HTML5 CSS3

网页的三个部分：结构（HTML），表现（CSS），行为（JavaScript）

# HTML

## 基础

### 结构：

<!-- doctype声明语法规范是html5 –>

<!doctype html>

<html>

<head>

<title>标题</title>

</head>

<body>

<header></header>

<main>

<h1>这是我的<font color=”yellow”>第一个</font>网页</h1>

</main>

<footer></footer>

<nav></nav>

<aside></aside>

<article></article>

</body>

</html>

### 一些label：

meta label：represent metadata，写在head中，元数据不给用户看。name（指定的数据名称，关键字）、charset（指定网页的字符集）、content（指定的数据内容）



<h1> ~ <h6>一级标题到六级标题，重要程度递减，一个页面只有一个h1（因为是最重要标题）。

<hgroup></hgroup>标题组，可以放入各种标题

block element：主要控制网页布局

inline element：主要用来包裹文字（例如上述h1）

一般来说是block内放置inline元素，block元素立马基本什么都能放。如果写的不符合要求，网页会自动修正，例如：把标签写在了html的外部，p元素中嵌套了block等等。

div表示一个block，是主要的布局元素，可以替代header，main，footer，nav等等。

列表：ul，ol，dl

超链接：<a href=“../01.html” target=”\_blank”></a>

href=”javascript:;” 可以创建没有任何功能的超链接

图片：<img width=”” height=”” src=”” alt=”松鼠”> img属于替换元素（介于inline和block之间），alt标签是辅助搜索引擎搜索关键字的

iframe：<iframe src=”[www.qq.com](http://www.qq.com)” width=”800” height=”900” frameborder=”0”></iframe>直接将其他网页显示在指定的长和宽框架内

媒体播放：<audio controls><source src=””><embed src=”” type=”audio/mp3” width=”” height=””></audio>; <video controls></video>

# CSS

第一种方式，写在标签内，比如<p style=color:red; font-size:60px;>hihihi</p>

第二种方式，内部样表式，在head中用<style></style>标签写样式，所有样式在style标签内。

第三种方式，写在CSS文件里，然后HTML文件里用<link rel=”stylesheet” href=”./style.css”>引入。

## 选择器

id选择：给元素一个id，然后针对此id设置css，例如：<p id=”red”></p> #red{color: red;}

class选择：给元素设置一个class，所有使用此class的元素公用一套css，例如：<p class=”red”></p> .red{color: red;}（可以同时为一个元素设置多个class）

交集选择：例如：div.red{font-size:30px;}此为将类型为div并且使用class=”red”的元素里的字体都调整为30px；.a.b.c{color=white;}意为将同时使用abc三个class的元素的颜色改成白色。

并集选择：例如：h1, span{color:green;}以为将所有h1和span类型改为绿色，和交集选择不同，交集选择是必须同时满足所有要求，并集选择只要满足其中一条要求就行（用逗号隔开）。

子元素选择：div#father > span{color:red;} 意为将id为father的div内部的span元素颜色改为红色。（子元素指直系，不包括子类中的子类）

后代选择：div span{color:red;} 意为将div下所有的后代span元素都设置为红色。（后代指所有父类下的元素，包括子类中的子类）

下一个选择：p + span{color:red;} 意为将p元素紧挨着的span元素设置为红色。（中间隔着别的元素就失效）

下边所有选择：p ~ span{color:red;} 意为将p元素下面的所有span元素都设置为红色。（中间隔着别的元素也无所谓）

属性选择：p[title]{color:red;} 意为将所有带有title属性值（<p title=””></p>）的p元素设置为红色。

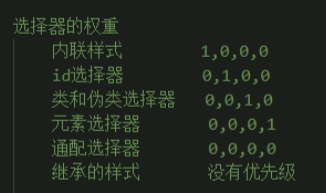
其他变体：p[title=abc]{} p[title^=abc]{} 意为title是abc开头的（包括abc，abcde等等） p[title$=abc]{} 意为title是abc结尾 p[title\*=abc]{} 意为含有abc

伪类选择：描述一个元素特殊状态比如：第一个元素，被点击元素，鼠标移入元素等等。例子：ul > li:first-child{} 指ul底下第一个子元素且必须是li。除此之外还有 last-child，nth-child()括号里可以输入n，2n或even，2n+1或odd。

其他变体：ul > li:first-of-type{} 第一个li子元素，除此之外还有last-of-type，nth-of-type()。

not：ul > li:not(:nth-of-type(3)){} 除了第三个li元素外其余的li元素。

选择器权重：

内联就是在<div style=””>里写的

## 长度单位

em：相对于元素的字体大小来计算，会根据字体的大小而改变。1em=1 font-size，2em=2font-size，假如设置font-size=20px，那么1em=20px。

rem：相对于html元素的字体大小来计算，只根据html根的font-size来计算。

## 文档流（normal flow）

normal flow是网页的最底层。通过css可以为每一层设置样式，作为用户只能看到最顶上一层，这些层中最底下的一层称为normal flow，是网页的基础。

文档流的特点：block元素会在页面独占一行（无论宽度高度都独占一行），默认宽度是父元素的全部（会把父元素撑满），默认高度是被内容撑开的高度（元素有多高block就有多高）

