**1、什么是HTML语义化？**

<基本上都是围绕着几个主要的标签，像标题（H1~H6）、列表（li）、强调（strong em）等等>

　　根据内容的结构化（内容语义化），选择合适的标签（代码语义化）便于开发者阅读和写出更优雅的代码的同时让浏览器的爬虫和机器很好地解析。

**2、为什么要语义化？**

为了在没有CSS的情况下，页面也能呈现出很好地内容结构、代码结构

用户体验：例如title、alt用于解释名词或解释图片信息、label标签的活用；

有利于SEO：和搜索引擎建立良好沟通，有助于爬虫抓取更多的有效信息：爬虫依赖于标签来确定上下文和各个关键字的权重；

方便其他设备解析（如屏幕阅读器、盲人阅读器、移动设备）以意义的方式来渲染网页；

便于团队开发和维护，语义化更具可读性，是下一步吧网页的重要动向，遵循W3C标准的团队都遵循这个标准，可以减少差异化。

**3、写HTML代码时应注意什么？**

尽可能少的使用无语义的标签div和span；

在语义不明显时，既可以使用div或者p时，尽量用p, 因为p在默认情况下有上下间距，对兼容特殊终端有利；

不要使用纯样式标签，如：b、font、u等，改用css设置。

需要强调的文本，可以包含在strong或者em标签中（浏览器预设样式，能用CSS指定就不用他们），strong默认样式是加粗（不要用b），em是斜体（不用i）；

使用表格时，标题要用caption，表头用thead，主体部分用tbody包围，尾部用tfoot包围。表头和一般单元格要区分开，表头用th，单元格用td；

表单域要用fieldset标签包起来，并用legend标签说明表单的用途；

每个input标签对应的说明文本都需要使用label标签，并且通过为input设置id属性，在lable标签中设置for=someld来让说明文本和相对应的input关联起来。

HTML 5的革新之一：**语义化标签一节元素标签**。

在HTML 5出来之前，我们用div来表示页面章节，但是这些div都没有实际意义。（即使我们用css样式的id和class形容这块内容的意义）。这些标签只是我们提供给浏览器的指令，只是定义一个网页的某些部分。但现在，那些之前没“意义”的标签因为因为html5的出现消失了，这就是我们平时说的“语义”。

但是也不要因为html5新标签的出现，而随意用之，错误的使用肯定会事与愿违。所以有些地方还是要用div的，就是因为div没有任何意义的元素，他只是一个标签，仅仅是用来构建外观和结构。因此是最适合做容器的标签。

**header元素：双标签**

header 元素代表“网页”或“section”的页眉。

通常包含h1-h6元素或hgroup，作为整个页面或者一个内容块的标题。也可以包裹一节的目录部分，一个搜索框，一个nav，或者任何相关logo。

整个页面没有限制header元素的个数，可以拥有多个，可以为每个内容块增加一个header元素

header使用注意：

可以是“网页”或任意“section”的头部部分；

没有个数限制。

如果hgroup或h1-h6自己就能工作的很好，那就不要用header。

**footer元素**

footer元素代表“网页”或“section”的页脚，通常含有该节的一些基本信息，譬如：作者，相关文档链接，版权资料。如果footer元素包含了整个节，那么它们就代表附录，索引，提拔，许可协议，标签，类别等一些其他类似信息。

footer使用注意：

可以是“网页”或任意“section”的底部部分；

没有个数限制，除了包裹的内容不一样，其他跟header类似。

**hgroup元素**

hgroup元素代表“网页”或“section”的标题，当元素有多个层级时，该元素可以将h1到h6元素放在其内，譬如文章的主标题和副标题的组合

hgroup使用注意：

如果只需要一个h1-h6标签就不用hgroup

如果有连续多个h1-h6标签就用hgroup

如果有连续多个标题和其他文章数据，h1-h6标签就用hgroup包住，和其他文章元数据一起放入header标签

**nav元素**

nav元素代表页面的导航链接区域。用于定义页面的主要导航部分。

但是我在有些时候却情不自禁的想用它，譬如：侧边栏上目录，面包屑导航，搜索样式，或者下一篇上一篇文章，但是事实上规范上说nav只能用在页面主要导航部分上。页脚区域中的链接列表，虽然指向不同网站的不同区域，譬如服务条款，版权页等，这些footer元素就能够用了。

