# 几个概念

## 前端

前端就是指用户能够看到的页面效果，都是前端。都是我们前端程序员做的。

## 后端

后端就是用来实现功能的。

## 操作系统

买回来来的电脑只是一个硬件，需要安装操作系统后才能使用。

常见的操作系统有：windows、Linux、mac、ios、Android、鸿蒙等等

## 浏览器

浏览器就可以运行我们写的html、css、js代码。

常见的浏览器：

谷歌浏览器、火狐浏览器、IE、欧朋、苹果等等。

# HTML

## 概念

HTML就是超文本标记语言，学习HTML就是学习里面的各种标签及标签具有的属性。

使用HTML就可以制作网页的结构。

一个网页由三部分组成：

* 结构：html，用来显示网页的内容
* 表现：css，用来给html标签设置样式，使页面更美观
* 行为：JavaScript，可以加一些动作行为、特效

html 代码就是写后缀是 .html 的文件中的。

## 标签

标签分为两类：单标签、双标签。

单标签：自己又是开始又是结束。比如 <br/> <hr/>等

双标签：有开始标签和结束标签。比如 <p></p> <span></span>等

## 属性

在标签的开始标签中可以写属性，可以写多个属性，多个属性之间使用空格隔开。

比如：<a href=”路径” title=””></a>

## 编码方式

我们见到的各种文件存储在计算机上都会转为 0、1。那对于字符需要使用响应的编码方式去存储。

ISO-8859-1 欧洲

ASCII 美国

GBK 中国

UTF-8 万国码

以后我们在开发中使用的就是utf-8。

## 文档结构

|  |
| --- |
| <!-- 告诉浏览器，我们使用的是html5 版本，浏览器就会使用标准模式去解析我们的网页；如果没有文档声明，浏览器就会使用混合模式去解析我们的网页 -->  <!DOCTYPE html>  <!-- html是网页的根标签，我们写的代码都在这里面 -->  <html>  <!-- head是头标签，这里面的内容不会显示在浏览器页面中，这里面的东西是帮助浏览器更好的解析我们的网页 -->  <head>      <!-- 指定编码方式 -->      <meta charset="UTF-8">      <!-- 为了兼容IE浏览器的，如果用户用的是IE浏览器，就使用IE最高版本去解析 -->      <meta http-equiv="X-UA-Compatible" content="IE=edge">      <!-- 为了做响应式布局写的。 -->      <meta name="viewport" content="width=device-width, initial-scale=1.0">      <!-- 标签页的标题 -->      <title>Document</title>  </head>  <!-- 网页的主体，写在这里面的代码会显示在浏览器页面中 -->  <body>      <a href="#" title="百度">百度一下</a>      爱仕达多  </body>  </html> |

