

# Html+CSS 第一节课的实验课

## 参考资料

菜鸟教程网址: [HTML5 教程 | 菜鸟教程 \(runoob.com\)](https://www.runoob.com/html/html5-tutorial.html)

相关课程参考博主: B 站 up 主: 黑马 pink 讲前端

[黑马 pink 讲前端博客-黑马 pink 讲前端专栏文章-文集-哔哩哔哩视频 \(bilibili.com\)](https://www.bilibili.com/video/BV184411y79Y/)

相关教程: [黑马程序员 pink 老师前端入门教程, 零基础必看的 h5\(html5\)+css3+移动端前端视频教程 哔哩哔哩 bilibili](https://www.bilibili.com/video/BV184411y79Y/)

## 一、 环境配置 (html 文件)

### 1.1 web 端构成

其实每一个网页文件就是 html 文件, 但我们所看到的网页一般都是由 html、css 和 javascript 封装而成。三者缺一不可。

主要包括**结构 (Structure)**、**表现 (Presentation)** 和**行为 (Behavior)** 三个方面。

标准	说明
结构	结构用于对 <b>网页元素</b> 进行整理和分类, 现阶段主要学的是HTML。
表现	表现用于设置网页元素的版式、颜色、大小等 <b>外观样式</b> , 主要指的是CSS
行为	行为是指网页模型的定义及 <b>交互</b> 的编写, 现阶段主要学的是 Javascript

### 1.2 html 介绍

本节课主要学习 html+css, 这是一个网页示例。他可以用 txt 写完, 然改变格式为 html 文件即可, 例子如下:

Step1: 复制下面这段文字到 txt 文件保存后, 修改文件后缀名成.html 看看效果, 并打开该网页

```
<h1>感谢大家参加第三次实验课</h1>
```

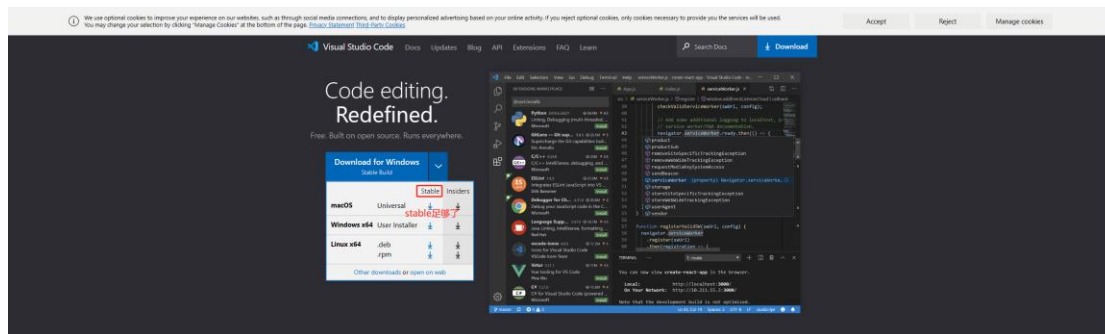




### 1.3 软件配置

其实需要的软件很少，用 txt 改扩展名也能写，但是为了方便各位程序员写代码，我们肯定不可以一直用记事本写代码吧，因此推荐使用 `vscode`，然后打开的浏览器用哪个都可以，推荐默认设置成谷歌。

下载地址：[Visual Studio Code - Code Editing. Redefined](https://code.visualstudio.com/)

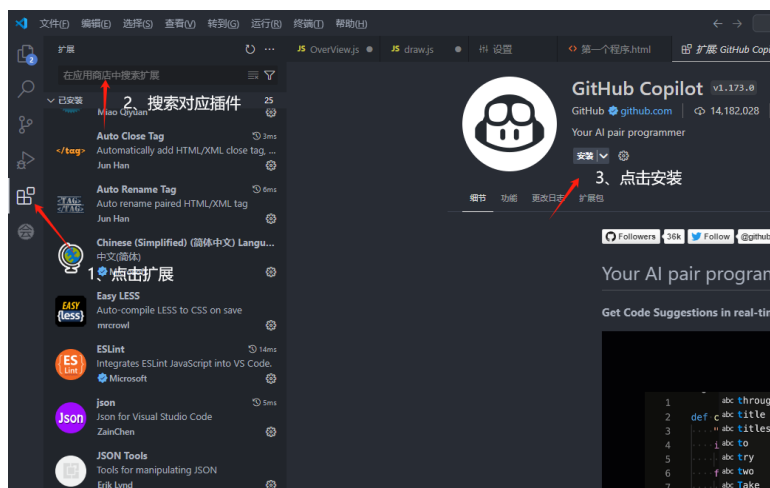


## VSCode 的使用

1. 双击打开软件。
2. 新建文件 ( Ctrl + N ) 。
3. 保存 ( Ctrl + S ) , 注意移动要保存为 .html 文件
4. Ctrl + 加号键 , Ctrl + 减号键 可以放大缩小视图
5. 生成页面骨架结构。  
输入! 按下 Tab 键。
6. 利用插件在浏览器中预览页面：单击鼠标右键，在弹出窗口中点击 “Open In Default Browser” 。

