## HTML

#### 概念

HTML全称“Hyper Text Markup Language”,即超文本标记语言，用来描述网页。HTML5在HTML的基础上进行了新增或者废弃一些标签和特性。

#### 基本结构

#### 基本标签

标签分为一般标签和自闭合标签。一般标签有开始符号和结束符号，可以在其中间加入其他标签和文字，例如：<body></body>；自闭合标签只有开始符号，没有结束符号，只能定义自身的属性，例如：<hr/>。可以通过<!--…-->来添加注释。

###### HTML

网页从<html>开始，到</html>结束。

###### head

定义特殊内容，在网页中“不可见”。其内部常见标签主要有<title><style>等。

###### body

定义网页展示内容，在网页中可见。其内部常见标签主要如下：

* 1. <h1>~<h6>

该标签为标题，级别依次递减。

* 1. <p>

该标签定义段落。

* 1. <hr>

该标签定义水平线。

* 1. <div>

该标签定义文档中的分区或节，是一个块级元素。

* 1. <ol>、<ul>

<ol>标签定义了一个有序列表，列表排序以数字来显示。<ul>标签定义了一个无序列表。列表内容由<li>引用。

* 1. <button>

该标签定义了一个按键。

* 1. <select>

该标签定义选项，内部以<option></option>定义选项，可设置多个。

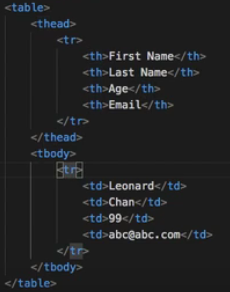
* 1. <a>

用以定义超链接。<a href=””></a>，target="\_Blank"可以用来开一个新的界面。

* 1. 除此之外，还有<strong>用以加粗字体，<sup>用来设置上标，<sub>用来设置

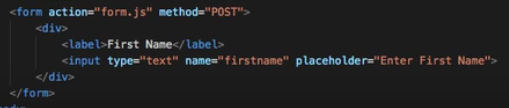
下标

* 1. <table>

定义表格，可用CSS对其进行美化，格式如下：

表格有两类单元格：

* 表头单元-包含头部信息(由th元素创建)
* 标准单元-包含数据（由td元素创建）
* td元素中的文本显示为正常字体且左对齐
  1. <form>

定义表单，参考格式如：

#### 4.块元素和行内元素

1. 块元素独占一行，包括块元素和行内元素，常见块元素有：h1~h6、p、hr、div；
2. 行内元素可以与其它行内元素位于同一行，可以容纳其它元素，但不可以容纳块元素，常见行内元素有：strong、em、span。

## CSS

#### 概念

“Cascading Style Sheet”即层叠样式表，是一组样式设置的规则，用于控制页面的外观样式。主要作用是页面外观的美化和其布局定位。与HTML一样并不是编程语言。

#### 语法

<head>

<style>

选择器{

属性名：属性值；

属性名：属性值；

}

</style>

</head>

* 选择器：要修饰的对象
* 属性名：修饰对象的哪一个属性
* 属性值：样式的取值

#### CSS添加方式

1. 外部样式表

保存在.css文件里；在HTML里使用<link>引用。（在<head>里引用）

1. 内部样式表

不适用外部CSS文件；将CSS放在HTML<style>里。（一般也放在<head>里）

1. 内联样式（不常使用）

仅影响一个元素；在HTML元素的style属性中添加。

#### 选择器选择方式

* 通配选择器（\*）
* 元素选择器（eg:p）
* 类选择器（.class）
* ID选择器（#id）

!important 优先级最高

#### CSS颜色表达方式

* 关键词：black,white等。
* 十六进制值：#ff0000。
* RGB：rgb（0，0，0）

#### CSS样式

###### 字体样式

font-size：字体大小

font-family：字体样式

font-weight：字重(normal 普通，bold粗体)