## 元素

元素就是和标签是一个意思。

元素：element。

## 语义化标签

我们学习的html标签，大部分都是语义化标签，也就是标签都是有含义的，比如p标签就是表示段落、a标签就是表示超链接。

语义化标签的作用：便于我们去阅读理解代码。

## 开发工具

VScode、Hbuilder、webstorm等等。DreamWeaver。

## 常见的标签

### 标题标签

标题标签，就是可以用来指定一个网页、一个部分的主题。

标签有6个，h1 到 h6。h1 最大、h6 最小。

标题标签有默认的样式。

### 段落标签

p标签，就是用来表示一个段落。

有默认的样式。

里面有align属性：可以设置段落中内容的对齐方式。left、center、right。

### 换行标签

br标签，用来换行。浏览器是不识别我们代码中的回车换行，要换行需要使用br标签进行换行。

浏览器会认为代码中的回车是一个空格。

<br/> <br>

### 水平线标签

hr标签，可以在页面显示一条水平线。

### 图片标签

img表示表示图片标签，可以让浏览器显示一张图片。

|  |
| --- |
| <img src="图片的路径" alt="图片加载失败时的提示信息" title="鼠标悬浮的提示信息"> |

### 路径

路径分为两种：

绝对路径：类似于,http://xxxx/xxxxx

相对路径：相对路径就是相对于当前文件所在的文件夹去找的。

. 表示当前文件所在目录，可以省略

.. 表示返回上级目录

### 超链接标签

超链接标签就是 a 标签。它的作用就是点击后可以进行跳转。

|  |
| --- |
| <a href="跳转的路径" target="\_blank">显示的文字</a> |

href指定要跳转到的路径

target指定以什么样的方式去另一个页面，可选值：

\_self：默认，就在当前标签页中打开

\_blank：可以新打开一个标签页显示另一个页面

### 列表标签

列表标签有三种：无序列表、有序列表、定义列表。列表主要就是用来显示信息。

有序列表：ol li

无序列表：ul li

定义列表：dl dt dd

重点使用 无序列表。列表是可以嵌套的。

|  |
| --- |
| <ul>          <li><a href="#"></a></li>          <li><a href="#"></a></li>          <li><a href="#"></a></li>      </ul> |

### 表格标签

table就表示表格标签，主要用来展示数据的。

|  |
| --- |
| <table>          <tr>              <td></td>              <td></td>              <td></td>          </tr>          <tr>              <td></td>              <td></td>              <td></td>          </tr>          <tr>              <td></td>              <td></td>              <td></td>          </tr>      </table> |

