# HTML

## （1） 是什么

### 参1：<https://www.runoob.com/html/html-tutorial.html>

超文本标记语言（英语：HyperText Markup Language，简称：HTML）是一种用于创建网页的标准标记语言。

您可以使用 HTML 来建立自己的 WEB 站点，HTML 运行在浏览器上，由浏览器来解析。

### 参2：<https://www.runoob.com/html/html-intro.html>

HTML 是用来描述网页的一种语言：

* HTML 指的是超文本标记语言: **H**yper**T**ext **M**arkup **L**anguage
* HTML 不是一种编程语言，而是一种**标记**语言
* 标记语言是一套**标记标签** (markup tag)
* HTML 使用标记标签来**描述**网页
* HTML 文档包含了HTML**标签**及**文本**内容
* HTML文档也叫做**web 页面**

## （2） 入门示例

<https://www.runoob.com/html/html-intro.html>

### 代码

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="utf-8">  <title>菜鸟教程(runoob.com)</title>  </head>  <body>    <h1>我的第一个标题</h1>    <p>我的第一个段落。</p>    </body>  </html> |

### 效果

Chrome：

图形用户界面, 文本, 应用程序

描述已自动生成

### 解析

图示

描述已自动生成

* **<!DOCTYPE html>** 声明为 HTML5 文档
* **<html>** 元素是 HTML 页面的根元素
* **<head>** 元素包含了文档的元（meta）数据，如 **<meta charset="utf-8">** 定义网页编码格式为 **utf-8**。
* **<title>** 元素描述了文档的标题
* **<body>** 元素包含了可见的页面内容
* **<h1>** 元素定义一个大标题
* **<p>** 元素定义一个段落

**注：**在浏览器的页面上使用键盘上的 F12 按键开启调试模式，就可以看到组成标签。

图形用户界面, 应用程序

描述已自动生成

## （x） HTML网页结构

<https://www.runoob.com/html/html-intro.html>

### 【1】 是什么

下面是一个可视化的HTML页面结构：

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

只有 <body> 区域 (白色部分) 才会在浏览器中显示。

### 【2】 示例

见（2）

图形用户界面, 应用程序

描述已自动生成

## （x） Web浏览器

<https://www.runoob.com/html/html-intro.html>

Web浏览器（如谷歌浏览器，Internet Explorer，Firefox，Safari）是用于读取HTML文件，并将其作为网页显示。

浏览器并不是直接显示的HTML标签，但可以使用标签来决定如何展现HTML页面的内容给用户：

如（2）中示例：chrome上

图形用户界面, 文本, 应用程序

描述已自动生成

## （x） HTML版本

<https://www.runoob.com/html/html-intro.html>

从初期的网络诞生后，已经出现了许多HTML版本:

|  |  |
| --- | --- |
| **版本** | **发布时间** |
| HTML | 1991 |
| HTML+ | 1993 |
| HTML 2.0 | 1995 |
| HTML 3.2 | 1997 |
| HTML 4.01 | 1999 |
| XHTML 1.0 | 2000 |
| HTML5 | 2012 |
| XHTML5 | 2013 |

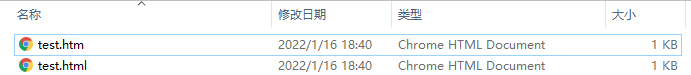
## （x） HTML文档的后缀名

<https://www.runoob.com/html/html-tutorial.html>

* .html
* .htm

以上两种后缀名没有区别，都可以使用。

示例：



2个均可显示

## （x） 语法

### 【1】 HTML标签

<https://www.runoob.com/html/html-intro.html>

HTML 标记标签通常被称为 **HTML 标签** (HTML tag)。

* HTML 标签是由尖括号包围的关键词，比如 <html>
* HTML 标签通常是成对出现的，比如 <b> 和 </b>
* 标签对中的第一个标签是开始标签，第二个标签是结束标签
* 开始和结束标签也被称为开放标签和闭合标签

标签示例：

<标签>内容</标签>

### 【2】 HTML元素

<https://www.runoob.com/html/html-intro.html>

"HTML 标签" 和 "HTML 元素" 通常都是描述同样的意思.

