什么是JavaScript：

JavaScript是网景公司开发的一种基于客户端浏览器、基于面向对象、事件驱动式的网页脚本语言。

JavaScript语言的前身叫作LiveScript，于1995年11月末于Sun公司联合宣布把其名称改为JavaScript；97年JavaScript1.1版本被ECMA（欧洲计算机制造商）标准化为ECMAScript的全新脚本语言

JavaScript是属于web的语言，它使用于PC、笔记本电脑、平板电脑和移动电话

JavaScript与ECMAScript的关系：

ECMAScript是JavaScript的标准和基础

JavaScript与JAVA的关系

雷锋和雷峰塔的关系

JavaScript和Java都与sun公司有合作，是借势Java

JavaScript特点

JavaScript是一种解释性脚本编写语言

JavaScript是基于对象的脚本语言

简单性

安全性

动态性

跨平台性

JavaScript作用：减轻服务器端压力

交互式操作、表单验证、网页特效、web游戏、服务器脚本开发

JavaScript组成：

ECMAScript（核心）、DOM（文档对象模型）、BOM（浏览器对象模型）

什么是ECMAScript：

ECMAScript定义了脚本语言的所有属性、方法和对象；包括语法、类型、关键字、保留字、运算符、对象等

DOM：把整个页面规划为层级式的节点结构

BOM：对浏览器窗口进行访问和操作

JavaScript的嵌入方式

使用JavaScript:前缀构建执行JavaScript代码

使用<script ../>标签来包含JavaScript代码

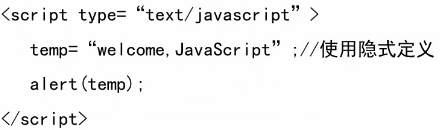
导入外部的JavaScript文件

<script src="hello.js" type=”text/javascript”></script>

变量的声明：

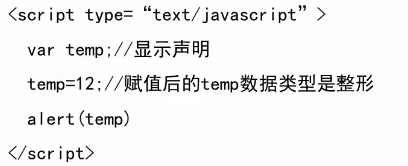
JavaScript是弱类型脚本语言，使用变量之前，可以无需定义，当使用某个变量的时候直接使用即可。主要分为以下两种定义方式：

隐式定义：直接给变量赋值



注意：在使用隐式定义的变量之前，要做好初始化工作，不然会报错；隐式定义的temp相当于window.temp= “welcome,JavaScript”; （不推荐）

显式定义：使用var关键字定义变量



可以同时定义多个变量

var num =10 ,str = "JavaScript";

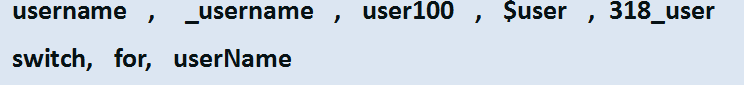
变量的名称需要遵循以下规则：

1、首字符必须是字母（大小写均可）、下划线（\_）、或者是美元符号（$）

2、余下的字符可以是下划线、美元符号、字母或者是数字字符

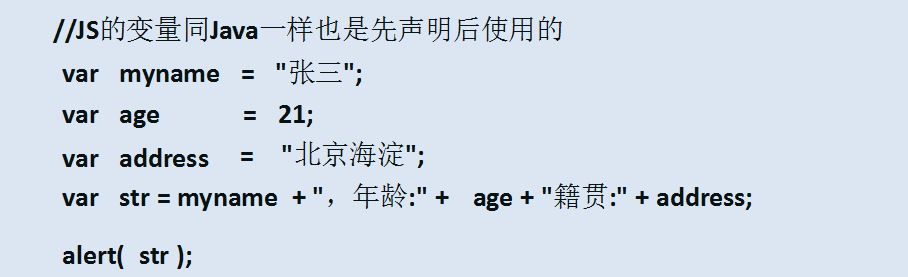
3、变量名不能使用关键字

查看一下变量的声明是否合法：



注意：变量名称区分大小写

变量的使用：



建议先定义后使用

+ 运算符用于把文本值或字符串变量加起来（连接起

来），如果是两个数字则为数学运算加号

= 运算符用于给变量赋值，变量可以重复赋不同类型

的值

JavaScript中的注释：

单行注释以 // 开始，以行末结束，例如：

alert("恭喜你！注册会员成功"); //在页同上弹出注册会员成功的提示框

多行注释以 /\* 开始，以 \*/ 结束，符号 /\*…… \*/ 指示中间的语句是该程序中的注释。例如：

/\*

使用for循环运行“document.write("<h3>Hello World</h3>");”5次

使用document.write在页面上输出“Hello World”

\*/

数据类型：

JavaScript是弱类型脚本语言，声明变量时无需指定变量的数据类型。

JavaScript常用的数据类型如下：

数值类型（number）：包含整数和浮点数

布尔类型（boolean）：只有true或false两个值

字符串类型（string）：字符串必须使用双引号或单引号括起来

未定义类型（undefined）：专门用来确定一个已经创建但是没有初始值的变量

对象类型（object）：JavaScript中的对象、数组、null等

函数对象类型（function）：JavaScript的函数

注：可以通过typeof来检测变量的数据类型

JavaScript和其他编程语言一样有类型转换的能力

1、通过“+”号或toString（）方法将数值转换成字符串

2、通过parseInt（）将字符串转换成整形

3、通过parseFloat（）将字符串转换成浮点型

注：NaN代表不是一个数字

运算符号：

|  |  |
| --- | --- |
| **类型** | **运算符** |
| 算术运算符 | +    -   \*    /    %    ++    -- |
| 赋值运算符 | =   +=   -=  \*=  /=  %= |
| 比较运算符 | ==、===、!=、!==、>、<、>=、<= |
| 逻辑运算符 | &&      ||     ! |