table表示表格

tr表示表格中的一行

td标签行中的一个单元格

平时我们写表格是就用上面的三个标签，但是浏览器在解析网页的时候会默认将所有的tr都用tbody给套起来。

thead表示表格中的头部

tbody表示表格的主体内容

tfoot表示表格的底部

valign属性可以设置单元格中的内容在垂直方向的对齐方式。

合并单元格需要掌握：

rowspan：跨行合并单元格

colspan：跨列合并单元格

合并是从左到右、从上到下。

### 表单标签

form就是表单标签，主要用来收集用户输入的信息提交给服务器。

表单项，经常是input标签，根据type属性的取值不同有不同的表现形式。

type=”text” 普通的文本输入框

type=”password” 密码框

type=”radio” 单选按钮

type=”checkbox” 复选框

type=”submit” 提交按钮

type=”file” 上传文件

type=”reset” 重置按钮

type=”button” 普通按钮

type=”hidden” 隐藏域

select、option 是下拉列表

textarea 是文本域，多行文本

label 标签需要配合 单选按妞 和 复选框使用，提升体验度的。

button 按钮标签，根据 type属性的不同有不同的作用。type可以是 submit、reset、button

### 表单项的属性

placeholder：可以在输入框给用户一个默认的文字提示，一旦用户输入内容时，会自动消失。

autofocus：打开页面后光标会自动聚焦。

checked属性：单选按钮和复选框默认选中一些选项时使用checked属性

selected属性：下拉列表默认选中某个选项时可以使用

minlength、maxlength：设置输入的最小和最大的字符个数

readonly：只读，用户只能看不能改里面的内容

disable：不可用。颜色是灰色的。

用户输入的内容要想都能够提交给服务器，表单项必须起name属性

用户通过鼠标点击的东西，比如单选按钮、复选框，这些必须给value属性

### 布局标签

div标签和span标签。

这两个标签是没有语义。它俩主要用来做布局，我们将其当做一个盒子、容器，里面放我们的内容。

块元素主要用来做布局。

一般我们都是用块元素去嵌套行内元素，而不会用行内元素去嵌套块元素。

### 视频标签

视频标签video，是h5新增的标签。

可用的属性：

controls：显示控制面板的

loop：循环播放

muted：静音

width：播放器宽度

height：播放器高度

autoplay：自动播放

preload：预加载

poster：设置视频的封面

### 音频标签

音频标签也是h5 新增的，audio。

可以用的属性：

controls：面板

loop：循环播放

muted：静音

autoplay：自动播放

preload：预加载

### 布局标签

h5 新增的语义化的布局标签有：

header：头部

nav：导航

aside：侧边栏

article：文章、博客

section：段落、评论

footer：底部

这些标签本质上和div没区别，只不过是多了个语义。

<header></header>

<div class=”header”></div>

## 元素的分类

html中的元素分为两类：

块元素：

特点就是会独占一整行，宽度默认是父元素的百分之百，高度默认由内容撑起来，在页面中垂直摆放。

行内元素：

特点就是不会独占一整行，宽度和高度都由内容撑起来。

## 元素之间的关系

父子关系

祖先后代关系（父子关系也是一种特殊的祖先后代关系）

兄弟关系

## 长度单位

px：像素，就是浏览器中的一个点。

%：百分比，一般是相对于父元素来说的。

# CSS

## 概念

一个网页包含三部分：结构、表现和行为。

CSS就是用来做表现，它可以使页面更美观。

CSS是层叠样式表。

学习CSS就是学习里面的选择器、样式属性。

## 特点

继承性：祖先元素的样式属性，后代元素是可以继承的。但是有些样式是不能继承的。而且如果后代元素有自己的样式，就不会用祖先的了。

能被继承的：

字体相关、行高、颜色、文本相关。

层叠性：可以用多种方式给同一个元素设置样式，但是注意，同样的样式最终是哪个生效要看优先级。

优先级：优先级高的样式生效，优先级一样的写在后面的生效。

## 引入方式

行内样式、内部样式、外部样式。

工作中使用的外部样式。实现了html代码和css代码分离；复用性强。

这三种方式的优先级：行内样式 > 内部样式|外部样式

行内样式：将样式写在开始标签的style属性中

内部样式：将样式写在head标签中的style标签中

外部样式：将样式写在外部的 .css 文件中，哪个页面要用就通过link标签引入

## 选择器

### 全局选择器

语法格式：

\*{ }

可以选中页面中所有的元素。

### id选择器

其实每个标签都可以有id属性，表示该元素的一个唯一标识。

在一个页面中id不要重复。

#id值{}

比如：#abc{} 选中id=”abc” 的元素

### 类选择器

我们经常用。每个标签都可以有 class 属性，class的值是可以重复的，而且class的值可以有多个，多个之间使用空格隔开。

<div class=”wrap nav”></div>

语法格式：

.class值{}

.abc{} 表示选中 class=”abc” 的元素

### 标签选择器

语法格式：

标签{}

比如： div{} 选中所有的div元素

### 权重

行内样式 1000

id选择器是100

class选择器是10

标签选择器是1

### 并集选择器

一次性选中多个选择器选择的元素。

语法： 选择器1,选择器2，选择器n{}

比如 .abc，#ccc,div{} 给class=”abc” 的元素、 id=”ccc”的元素、所有的div元素 都设置样式

### 交集选择器

通过多个条件选中元素。

语法：选择器1选择器2选择器n{}

比如： div.abc{} 选中class=“abc”的div标签。

### 子元素选择器

语法格式：父元素>子元素{}

比如： .abc>#ccc{} 选择class=”abc”里面的id=”ccc”的儿子元素

### 后代元素选择器

语法格式：祖先 后代元素{}

比如：.abc div{} 选择 class=”abc” 里面的所有的div元素

### 相邻兄弟选择器

语法格式：前一个兄弟+后一个兄弟{}

比如： .abc+button{} 选择class=”abc”的元素后面紧挨着的button元素

### 通用兄弟选择器

语法格式：A~B{}

比如：.abc~.ccc{} 选择 class=”abc”后面的所有的class=”ccc”的兄弟

## 布局的思路

布局的思路，一见到页面，不要急着写内容，先分析布局，做好布局，然后往里面填充内容。

具体做：从上到下、从左到右、从外到内去看。

## 样式属性

### 字体相关的属性

#### color

设置字体的颜色。

#### font-size

设置字体大小，单位是 px。浏览器默认字体大小是 16px，支持的最小字体是 12px。

#### font-weight

设置字体的粗细。

400 - 正常粗细，相当于 normal

700 - 加粗，相当于 bold

#### font-style

设置字体是否倾斜。

#### font-family

设置字体，比如 宋体、黑体。一般写多个值，用户电脑支持哪个就用哪个。

比如：font-family: 黑体 宋体 ;

