注意：

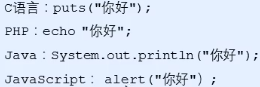
1. 红色为知识大部分。
2. 【】为知识分块。
3. ⚪黑点为【】内知识点。
4. 数字标号为⚪黑点分点。
5. 黄色为显眼重点。
6. 蓝色语法截图为重点语法。
7. 灰色表格截图为语法详解。
8. 黑色背景为编程示例。
9. 具体案例在相关html文件中。

***基础部分***

【基础】

* 编程语言：主动，有逻辑能力和行为能力，如if else，while，for等指令。

标记语言（html）：被动，被读取，不向计算机发送指令，常用于格式化和链接。

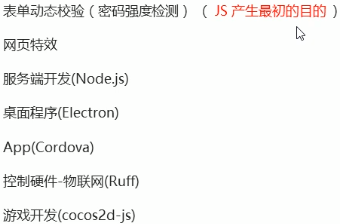


* JavaScript是运行在客户端的脚本语言。

脚本语言：不需编译，运行时由js解释器（js引擎）逐行解释和执行。

现在可以基于node.js技术进行服务器端编程。

* js作用：业务逻辑、页面控制



* 浏览器组成：

1. 渲染引擎：解析html和css，俗称内核，如chrome的blink、webkit。
2. js引擎（js解释器）：读取网页的JavaScript代码处理后运行（浏览器本身不执行JavaScript代码），如chrome的v8。

* js组成：

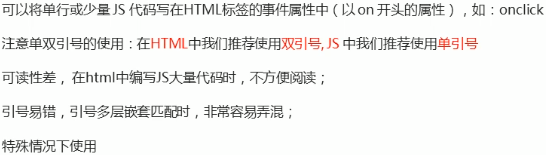
1. ECMAscript：JavaScript语法



1. dom（document object model）：页面文档对象模型。处理可扩展标记语言的标准编程接口，通过dom提供的接口可对页面上元素进行操作（大小、位置、颜色等）。
2. bom（browser object model）：浏览器对象模型。提供独立于内容、可与浏览器窗口进行互动的对象结构，通过bom可操作浏览器窗口（弹出框、控制浏览器跳转、获取分辨率等）。

* js书写位置

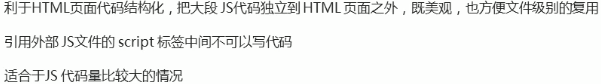
1. 行内



1. 内嵌



1. 外部



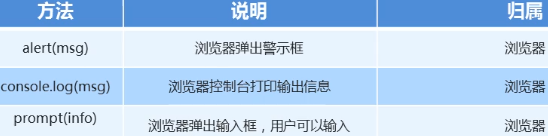
* js注释

单行：// 快捷键ctrl+/

多行：/\*\*/ 快捷键ctrl+alt+a

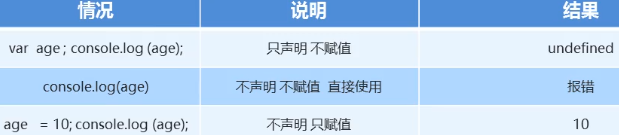
修改快捷键…

* js输入输出语句



* 变量声明var





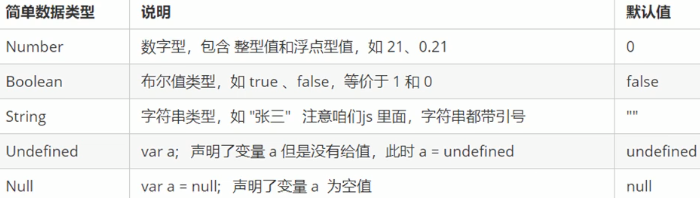
命名规范;

1. 变量名由字母、数字、下划线、$组成；
2. 不能以数字开头；
3. 驼峰命名法：首字母小写，后面单词首字母大写。

* js数据类型

JavaScript是一种弱类型/动态语言，不用提前声明变量类型，运行时由js引擎根据=右侧确定数据类型，同一变量可用于不同数据类型。

1. 简单数据类型



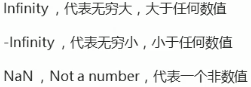
1. 数字型number：整数、小数（浮点数）

二进制、八进制、十进制、十六进制

js中数值的最大值和最小值：



数字型三个特殊值Infinity、-Infinity、NaN：



IsNaN(x)：x为数字返回false，x不是数字返回true。

1. 字符串型string
2. 复杂数据类型

* 25