但是严格来讲, 一个 HTML 元素包含了开始标签与结束标签。

元素示例:

<p>这是一个段落。</p>

### 【3】 <!DOCTYPE>声明

<https://www.runoob.com/html/html-intro.html>

#### 《1》 是什么

<!DOCTYPE>声明有助于浏览器中正确显示网页。

网络上有很多不同的文件，如果能够正确声明HTML的版本，浏览器就能正确显示网页内容。

doctype 声明是不区分大小写的，以下方式均可：

|  |
| --- |
| <!DOCTYPE html>  <!DOCTYPE HTML>  <!doctype html>  <!Doctype Html> |

#### 《2》 通用声明

##### HTML5

<!DOCTYPE html>

##### HTML 4.01

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">

##### XHTML 1.0

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

查看完整网页声明类型 [DOCTYPE 参考手册](https://www.runoob.com/tags/tag-doctype.html)。

### 编码

目前在大部分浏览器中，直接输出中文会出现中文乱码的情况，这时候我们就需要在头部将字符声明为 UTF-8 或 GBK。

示例：

|  |
| --- |
| <!DOCTYPE html>  <html>  <head>  <meta charset="UTF-8">  <title>  页面标题</title>  </head>  <body>    <h1>我的第一个标题</h1>    <p>我的第一个段落。</p>    </body>  </html> |

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

## （x） h5

### 【1】 是什么

<https://www.zhihu.com/question/30363342>

#### 回答1：文本 中度可信度描述已自动生成

注意注意： **H5 != HTML5**

*H5 是一个产品名词  
HTML5 是一个技术名词*

所以，H5指的不是HTML5，而是**某种在微信等移动端看上去酷炫能够提升公司格调顺便亮瞎访问者氪金狗眼顿升膜拜之心就算没有内容也能被广泛转发分享的一种东西**。

#### 回答2：图示 中度可信度描述已自动生成

H5 是中国人制造的一个专有名词，是一种用来替代 Adobe Flash，实现与之相同或相似网页特效的产品。

文本, 信件

描述已自动生成

#### 回答3：文本 低可信度描述已自动生成

我目前为止的理解：

「H5」是指：在iPhone这类无法播放Flash的移动端上呈现的，可以达到Flash效果（如各种动画，互动）的，用于广告、营销的，具有酷炫效果的网页。可以简略称为「[移动端PP](https://www.zhihu.com/search?q=%E7%A7%BB%E5%8A%A8%E7%AB%AFPP&search_source=Entity&hybrid_search_source=Entity&hybrid_search_extra=%7B%22sourceType%22%3A%22answer%22%2C%22sourceId%22%3A146323760%7D)T」。

所以根据以上定义，「H5」和「HTML5」没有任何直接的关系。

**所以做H5可能用到HTML5的标准，也可能完全没有用到**。

# CSS

## （1） 是什么

<https://www.runoob.com/css/css-tutorial.html>

***CSS****(Cascading Style Sheets，层叠样式表），是一种用来为结构化文档（如 HTML 文档或 XML 应用）添加样式（字体、间距和颜色等）的计算机语言，****CSS****文件扩展名为****.css****。*

*通过使用****CSS****我们可以大大提升网页开发的工作效率！*

*在我们的****CSS****教程中，您会学到如何使用 CSS 同时控制多重网页的样式和布局。*

***CSS3****现在已被大部分现代浏览器支持，而下一版的****CSS4****仍在开发中。*

## （x） 写法

### 【1】 在html中

<https://blog.csdn.net/weixin_56226042/article/details/120854793>

<https://www.cnblogs.com/64Byte/p/12497936.html>

<https://blog.csdn.net/qq_16255321/article/details/54706914>

<https://cloud.tencent.com/developer/article/1654547>

共4种写法。

#### 内嵌式

##### x） 示例

<https://www.runoob.com/css/css-tutorial.html>

图形用户界面, 文本, 应用程序, 聊天或短信

描述已自动生成

对应的完整html代码：

<https://www.runoob.com/try/try.php?filename=trycss_default>

#### 外链式

##### x） 示例

<https://www.runoob.com/try/demo_source/demo_default.htm>

### 【2】 在xml中

## （x） 语法

<https://www.runoob.com/css/css-syntax.html>

### 【x】 伪类

<https://www.runoob.com/css/css-pseudo-classes.html>

# JavaScript<JS>

## （1） 是什么

参考：

1. <https://www.w3schools.com/js/default.asp>
2. <https://developer.mozilla.org/en-US/docs/learn/JavaScript>

JavaScript is the programming language of the Web.