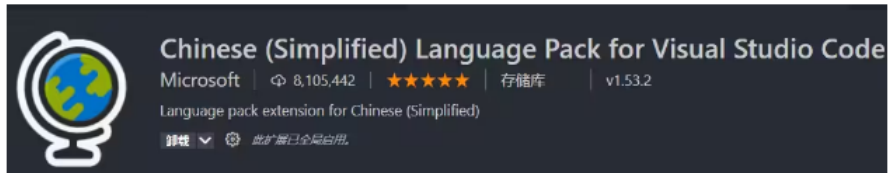
那么为了大家可以使用起来更方便给大家推荐安装如下几个插件：

1. Chinese (Simplified) 汉化
2. Auto Rename Tag
3. One Dark Pro 颜色主题
4. 格式化代码 (vscode 系统自带)
5. open in browser 浏览器预览页面
6. Live Server 实时预览 (推荐)
7. vscode-icons 设置文件图标主题



## 1.Chinese (Simplified)

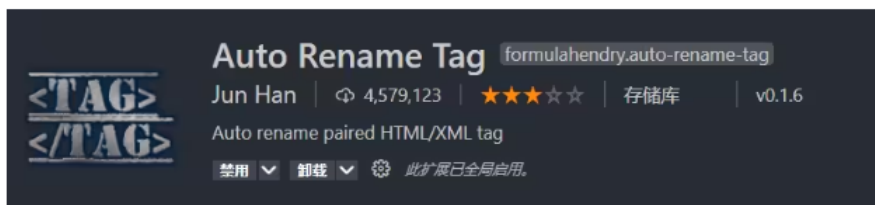
vscode下载完毕是英文版的，先安装这个插件，改为中文版，所以是我们第一个安装的插件。



中文插件chinese

## 2.Auto Rename Tag

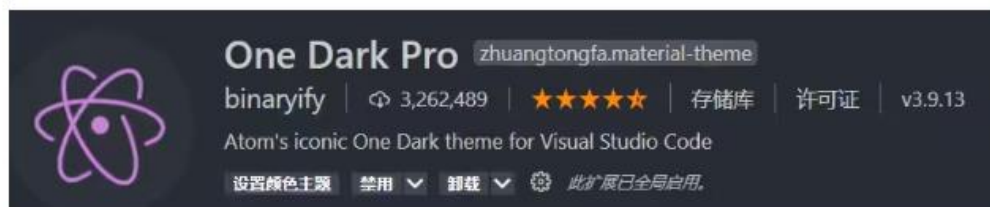
修改开始标签，结束标签跟着自动变化，比较好用。



auto rename tag

## 3.One Dark Pro 颜色主题

我比较喜欢的一个颜色主题

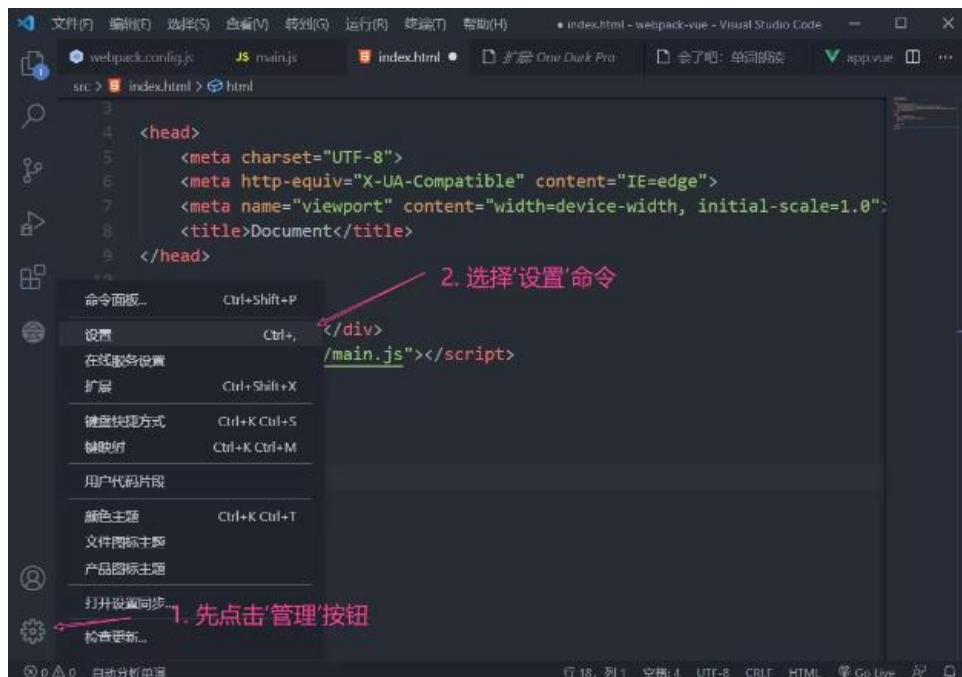


页面的效果：

```
import React from 'react'

export default class App extends React.Component {
  constructor() {
    super()
    this.state = {
      count: 0
    }
    this.handleClick = this.handleClick.bind(this)
  }
  handleClick() {
    this.setState(preState => {
      return {
        count: preState.count + 1
      }
    })
  }
}
```

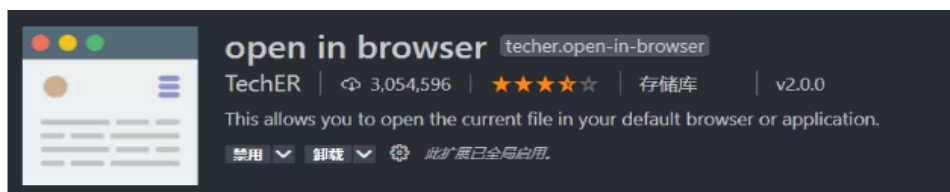
#### 4. 格式化代码 (vscode 系统自带)