#### line-height

设置行高。

浏览器中的文字都是在看不到的两条线之间垂直居中的。这两条线之间的距离就是行高。

行高的单位可以是 px，也可以是数字，表示 几倍的当前字体大小。比如 1.5。

行高一个很重要的应用就是可以设置一行文字在元素中垂直居中。

font:14px/1.5 宋体,黑体;

### 文本相关属性

#### text-align

设置元素中内容的对齐方式。

内容指定是行内元素和文字内容。（行内块元素也可以）、

#### text-decoration

可以去掉超链接默认的下划线。 也可以加下划线、删除线等

#### text-transform

文本转换，主要针对英文的。可以设置单词全大写、全小写、首字母大写等。

#### text-indent

设置文本缩进的，比如一个段落首行缩进两个字符。

em的单位：是一个百分比单位，相对于当前字体的大小。 1em = 1 font-size

### 背景相关的属性

#### background-color

设置背景颜色。

#### background-image

设置背景图片。

background-image:url();

#### background-repeat

设置背景平铺方式。默认背景会水平垂直平铺。

no-repeat是不平铺

#### background-size

设置背景图片的大小。

#### background-position

设置背景图片的位置。

#### background-attachment

设置背景图片固定，不随着滚动条而滚动。

fixed。

#### 简写属性

background:xxx xx xxxx;

要注意：背景图片，只是背景而已，其实元素里面什么东西都没有呢，所以可以给元素中写内容，就会在图片上了。 和 img 标签区分开。

### 列表属性

#### list-style

可以去掉有序和无序列表的项目符号！

### overflow属性

overflow属性很重要，作用：

1. 隐藏溢出的那部分内容
2. 开启BFC，使元素变为独立的一块，布局不会受到子元素的影响。

### 其它小属性

#### letter-spacing

设置字符之间的距离。

#### opacity

给元素设置透明度。

1. 完全透明 1-完全不透明。

## 颜色的写法

### 直接写颜色单词

直接写颜色单词，比如 red blue green等等，用的比较少。

### rgb表示

通过红色、绿色、蓝色的浓度去调配颜色。

rgb(红色的浓度，绿色的浓度,蓝色的浓度)

浓度的取值：0 - 255

0 0 0 表示黑色

255 255 255 表示白色。

### 十六进制的写法

本质上还是rgb的表示方式，只不过是将 0-255 用十六进制来表示的，

0-255 对应的 00 - ff 表示了。

语法格式：#红色绿色蓝色，比如 #ab1122

#aabbcc 可以简写为 #abc。但是 #aabbcd 不能简写。

### rgba的写法

本质上和 rgb一样，就是多了一个颜色的透明度。

语法rgba(红色，绿色，蓝色，透明度)

透明度是 0 到 1， 0-表示完全透明，1-完全不透明

## 盒模型

### 标准盒模型

#### 介绍

标准盒模型是我们最常用的盒模型。

标准的盒模型的组成部分：

内容区（content）：就是能到内容的区域，设置的width和height就是设置给了内容区

内边距（padding）：就是内容与边框之间的距离

边框（border）

外边距（margin）：盒子与盒子边框之间的距离

实际占据的大小：

水平方向：内容区+左右内边距+左右边框+左右外边距

垂直方向：内容区+上下内边距+上下边框+上下外边距

规律：

1个数 4个方向都一样

2个数 上下 左右

3个数 上 左右 下

4个数 上 右 下 左

#### 元素水平居中

可以使用 margin：0 auto；实现块元素水平居中。

而且一个块元素要使用margin水平居中，得设置块元素的宽度。

#### 外边距合并问题

垂直摆放的两个元素，上面的设置了下外边距a，下面的设置的上外边距b，那它们两个之间的外边距是较大的那个数。

水平摆放的两个元素，左面的设置了右外边距a，右面的设置的左外边距b，那它们两个之间的外边距是a+b。

#### 垂直方向外边距重叠问题

垂直方向外边距重叠：给子元素设置了上外边距，发现父元素也跟着一块下来了。

怎么解决？

1. 给父元素开启BFC，overflow：hidden。
2. 子元素浮动，或者父元素浮动
3. 给父元素加个1px 的上边框
4. 给父元素加个 1px 上内边距