

c = b + a++;

c = ++a + b++;

c = --a + b;

c = a + b--;

问：c的值；