(2) 选择用户 --> 文本编辑器 --> 正在格式化 --> 勾选如下图的2个按钮



#### 5. open in browser 浏览器预览页面

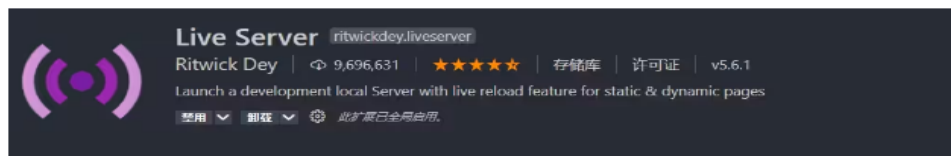


浏览器预览页面



## 6.Live Server 实时预览

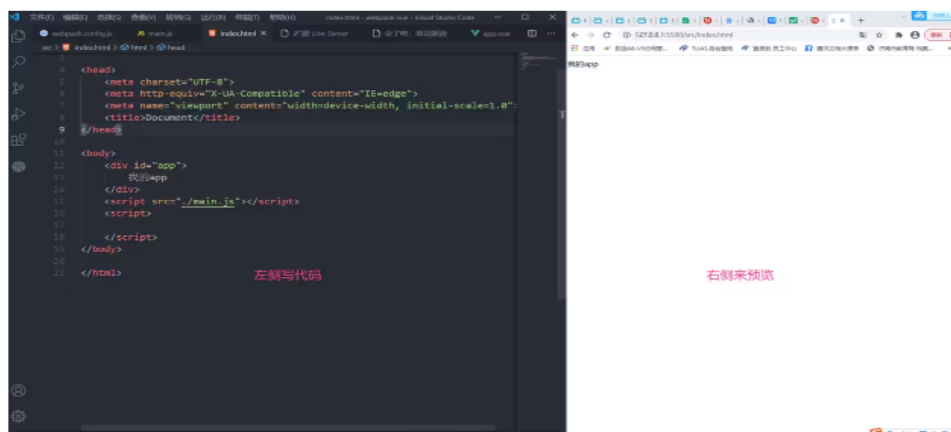
写完代码要启动并刷新网页才可以看到新的效果，但是这个插件可以自动更新保存的代码：



live server 自动更新页面

### 小技巧：

刚开始学习前端，基本页面很少，可以左边编辑器，右边浏览器，这样编写完毕代码，保存完毕后，浏览器自动更新了。



## 1.4 html 基本语句

### 1.4.1 文档类型声明

<!DOCTYPE> 文档类型声明，作用就是告诉浏览器使用哪种HTML版本来显示网页。

```
<!DOCTYPE html>
```

这句代码的意思是：当前页面采取的是 HTML5 版本来显示网页。

注意：

1. <!DOCTYPE> 声明位于文档中的最前面的位置，处于 <html> 标签之前。
2. <!DOCTYPE> 不是一个 HTML 标签，它就是 文档类型声明标签。

在 vscode 中直接感叹号！+ tab 就会自动生成了，具体操作如下

Step1:

先找个地方生成一个文件夹：



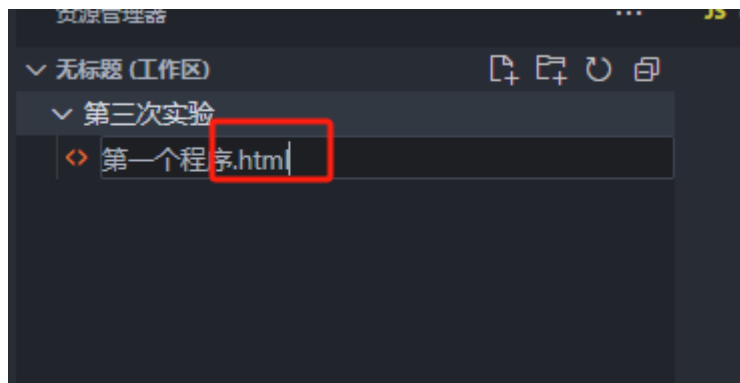
Step2:

然后打开这个文件夹：

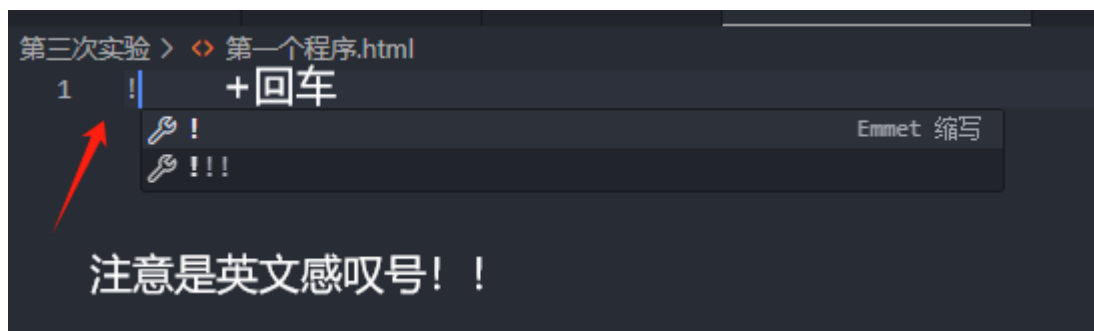




3、新建文件并记得加后缀.html



4、在该文件下输入感叹号并回车



### 1.4.2 lang 语言种类

用来定义当前文档显示的语言。

1. en定义语言为英语
2. zh-CN定义语言为中文

简单来说,定义为en 就是英文网页, 定义为 zh-CN 就是中文网页

其实对于文档显示来说, 定义成en的文档也可以显示中文, 定义成zh-CN的文档也可以显示英文

这个属性对浏览器和搜索引擎(百度.谷歌等)还是有作用的



### 1.4.3 字符集

字符集 (Character set)是多个字符的集合。以便计算机能够识别和存储各种文字。

在<head>标签内，可以通过<meta> 标签的 **charset** 属性来规定 HTML 文档应该使用哪种字符编码。

```
<meta charset="UTF-8" />
```

charset 常用的值有：GB2312、BIG5、GBK 和 UTF-8，其中 **UTF-8** 也被称为**万国码**，基本包含了全世界所有国家需要用到的字符。