nav使用注意：

用在整个页面主要导航部分上，不合适就不要用nav元素；

**aside元素**

aside元素被包含在article元素中作为主要内容的附属信息部分，其中的内容可以是与当前文章有关的相关资料、标签、名次解释等。（特殊的section）

在article元素之外使用作为页面或站点全局的附属信息部分。最典型的是侧边栏，其中的内容可以是日志串连，其他组的导航，甚至广告，这些内容相关的页面。

aside使用总结：

aside在article内表示主要内容的附属信息，

在article之外则可做侧边栏，没有article与之对应，最好不用。

如果是广告，其他日志链接或者其他分类导航也可以用

**section元素**

section元素代表文档中的“节”或“段”，“段”可以是指一篇文章里按照主题的分段；“节”可以是指一个页面里的分组。

section通常还带标题，虽然html5中section会自动给标题h1-h6降级，但是最好手动给他们降级。

section使用注意：

一张页面可以用section划分为简介、文章条目和联系信息。不过在文章内页，最好用article。section不是一般意义上的容器元素，如果想作为样式展示和脚本的便利，可以用div。

表示文档中的节或者段；

article、nav、aside可以理解为特殊的section，所以如果可以用article、nav、aside就不要用section，没实际意义的就用div

**article元素**

article元素最容易跟section和div容易混淆，其实article代表一个在文档，页面或者网站中自成一体的内容，其目的是为了让开发者独立开发或重用。譬如论坛的帖子，博客上的文章，一篇用户的评论，一个互动的widget小工具。（特殊的section）

除了它的内容，article会有一个标题（通常会在header里），会有一个footer页脚。我们举几个例子介绍一下article，好更好区分article、section、div

设计师、程序员、前端工程师都是一个独立的整体，他们组成了网站制作基本配备，当然还有其他成员~~。设计师、程序员、前端工程师就像article，是一个个独立的整体，而section将这些自成一体的article包裹，就组成了一个团体。

article和section和例子就例举这么多了，具体情况具体分析，不易深究。漏了divd，其实div就是只是想用来把元素组合或者给它们加样式时使用。

article使用注意：

自身独立的情况下：用article

是相关内容：用section

没有语义的：用div

**HTML5其他结构元素标签**

HTML5节元素标签包括body article nav aside section header footer hgroup ，还有h1-h6 address。

address代表区块容器，必须是作为联系信息出现，邮编地址、邮件地址等等,一般出现在footer。

h1-h6因为hgroup，section和article的出现，h1-h6定义也发生了变化，允许一张页面出现多个h1。

（特别鸣谢：<https://www.cnblogs.com/freeyiyi1993/p/3615179.html>）

<embed>

<marquee></marquee>

**CSS 盒子模型(Box Model)**

所有HTML元素可以看作盒子，在CSS中，"box model"这一术语是用来设计和布局时使用。

CSS盒模型本质上是一个盒子，封装周围的HTML元素，它包括：边距，边框，填充，和实际内容。

盒模型允许我们在其它元素和周围元素边框之间的空间放置元素。

不同部分的说明：

**Margin(外边距**) - 清除边框外的区域，外边距是透明的。

**Border(边框)** - 围绕在内边距和内容外的边框。

**Padding(内边距)** - 清除内容周围的区域，内边距是透明的。

**Content(内容)** - 盒子的内容，显示文本和图像。

元素的宽度和高度

Remark重要: 当您指定一个CSS元素的宽度和高度属性时，你只是设置内容区域的宽度和高度。要知道，完全大小的元素，你还必须添加填充，边框和边距。.

最终元素的总宽度计算公式是这样的：

总元素的宽度=宽度+左填充+右填充+左边框+右边框+左边距+右边距

元素的总高度最终计算公式是这样的：

总元素的高度=高度+顶部填充+底部填充+上边框+下边框+上边距+下边距

浏览器的兼容性问题

一旦为页面设置了恰当的 DTD，大多数浏览器都会按照上面的图示来呈现内容。然而 IE 5 和 6 的呈现却是不正确的。根据 W3C 的规范，元素内容占据的空间是由 width 属性设置的，而内容周围的 padding 和 border 值是另外计算的。不幸的是，IE5.X 和 6 在怪异模式中使用自己的非标准模型。这些浏览器的 width 属性不是内容的宽度，而是内容、内边距和边框的宽度的总和。

虽然有方法解决这个问题。但是目前最好的解决方案是回避这个问题。也就是，不要给元素添加具有指定宽度的内边距，而是尝试将内边距或外边距添加到元素的父元素和子元素。

IE8 及更早IE版本不支持设置填充的宽度和边框的宽度属性。

解决IE8及更早版本不兼容问题可以在HTML页面声明 <!DOCTYPE html>即可。

一些心得：HTML和CSS还有好多好多要学习的东西呀，标签其实只用了好少好少的一部分，因为上ssd1的原因，大概熟悉一点点。知道CSS外联样式，内联样式表，内联样式什么的，但要熟练掌握和运用还是比较需要再花时间的。

好像有的浏览器排版会炸诶……目前还没找到解决方案。