JavaScript is one of the **3 languages** all web developers **must** learn:

   1. [**HTML**](https://www.w3schools.com/html/default.asp) to define the content of web pages

   2. [**CSS**](https://www.w3schools.com/css/default.asp) to specify the layout of web pages

   3. **JavaScript** to program the behavior of web pages

[JavaScript](https://developer.mozilla.org/en-US/docs/Glossary/JavaScript) is a programming language that allows you to implement complex things on web pages. Every time a web page does more than just sit there and display static information for you to look at—displaying timely content updates, interactive maps, animated 2D/3D graphics, scrolling video jukeboxes, or more—you can bet that JavaScript is probably involved.

It is the third layer of the layer cake of standard web technologies, two of which ([HTML](https://developer.mozilla.org/en-US/docs/Learn/HTML) and [CSS](https://developer.mozilla.org/en-US/docs/Learn/CSS)) we have covered in much more detail in other parts of the Learning Area.

图片包含 图表

描述已自动生成

* [HTML](https://developer.mozilla.org/en-US/docs/Glossary/HTML) is the markup language that we use to structure and give meaning to our web content, for example defining paragraphs, headings, and data tables, or embedding images and videos in the page.
* [CSS](https://developer.mozilla.org/en-US/docs/Glossary/CSS) is a language of style rules that we use to apply styling to our HTML content, for example setting background colors and fonts, and laying out our content in multiple columns.
* [JavaScript](https://developer.mozilla.org/en-US/docs/Glossary/JavaScript) is a scripting language that enables you to create dynamically updating content, control multimedia, animate images, and pretty much everything else. (Okay, not everything, but it is amazing what you can achieve with a few lines of JavaScript code.)

翻译（

* [HTML](https://developer.mozilla.org/zh-CN/docs/Glossary/HTML)是一种标记语言，用来结构化我们的网页内容并赋予内容含义，例如定义段落、标题和数据表，或在页面中嵌入图片和视频。
* [CSS](https://developer.mozilla.org/zh-CN/docs/Glossary/CSS) 是一种样式规则语言，可将样式应用于 HTML 内容， 例如设置背景颜色和字体，在多个列中布局内容。
* [JavaScript](https://developer.mozilla.org/zh-CN/docs/Glossary/JavaScript) 是一种脚本语言，可以用来创建动态更新的内容，控制多媒体，制作图像动画，还有很多。（好吧，虽然它不是万能的，但可以通过简短的代码来实现神奇的功能。）

）

## （x） 运行环境

<https://www.cnblogs.com/wilburn/p/13444630.html>

js 脚本的运行环境有浏览器环境和 Node.js 两种。

### 【1】 浏览器环境

#### 示例：

<https://developer.mozilla.org/zh-CN/docs/Learn/JavaScript/First_steps/What_is_JavaScript>



### 【2】 Node.js环境

见下面图片包含 徽标

描述已自动生成

## （x） 语法

### export default

<https://zhuanlan.zhihu.com/p/389813789>

<https://kimi.moonshot.cn/chat/cpc69rmaoforvtkrv70g>

<https://developer.huawei.com/consumer/cn/forum/topic/0203133560842675006>

### 数组的push操作

<https://kimi.moonshot.cn/chat/cpc7pi8nsmmu8s08vnog>



### 省略函数实参

<https://www.cnblogs.com/unclekeith/p/5792485.html>

<https://blog.csdn.net/neweastsun/article/details/104108497>

### Promise接口

<https://kimi.moonshot.cn/chat/cpq38upmqu0ec29r8ing>

### ?? 和 ||

<https://deepinout.com/typescript/typescript-questions/22_typescript_double_double_questionmarks_in_typescript.html>

<https://developer.mozilla.org/zh-CN/docs/Web/JavaScript/Reference/Operators/Nullish_coalescing>