**注意：**上面语法是必须要写的代码，否则可能引起乱码的情况。一般情况下，统一使用“UTF-8”编码，尽量统一写成标准的“UTF-8”，不要写成“utf8”或“UTF8”。

### 1.4.4 总结

1. 以上三个代码 vscode 自动生成,基本不需要我们重写.
2. <!DOCTYPE html> 文档类型声明标签,告诉浏览器这个页面采取html5版本来显示页面.
3. <html lang= "en" > 告诉浏览器或者搜索引擎这是一个英文网站. 本页面采取英文来显示.
4. <meta charset=" UTF-8" /> 必须写. 采取 UTF-8来保存文字. 如果不写就会乱码. 具体原理后面分析.

## 1.5 html 基础语法

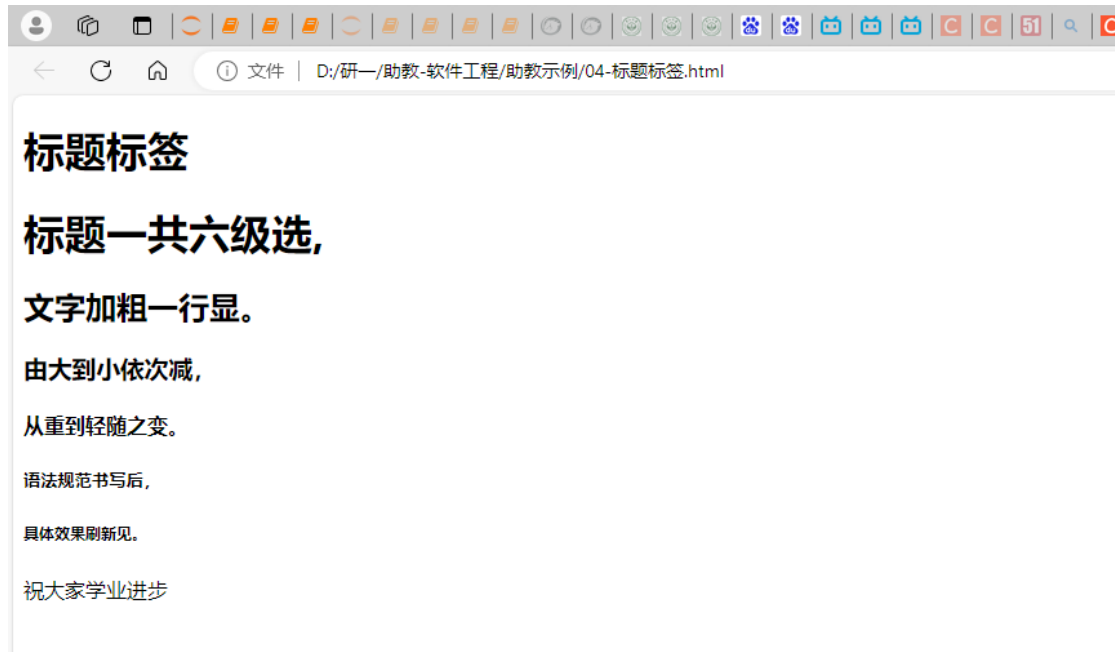
### 1.5.1 标题标签

有 h1 到 h6，意思为主标题、副标题等等等等

大家可以根据下面的代码自己尝试

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>Document</title>
</head>
<body>
  <h1>标题标签</h1>
  <h1>标题一共六级选,</h1>
  <h2>文字加粗一行显.</h2>
  <h3>由大到小依次减,</h3>
  <h4>从重到轻随之变.</h4>
  <h5>语法规范书写后,</h5>
  <h6>具体效果刷新见.</h6>
</body>
</html>
```

效果如下：



### 1.5.2 段落和换行标签

在网页中，要把文字有条理地显示出来，就需要将这些文字分段显示。在 HTML 标签中，`<p>` 标签用于定义段落，它可以将整个网页分为若干个段落。

```
<p> 我是一个段落标签 </p>
```

单词 paragraph ['pærəgræf] 的缩写，意为段落。

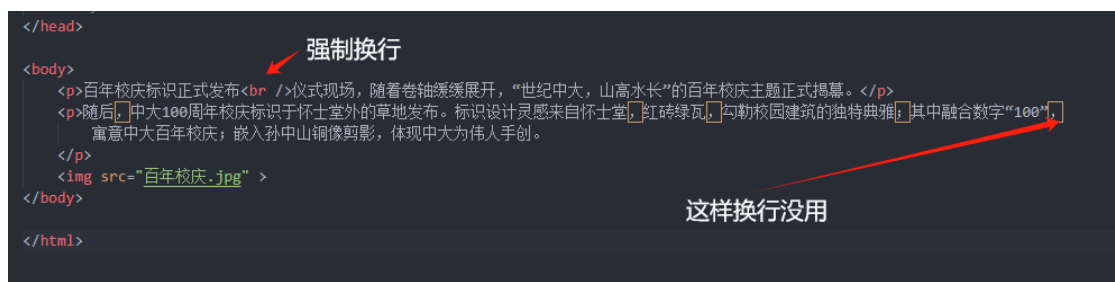
**标签语义：**可以把 HTML 文档分割为若干段落。

在 HTML 中，一个段落中的文字会从左到右依次排列，直到浏览器窗口的右端，然后才自动换行。如果希望某段文本强制换行显示，就需要使用换行标签 `<br />`。