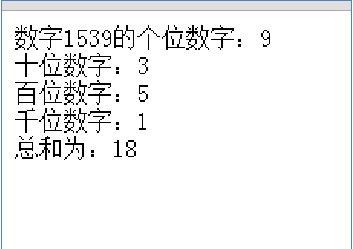
+运算符在变量都为数值类型（number）时，做加法运算；

如果两个变量中，有一个变量时字符串类型（string），则+运算符做连接符号

++在变量之前，先++，再把加出来的结果跟后面的变量做运算

++在变量之后，先跟后面的变量做运算，再++

**练习**——现有一个数字为 1539，使用”/”与“%”运算符获取该数字各个位上的数字；将各个位上的数字求和



常用的输入/输出：

alert()

alert(“提示信息”);

prompt()

prompt(“提示信息”, “输入框的默认信息”);

prompt(“请输入姓名”, “张三”);

prompt(“请输入姓名”);

JavaScript支持的分支语句主要有if和swicth语句：

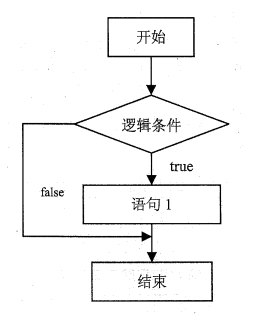
if语句：单分支选择结构语句，只有当指定条件为 true 时，使用该语句来执行代码

if (条件)

{

只有当条件为 true 时执行的代码

}



if...else语句：双向分支选择结构语句，if...else当条件为 true 时执行代码，当条件为 false 时执行其他代码

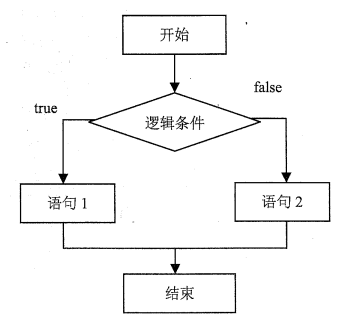
if(条件){

//执行代码块

}else{

//执行代码块

}



if...else if...else语句：多向分支选择结构语句

if(条件){

//执行代码块

}else if(条件){

//执行代码块

}else{

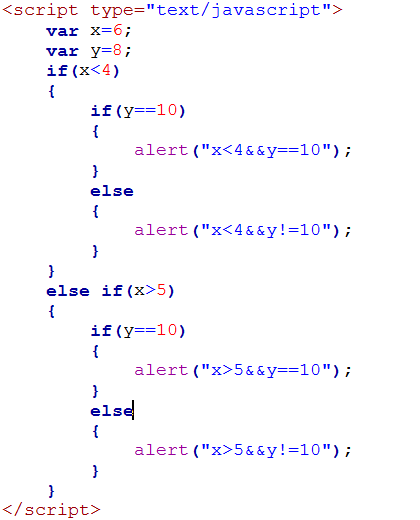
//执行代码块

}

**示例**——使用if...else语句制作一周的工作情况：输入一个周几求当天做什么；周一“学语文”，周二“学数学”，周三“学英语”，周四“学物理”，

周五“学化学”，周日“休息”

if...else语句嵌套示例：



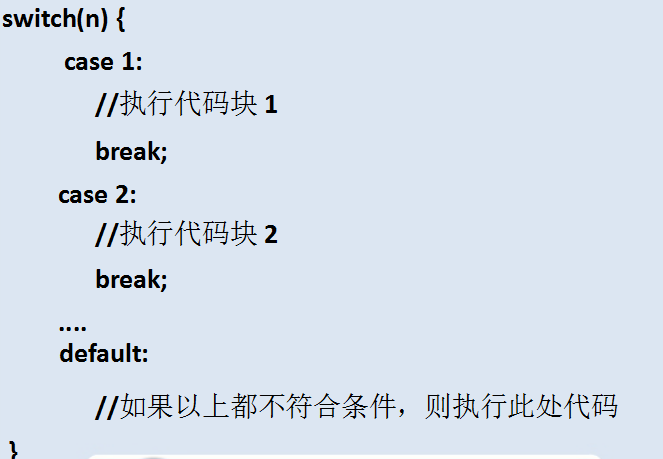
**练习**——使用if...else语句判断输入分数的等级：90-100：优秀，70-89：良好，60-69：及格，0-59：不及格

条件运算符：variablename = ( condition ) ? value1 : value2 ;

例如：



switch多分支语句：



**练习**——使用switch...case语句判断输入分数的等级：90-100：优秀，70-89：良好，60-69：及格，0-59：不及格

if...else if...else语句与switch case语句的比较：

范围：前者可以比较定值也可以比较范围

后者只能比较定值

效率：前者效率低（每一个表达式都要求值对比）

后者效率高（表达式只需要跟case中的一个匹配就可以）

**练习**——编写一个四则运算（+、-、\*、/）

需求说明

实现使用prompt()输入两个数和运算符号(+、-、\*、/)，并计算两个数的操作结果

提示：

通过prompt ()来接受数据和处理方式

通过alert ()来显示处理