## （x） 不同平台执行js

### 【1】 chrome上执行js

<https://www.runoob.com/js/js-chrome.html>

### 【2】 VSCode上执行js

<https://juejin.cn/s/%E7%94%A8vscode%E5%86%99javascript%E7%9A%84%E6%95%99%E7%A8%8B>

<https://cloud.tencent.com/developer/article/2114052>

# Node.js及其相关

node.js官网：<https://nodejs.org/en>

## （1） 是什么

### 【1】 Node.js

#### 参1：<https://c.biancheng.net/view/9338.html>

在 Node.js 之前，JavaScript 只能运行在浏览器中，作为网页脚本使用，为网页添加一些特效，或者和服务器进行通信。有了 Node.js 以后，JavaScript 就可以脱离浏览器，像其它编程语言一样直接在计算机上使用，想干什么就干什么，再也不受浏览器的限制了。  
  
Node.js 不是一门新的编程语言，也不是一个 JavaScript 框架，它是一套 JavaScript 运行环境，用来支持 JavaScript 代码的执行。用编程术语来讲，Node.js 是一个 JavaScript 运行时（Runtime）。

#### 参2：<https://zhuanlan.zhihu.com/p/47822968>

Nodejs 是基于 Chrome 的 V8 引擎开发的一个 C++ 程序，目的是提供一个 JS 的运行环境。最早 Nodejs 主要是安装在服务器上，辅助大家用 JS 开发高性能服务器代码，但是后来 Nodejs 在前端也大放异彩，带来了 Web 前端开发的革命。

### 【2】 NPM

<https://www.itheima.com/news/20211012/175847.html>

npm(Node.js Package Manager)是一个Node.js的包管理工具，用来解决Node.js代码部署问题。在安装Node.js时会自动安装相应的npm版本,不需要单独安装。使用npm包管理工具可以解决如下场景的需求。

(1）从npm服务器下载别人编写的第三方包到本地使用。

(2）从npm服务器下载并安装别人编写的命令程序到本地使用。

(3）将自己编写的包或命令行程序上传到npm服务器供别人使用。

## （2） 安装

<https://cloud.tencent.com/developer/article/2114052>

## （3） 使用

<https://zhuanlan.zhihu.com/p/47822968>

### 【1】 交互环境

退出命令“.exit”：

<https://segmentfault.com/q/1010000000656245>

### 【2】 执行js文件

# TypeScript<TS>

官网（英）：<https://www.typescriptlang.org/>

官网（中）：<https://ts.nodejs.cn/>

## 是什么

<https://ts.nodejs.cn/docs/handbook/intro.html>

<https://ts.nodejs.cn/docs/handbook/typescript-from-scratch.html>

<https://www.runoob.com/typescript/ts-tutorial.html>

自总结：

1. ts是js的超集，它在js语法的基础上增加了新的功能
2. ts编译后是js，真正运行的是js
3. ts主要是为了对js代码进行编译时(静态)类型检查

## 安装

<https://ts.nodejs.cn/docs/handbook/typescript-tooling-in-5-minutes.html#%E5%AE%89%E8%A3%85-typescript>