```
<br />
```

单词 break 的缩写，意为打断、换行。

**标签语义：**强制换行。



效果图：



现在字体很小，我们在后续的 css 中可以进行调整，下面是字体放大后的效果。这是 css 的用法，先卖个关子，后面会说。





### 1.5.3 文本格式化标签

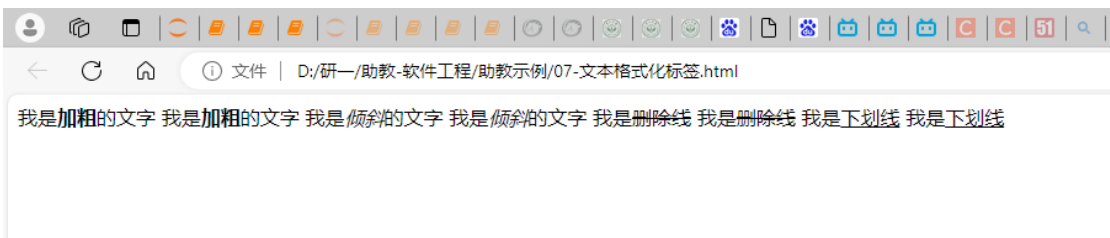
在网页中，有时需要为文字设置**粗体**、*斜体*或下划线等效果，这时就需要用到 HTML 中的文本格式化标签，使文字以特殊的方式显示。

标签语义：突出重要性，比普通文字更重要。

语义	标签	说明
加粗	<code>&lt;strong&gt;&lt;/strong&gt;</code> 或者 <code>&lt;b&gt;&lt;/b&gt;</code>	更推荐使用 <code>&lt;strong&gt;</code> 标签加粗 语义更强烈
倾斜	<code>&lt;em&gt;&lt;/em&gt;</code> 或者 <code>&lt;i&gt;&lt;/i&gt;</code>	更推荐使用 <code>&lt;em&gt;</code> 标签加粗 语义更强烈
删除线	<code>&lt;del&gt;&lt;/del&gt;</code> 或者 <code>&lt;s&gt;&lt;/s&gt;</code>	更推荐使用 <code>&lt;del&gt;</code> 标签加粗 语义更强烈
下划线	<code>&lt;ins&gt;&lt;/ins&gt;</code> 或者 <code>&lt;u&gt;&lt;/u&gt;</code>	更推荐使用 <code>&lt;ins&gt;</code> 标签加粗 语义更强烈

```
<body>
  我是<strong>加粗</strong>的文字
  我是<b>加粗</b>的文字
  我是<em>倾斜</em>的文字
  我是<i>倾斜</i>的文字
  我是<del>删除线</del>
  我是<s>删除线</s>
  我是<ins>下划线</ins>
  我是<u>下划线</u>
</body>
</html>
```

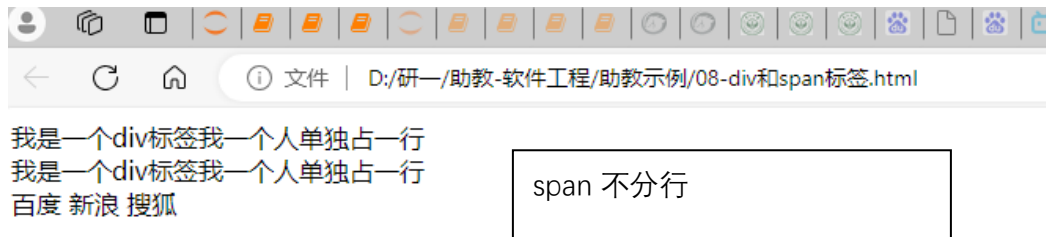
效果图如下：



### 1.5.4 div 和 span 标签

`<div>` 和 `<span>` 是没有语义的，它们就是一个盒子，用来装内容的。

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>div和span标签</title>
</head>
<body>
  <div>我是一个div标签我一个人单独占一行</div>
  <div>我是一个div标签我一个人单独占一行</div>
  <span>百度</span>
  <span>新浪</span>
  <span>搜狐</span>
</body>
</html>
```



### 1.5.5 图像标签

在 HTML 标签中，<img> 标签用于定义 HTML 页面中的图像。

```

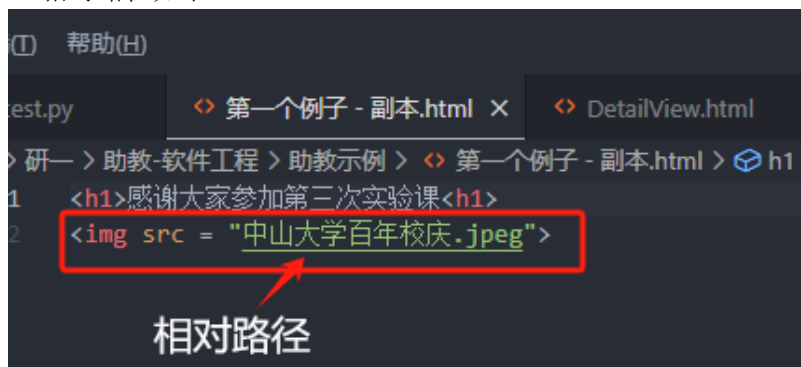
```