letter-spacing：字母间距

word-spacing：单词间距

line-height：行距

……

###### box模型

margin-top:上边距 4个上右下左

margin-bottom:下边距

margin-right:右边距

margin-left:左边距

border：加边框，可具体指哪条边框

border-radius:边框圆角

#### 7.CSS定位方式

###### static 默认定位

默认值，即没有定位。

top，left，right，bottom属性无效。

relative 相对定位

相对于原来位置的偏移，可能会覆盖其他元素。

absolute 绝对定位

以body元素的左上角为原点进行相对定位，以所有自己的父元素中离自己最近的一个除static定位以外的父元素的左上角为原点。脱离文档流。

fixed 固定定位

相对于屏幕视窗的位置定位，元素的位置不会随着屏幕滚动发生变化。

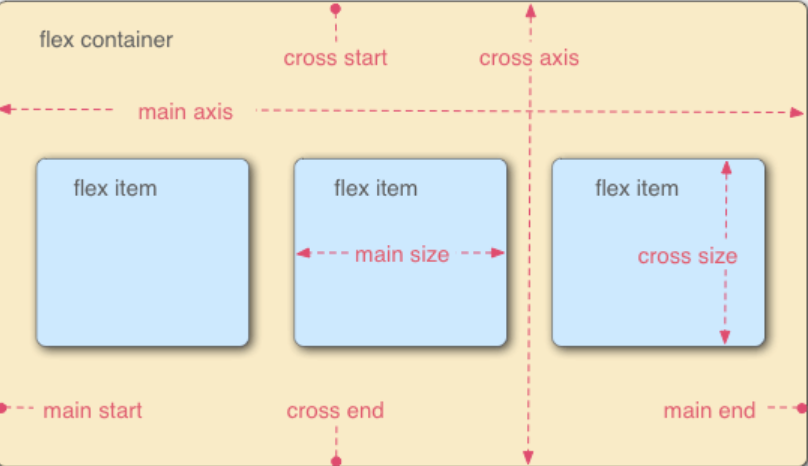
#### 浮动

* float：left 让元素向左浮动
* float：right 让元素向右浮动
* inherit 让元素从父级继承浮动属性

脱离文本流

宽度不能写百分比

#### Flex

即弹性布局，任何一个容器都可以指定为Flex布局。设定为Flex布局后子元素的float属性将失效。

* main axis 水平的主轴
* cross axis 垂直的交叉轴

###### 容器的属性

flex-direction

决定项目排列的方向

* row：主轴沿水平方向，起点在左端
* column：主轴沿垂直方向，起点在上端
* row-reverse：主轴沿水平方向，起点在右端
* column-reverse：主轴沿垂直方向，起点在下端

flex-wrap

flex-wrap属性定义，项目如果在一条轴线上放不下，如何换行。

* nowrap：不换行
* wrap：换行，第一行在上方
* wrap-reverse：换行，第一行在下方

flex-flow

是flex-direction和flex-wrap的简写方式，默认值为row nowrap

justify-content

该属性定义了项目在主轴上的对齐方式

假设主轴沿水平方向，起点在左端（row）

* flex-start：左对齐
* flex-end：右对齐
* center：居中
* space-between：两侧对齐，项目之间的间隔都相等
* space-around：每个项目两侧的距离相等

align-items

该属性定义项目在交叉轴上如何对齐

假设交叉轴从上到下

* flex-start：交叉轴的起点对齐
* flex-end：交叉轴的终点对齐
* center：交叉轴的中点对齐
* baseline：项目第一行文字的基线对齐
* stretch：默认值

align-content

该属性定义了多根轴线的对齐方式，如果轴线只有一条，则该属性不起作用。

* flex-start：与交叉轴的起点对齐
* flex-end：与交叉轴的终点对齐
* center：与交叉轴的中点对齐
* space-betwee：与交叉轴两端对齐，轴线之间的间隔平均分布。
* stretch：轴线占满整个交叉轴
* space-around：每根轴线两侧的间隔都相等。

###### 项目的属性

* order：定义项目的排列顺序，数值越小，排列越靠前。
* flex-grow：定义项目的放大比例。

为1：放大

为0：默认不放大

* flex-shrink：定义项目的缩小比例。

为1：空间缩小时，所有项目都等比缩小。

为0：即便空间不够，自身也不缩小。

* flex-basis：用于设置项目宽度，默认auto，项目会保持默认宽度，或者以width为自身的宽度。
* flex：是flex-grow、flex-shrink和flex-basis三个属性的简写，用于定义项目放大，缩小与宽度。

为auto：计算值为1，1，auto

为initial：计算值为0，1，auto

为none：计算值为0，0，auto

为inherit：从父元素继承。

#### grid

grid是基于网格的二维布局系统，目的是为了优化用户界面设计。

* Grid lines 单元格线
* Columns 列
* Rows 行
* Cells单元格
* Gutters 间距：两个网格轨道之间的区域
* Grid area网格区域：是由任何四个网格线组成的区域，它可以包含任意数量的网格单元格