### 通过npm

<https://www.runoob.com/typescript/ts-install.html>

## 语法

<https://www.typescriptlang.org/docs/handbook/2/basic-types.html>

<https://ts.nodejs.cn/docs/handbook/2/basic-types.html>

<https://www.runoob.com/typescript/ts-basic-syntax.html>

### 问号

<https://www.cnblogs.com/gg-qq/p/14649871.html>

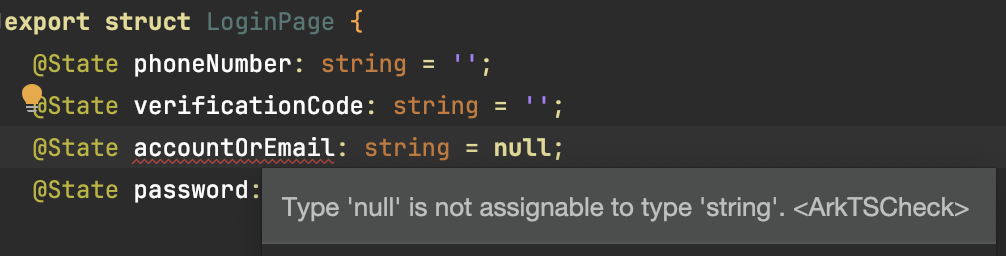
### .d.ts文件 和 declare

<https://blog.csdn.net/qq_34551390/article/details/118800743>

<https://www.51cto.com/article/710348.html>

### 给变量赋值null

一开始报错



<https://www.jiyik.com/tm/xwzj/web_891.html>

<https://stackoverflow.com/questions/59715119/type-null-is-not-assignable-to-type-t>

改成



解决。

### ts语句结尾是否要加分号

可加可不加。

<https://blog.csdn.net/Level_Gram/article/details/130524037>

<https://juejin.cn/s/typescript%20%E5%A3%B0%E6%98%8E%20%E4%B8%8D%E5%8A%A0%E5%88%86%E5%8F%B7>

<https://segmentfault.com/q/1010000020688161>

### $符号：模板字符串

<https://blog.csdn.net/sunxiaoju/article/details/98622352>

<https://blog.csdn.net/weixin_45819141/article/details/135708220>

<https://blog.csdn.net/qq_42444795/article/details/106148875>

注：模板字符串使用反引号包裹的``

(mac上打出反引号的方法<https://blog.csdn.net/u014530058/article/details/125997102>)