单词 image 的缩写，意为图像。

src 是 <img> 标签的**必须属性**，它用于**指定图像文件的路径和文件名**。

所谓属性：简单理解就是属于这个图像标签的特性。

路径是类似于 python 的路径，包括绝对路径和相对路径都可以使用  
相对路径如下：



效果图：



## 1.5.6 图像标签

### 1. 图像标签

图像标签的其他属性：

属性	属性值	说明
src	图片路径	必须属性
alt	文本	替换文本。图像不能显示的文字
title	文本	提示文本。鼠标放到图像上，显示的文字
width	像素	设置图像的宽度
height	像素	设置图像的高度
border	像素	设置图像的边框粗细

具体代码这样执行：

```
<body>
<h4> 图像标签的使用:</h4>
<img src = "百年校庆.jpg"/>
<h4> alt 替换文本 图像显示不出来的时候用文字替换:</h4>

<h4> title 提示文本 鼠标放到图像上,提示的文字:</h4>
<img src = "百年校庆.jpg" title="祝中大百生日快乐" alt="“博学、审问、慎思、明辨、笃行”" />

<h4> width 给图像设定宽度:</h4>
<img src = "百年校庆.jpg" title="祝中大百生日快乐" alt="“博学、审问、慎思、明辨、笃行”" width="500" />
<h4> height 给图像设定高度:</h4>
<img src = "百年校庆.jpg" title="祝中大百生日快乐" alt="“博学、审问、慎思、明辨、笃行”" height="100" />
<h4> border 给图像设定边框:</h4>
<img src = "百年校庆.jpg" title="祝中大百生日快乐" alt="“博学、审问、慎思、明辨、笃行”" width="500" border="15" />
</body>
```

图像标签的使用:



alt 替换文本 图像显示不出来的时候用文字替换:

“博学、审问、慎思、明辨、笃行”

剩下的效果可以自己实验看看

title 提示文本 鼠标放到图像上提示的文字



width 给图像设定宽度:



height 给图像设定高度:



border 给图像设定边框:



## 1.5.7 超链接标签

### 1. 链接的语法格式

```
<a href="跳转目标" target="目标窗口的弹出方式"> 文本或图像 </a>
```

单词 anchor [ˈæŋkə(r)] 的缩写，意为：锚。

两个属性的作用如下：

属性	作用
href	用于指定链接目标的url地址，（必须属性）当为标签应用href属性时，它就具有了超链接的功能
target	用于指定链接页面的打开方式，其中_self为默认值，__blank为在新窗口中打开方式。



## 2.链接分类：

1. 外部链接: 例如 `<a href="http://www.baidu.com"> 百度</a>`。
2. 内部链接: 网站内部页面之间的相互链接. 直接链接内部页面名称即可, 例如 `<a href="index.html"> 首页 </a>`。
3. 空链接: 如果当时没有确定链接目标时, `<a href="#"> 首页 </a>`。
4. 下载链接: 如果 href 里面地址是一个文件或者压缩包, 会下载这个文件。
5. 网页元素链接: 在网页中的各种网页元素, 如文本、图像、表格、音频、视频等都可以添加超链接。
6. 锚点链接: 点我们点击链接可以快速定位到页面中的某个位置。
  - 在链接文本的 href 属性中, 设置属性值为 **#名字** 的形式, 如 `<a href="#two"> 第2集 </a>`
  - 找到目标位置标签, 里面添加一个 id 属性 = 刚才的名字, 如: `<h3 id="two">第2集介绍</h3>`

具体代码执行如下：

```
<body>
  <a href="http://www.qq.com"> 腾讯</a>
  target 打开窗口的方式 默认的值是 _self 当前窗口打开页面 _blank 新窗口打开页面
  <a href="http://www.itcast.cn" target="_blank">传智播客</a>
  <h4>2.内部链接：网站内部页面之间的相互链接</h4>
  <a href="gongsijianjie.html" target="_blank">公司简介</a>
  <h4>3.空链接：#</h4>
  <a href="#">公司地址</a>
  <h4>4.下载链接：地址链接的是 文件 .exe 或者是 zip 等压缩包形式</h4>
  <a href="img.zip">下载文件</a>
  <h4>5.网页元素的链接</h4>
  <a href="https://www.sysu.edu.cn/news/info/2161/1071531.htm"></a>
</body>
```

腾讯 target 打开窗口的方式 默认的值是 \_self 当前窗口打开页面 \_blank 新窗口打开页面 传智播客

2.内部链接: 网站内部页面之间的相互链接

[公司简介](#)

3.空链接: #

[公司地址](#)

4.下载链接: 地址链接的是 文件 .exe 或者是 zip 等压缩包形式

[下载文件](#)

5.网页元素的链接



点击图片会跳到别的链接

1.5.8 锚点链接 （需要提交这一步的文件）

锚点链接: 点我们点击链接,可以快速定位到页面中的某个位置.

- 在链接文本的 href 属性中, 设置属性值为 #名字 的形式, 如<a href="#two"> 第 2 集</a>
- 找到目标位置标签, 里面添加一个 id 属性 = 刚才的名字 , 如: <h3 id="two">第 2 集介绍</h3>
- 在 body 中选择某位置设置锚点, 添加 id.