行内元素不会独占一行，只占自身大小，行内元素在页面中左向右排列。

## 盒模型（box model）

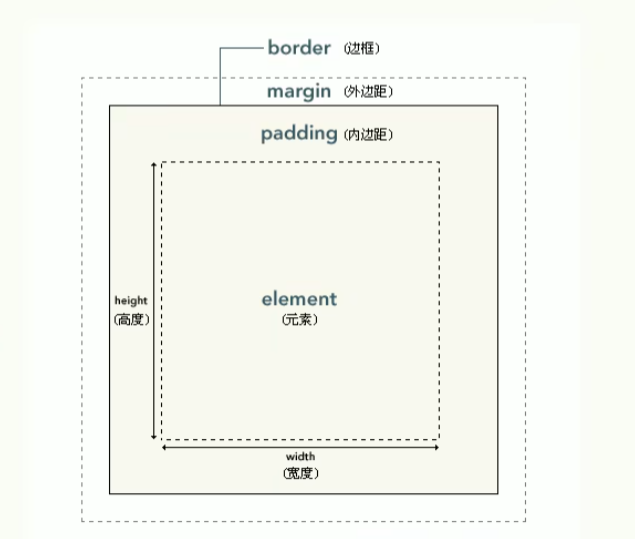
盒子由以下部分组成：content（内容区），padding（内容和边框之间的距离），border（边框），margin（边框和外部之间的距离）

width和height设置的是内容区的大小，border-width设置的是边框的宽度。border-width可以设置4个值（上 右 下 左），3个值（上 左右 下），2个值（上下 左右）。

padding设置内边距，会影响盒子大小，背景颜色也会延伸到内边距上。

一个盒子的大小=内容区+内边距+边框

margin设置外边距，设置左和上外边距会移动自身，设置下和右外边距会移动相邻的其他元素。



### 水平方向布局

元素的水平方向布局：元素在父元素中水平方向的位置由以下几个属性决定

margin-left；border-left；padding-left；width；padding-right；border-right；margin-right

将任意属性设置为auto，意思就是系统会自动给auto的属性增值使得子类总宽度=父类宽度，如果没有auto，系统会自动调整margin-right（所以默认的元素都是靠左）

如果子类宽度大于父类宽度，比如子类1000px父类800px，那么系统会自动把margin-right设置为-200px（视觉上溢出）

想要居中的话：

margin-left+border-left+padding-left+width+padding-right+border-right+margin-right

将margin-left和margin-right设置为0，或margin：0 auto （上下是0，左右auto）

### 垂直方向布局

子类超过父类高度会向下溢出，可以通过overflow来调整。overflow默认值为visible，默认溢出，可以调整为hidden（剪裁），scroll（生成滚动条），auto（根据需要生成滚动条）。

### 行内元素转块元素

行内元素例如<a>不能设置高宽，想要设置高宽只能使用display来设置元素显示类型，例如display: inline。（特点：既可以设置高宽又不会独占一行，兼具了两种元素的特点）

### 盒子大小

box-sizing: content-box 默认值，宽度高度是内容区大小（其他额外还要加上padding和border）；border-box 宽度和高度设置整个盒子大小（padding大的话会压缩内容区大小，整体宽高不变）。

## 浮动

通过浮动能让盒子向左或者向右，设置浮动以后会从文档流脱离，让box变成横向排列。可以用浮动来实现文字环绕图片（浮动元素不会盖住文字）

## 定位

position：static默认的元素是静止没有开启定位；relative开启元素相对定位；absolute开启元素绝对定位；fixed开启元素固定定位；sticky开启元素粘滞定位

### relative相对元素定位

相对定位参照的是元素在文档流中的位置进行移动，相对定位不会改变元素性质（block还是block，inline还是inline）只有相对元素定位可以设置top，bottom，left，right

### absolute绝对定位

设置为absolute后，如果不设置偏移元素位置不会发生变化，开启绝对定位后会从文档流中脱离，而且会改变元素性质（inline会变成block，block会被内容撑开），absolute会使元素提升一个层级。绝对定位元素相对其包含块定义（包含块就是当前元素最近的block元素），absolute的包含块：离当前最近的开启了定位的祖先元素（开启relative，absolute，fixed等等），如果所有祖先元素都没开启定位，那么就以root定义位置。

### fixed固定定位（也是一种绝对定位）

fixed属于一种特殊的绝对定位，大部分属性都和absolute相似。唯一的不同：位置永远参考浏览器的视口进行定位（absolute还会参考开启了定位的祖先block，但fixed不会，fixed只会参考浏览器定位）。适合制作那种随着滚动条动的header。

### sticky粘滞定位

比较新的定位，因为很新所以兼容性不太好。

### 元素层级

xy轴分别对应元素的高宽，z轴对应的是层级的优先级，例如两个block重叠，block1的z-index: 1; block2的z-index: 2；那么block2就会盖在block1上因为拥有更大的z轴。（如果都不设置，那么最后定义的元素显示在最上层；祖先元素z index再高也不会盖住子元素）