### static关键字

<https://blog.csdn.net/qq_41709082/article/details/96479594>

<https://ts.xcatliu.com/advanced/class.html>

<https://kimi.moonshot.cn/chat/cpg699g3r07fpuq1rlbg>

<https://blog.csdn.net/w440149517/article/details/113924535>

### 三个点（...）

<https://www.cnblogs.com/ProjectDD/p/11796998.html>

<https://blog.csdn.net/pujun1201/article/details/119775477>

<https://kimi.moonshot.cn/chat/cpndts83r074slpq5s10>

<https://www.jianshu.com/p/86cbabeda999>

<https://developer.aliyun.com/article/813180>

### 数据类型

#### any和unknown

<https://kimi.moonshot.cn/chat/cpnf6o9hd0n3mqu38lgg>

### 空安全

<https://blog.csdn.net/iamlujingtao/article/details/110573421>

<https://kimi.moonshot.cn/chat/cppvvs5dl8mppi4phqlg>

### 内部类

<https://segmentfault.com/q/1010000042706944>

问了枫楠，说ts中没有内部类的用法。

### 列表数据结构

<https://kimi.moonshot.cn/chat/cpqhr8u0atp3r6v3u47g>

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

<https://kimi.moonshot.cn/chat/cpqp8a0nsmmh6o5vlbtg> 相关操作

<https://www.runoob.com/typescript/ts-array.html> Array对象

### let和var

<https://www.cnblogs.com/RitaLee/p/10375068.html>

<https://kimi.moonshot.cn/chat/cpqq7ir5cfumdhrae9jg>

<https://blog.csdn.net/qq_41791705/article/details/139243408>

### 类中的属性

<https://typescript.p6p.net/typescript-tutorial/class.html>

<https://www.runoob.com/typescript/ts-variables.html>

注：属性不能用var、let、const修饰

### 正则表达式

<https://kimi.moonshot.cn/chat/cpqq7ir5cfumdhrae9jg>

### 整型

<https://kimi.moonshot.cn/chat/cpqrjnj3vddp3l5arte0>

### 异常机制

<https://kimi.moonshot.cn/chat/cpr49otvbf6l0atd3uog>

### 条件语句if的条件表达式的类型

不光能直接传boolean，其他类型会被转为boolean

<https://blog.csdn.net/dalqyou/article/details/85340707>

<https://blog.csdn.net/qq_39962271/article/details/132245400>

### 枚举

<https://blog.csdn.net/qq_40340943/article/details/125485741>

<https://juejin.cn/post/6979412028985180196>

<https://jkchao.github.io/typescript-book-chinese/typings/enums.html>

### type和interface

<https://zhuanlan.zhihu.com/p/92906055>

<https://blog.csdn.net/u013948858/article/details/124827980>

<https://juejin.cn/post/7007732766456414244>

### 感叹号（!）和问号（?）

<https://kimi.moonshot.cn/chat/cptme81kqq4h1ihb7vog>

### Map数据结构

<https://www.runoob.com/typescript/ts-map.html>

<https://cloud.tencent.com/developer/article/2300009>

### never类型

<https://jkchao.github.io/typescript-book-chinese/typings/neverType.html>

<https://juejin.cn/post/7201696941330939959>

## 不同平台执行ts

### 命令行

<https://ts.nodejs.cn/docs/handbook/typescript-tooling-in-5-minutes.html#%E6%9E%84%E5%BB%BA%E4%BD%A0%E7%9A%84%E7%AC%AC%E4%B8%80%E4%B8%AA-typescript-%E6%96%87%E4%BB%B6>

<https://www.runoob.com/typescript/ts-install.html>

### WebStorm

<https://www.cnblogs.com/yangfanjie/p/12036118.html>

## html中使用ts

<https://www.jianshu.com/p/186315755b3f>

<https://blog.csdn.net/zy21131437/article/details/121925159>

# DOM

## 是什么

### 参1：<https://kimi.moonshot.cn/chat/coscnp22jkock8anbb80>

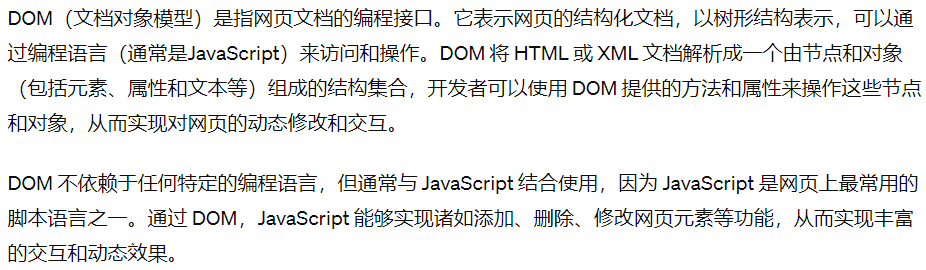
文本

描述已自动生成

### 参2：<https://en.wikipedia.org/wiki/Document_Object_Model>

The **Document Object Model** (**DOM**) is a [cross-platform](https://en.wikipedia.org/wiki/Cross-platform) and [language-independent](https://en.wikipedia.org/wiki/Language-independent_specification) interface that treats an [HTML](https://en.wikipedia.org/wiki/HTML) or [XML](https://en.wikipedia.org/wiki/XML) document as a [tree structure](https://en.wikipedia.org/wiki/Tree_structure) wherein each [node](https://en.wikipedia.org/wiki/Node_(computer_science)) is an [object](https://en.wikipedia.org/wiki/Object_(computer_science)) representing a part of the document.

### 参3：<https://chatgpt.com/c/fd61887b-c63e-4c92-9ccd-82d5cc36b757>



### 参4：<https://developer.mozilla.org/zh-CN/docs/Web/API/Document_Object_Model/Introduction>

DOM 被设计成与特定编程语言相独立，使文档的结构化表述可以通过单一，一致的 API 获得。尽管我们在本参考文档中会专注于使用 JavaScript，但 DOM 也可以使用其他的语言来实现，以 Python 为例，代码如下：

|  |
| --- |
| # Python DOM 示例  import xml.dom.minidom as m  doc = m.parse(r"C:\Projects\Py\chap1.xml")  doc.nodeName # 文档对象的 DOM 属性  p\_list = doc.getElementsByTagName("para") |

### 参5：<https://www.runoob.com/htmldom/htmldom-intro.html>

DOM 是 W3C（万维网联盟）的标准。

DOM 定义了访问 HTML 和 XML 文档的标准：

"W3C 文档对象模型 （DOM） 是中立于平台和语言的接口，它允许程序和脚本动态地访问和更新文档的内容、结构和样式。"

W3C DOM 标准被分为 3 个不同的部分：

* 核心 DOM - 针对任何结构化文档的标准模型
* XML DOM - 针对 XML 文档的标准模型
* HTML DOM - 针对 HTML 文档的标准模型