Step1: 打开发送给你们的文件: 锚点链接.html

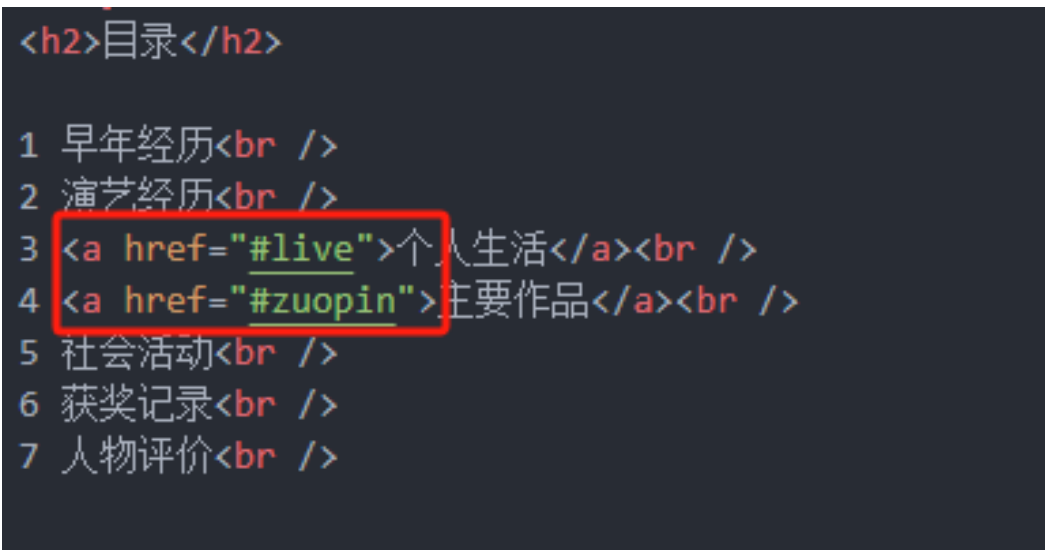
Step2:

在下图所示的地方 h3 标签加入 id="live" 和 id="zuopin"



Step3: 在需要加超链接的位置添加超链接, 用 “#” +id 名称

在下图所示的位置加入 href="#live" 和 href="#zuopin"





Step2: 开始先写基本元素（大标题 h1 和目录）

目录就不用带标签先了

```
<h1> 圣诞节的那些事</h1>
1. 圣诞是怎样由来的 <br>
2.<a href="#oldman">圣诞老人的由来</a> <br>
3.<a href="#tree">圣诞树的由来</a> <br>
```

效果如下：

# 圣诞节的那些事

- 1. 圣诞是怎样由来的
- 2. [圣诞老人的由来](#)
- 3. [圣诞树的由来](#)

Step3: 副标题

我们有三小主题，因此有三个副标题

```
<h2> 圣诞老人的由来</h2>
<h2> 圣诞树的由来</h2>
<h2> 圣诞树的由来</h2>
```

效果如下：

# 圣诞节的那些事

- 1. 圣诞是怎样由来的
- 2. [圣诞老人的由来](#)
- 3. [圣诞树的由来](#)

## 圣诞是怎样由来的

## 圣诞老人的由来

## 圣诞树的由来

Step4:

文字元素，对每个标题的详细介绍，放在副标题标签下面：

```
<h2> 圣诞是怎样由来的</h2>
    <p> 圣诞节是基督教世界最大的节日。一般认为 12 月 25 日作为圣诞节可能开始于公元 336 年的罗马教会。这一天又是罗马历书的冬至节，意味着万物复苏的开始。可能由于这个原因，罗马教会才选择这一天作为圣诞节。后来，因为各地教会使用的历书不同，具体日期不能统一，于是就把 12 月 24 日到第二年的 1 月 6 日定为圣诞节节期（Christmas Tide），各地教会可以根据当地具体情况在这段节期之内庆祝圣诞节。在欧美许多国家里，人们非常重视这个节日，把它和新年连在一起，而庆祝活动之热闹与隆重大大超过了新年，成为一个全民的节日。12 月 24 日平安夜、12 月 25 日圣诞节的主要纪念活动都与耶稣降生的传说有关 。</p>
```

效果如下：

圣诞节的那些事

- 1.圣诞节是怎样由来的
- 2.圣诞老人是由谁
- 3.圣诞树的由来

圣诞是怎样由来的

圣诞节是基督教世界最大的节日，一般认为12月25日作为圣诞节可能开始于公元336年的罗马教会，4世纪初，1月6日是罗马帝国东部各教会纪念耶稣降生和受洗的双重节日，即上帝通过耶稣向世人显示自己。这一天又是罗马历书的冬至节，象征着万物复苏的开始。可能由于这个原因，罗马教会便根据当地具体情况在这段时期之内庆祝圣诞节。在欧洲许多国家里，人们非常重视这个节日，把它和新年连在一起，而庆祝活动的热情与隆重大大超过了新年，成为一个全民的节日。12月24日平安夜，12月25日圣诞节的主要纪念活动都与耶稣降生的传说有关。

Step5：剩下的文字一样的方法放在各自的标签下  
完成后效果如下：

圣诞节的那些事

- 1.圣诞节是怎样由来的
- 2.圣诞老人是由谁
- 3.圣诞树的由来

圣诞是怎样由来的

圣诞节是基督教世界最大的节日，一般认为12月25日作为圣诞节可能开始于公元336年的罗马教会，4世纪初，1月6日是罗马帝国东部各教会纪念耶稣降生和受洗的双重节日，即上帝通过耶稣向世人显示自己。这一天又是罗马历书的冬至节，象征着万物复苏的开始。可能由于这个原因，罗马教会才选择这一天作为圣诞节。后来，因为教会会使用圣经中关于牧羊人看见以马忤斯之星之内庆祝圣诞节，在教皇许多国家里，人们非常重视这个节日，把它和新年连在一起，而庆祝活动的热情与隆重大大超过了新年，成为一个全民的节日。12月24日平安夜，12月25日圣诞节的主要纪念活动都与耶稣降生的传说有关。

圣诞老人的由来

圣诞节当然少不了**圣诞老人**，根据圣经记载，书中并没有提起这一号人物。到底他是如何成为圣诞的主角之一呢？

相传在一千六百多年前，荷兰巴里地方有一个老人，名叫尼古拉斯，他一生最爱帮助贫穷的人家，其中有一次他帮助三个贫穷的少女，送给她们三金盒子以度过难关的困境。

到了十八二三年，由荷兰传教士把这位伟大慈善家的故事传到美国，献给圣诞老人并被地流行于世界各属。根据调查，以上故事还有下文：当尼古拉斯准备把其中一便金子送给其中一名女子时，他把金子从其中一个窗户的进去，恰好被睡梦在壁炉上的一只长袜中，于是，将礼物放在圣诞袜子的习俗才方便传延。

在平安夜里，孩子们会把一双双色彩缤纷的袜子挂在床头，然后在袜子旁边放好热牛奶让圣诞老人喝，以让好梦功成圣诞老人顺嘴，并送大礼给自己。？

圣诞树的由来

圣诞一直是庆祝圣诞节不可少的装饰物，如果家中没有圣诞树，就大大减少了过节气氛。关于圣诞树的来源有多种不同的传说。

其中一个是说：大约在十六世纪，基督教徒先出现在德国，德国人把长青的松柏枝拿到屋中去摆设，称之为圣诞树。后来，由德国人马丁路德把圣诞树放在教堂中的松树旁，然后点燃蜡烛，使它看起来像星星引导人们到伯利恒去。而至今仍，人们已经改用彩色的小灯亮了。

另一个传说是：在很久以前，曾有一位农民，在圣诞节那天看到一个奇异的景象，儿童们在树下将一些礼物放在地上，松枝立即变成一棵树，上面挂满礼物，以答谢农民的好意。

圣诞树真正出现在圣诞节时，首先见于德国，之后又传入欧洲和美洲，并以其壮观的标志，成为圣诞节不可缺少的装饰。圣诞树的种类繁多，有天然和人造圣诞树，也有人造圣诞树及白色圣诞树。每棵圣诞树上都挂满琳琅满目的装饰品，但每棵树的顶端必须有个很大的星星，象征三博士跟随星星而找到耶稣，而且也只有该家庭的一家之主可以把这棵更多内容可以[点击查看](#)

Step6：增加图片元素

目前文字已经实现完成，于是增加图片，根据 word 的内容，在对应的位置，引用图片

下面只是其中一个图片放的位置，大家根据 word 的布局，自己把图片放全

```
<p>圣诞节当然少不了<a href="oldman.html" target="_blank">圣诞老人</a>，根据圣经记载，书中并没有提起这一号人物。到底他是如何成为圣诞的主角之一呢？</p>
```

效果如下：

圣诞树的由来

圣诞节当然少不了**圣诞老人**，根据圣经记载，书中并没有提起这一号人物。到底他是如何成为圣诞的主角之一呢？







## 二、CSS 介绍

CSS 的主要使用场景就是美化网页,布局页面的.

虽然 HTML 有局限性 但 CSS 刚好是网页的美容师

### 2.1 HTML

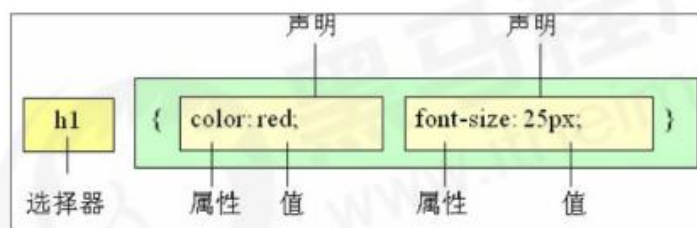
这其实是个非常单纯的家伙,他**只关注内容的语义**。比如 `<h1>` 表明这是一个大标题, `<p>` 表明这是一个段落, `<img>` 表明这儿有一个图片, `<a>` 表示此处有链接。很早的时候,世界上的网站虽然很多,但是他们都有一个共同的特点:**丑**。虽然 HTML 可以做简单的样式,但是带来的是无尽的**臃肿和繁琐**……

### 2.2 CSS

其实也是一种标记语言 CSS 主要用于设置 HTML 页面中的**文本内容**(字体、大小、对齐方式等)、**图片的外形**(宽高、边框样式、边距等)以及**版面的布局**和**外观显示样式**。CSS 让我们的网页更加丰富多彩,布局更加灵活自如。简单理解:**CSS 可以美化 HTML , 让 HTML 更漂亮, 让页面布局更简单。**

### 2.3 选择器和声明

CSS 规则由两个主要的部分构成: **选择器**以及**一条或多条声明**。



- **选择器**是用于指定 CSS 样式的 **HTML 标签**, 花括号内是对该对象设置的具体样式
- 属性和属性值以“键值对”的形式出现
- 属性是对指定的对象设置的样式属性, 例如字体大小、文本颜色等
- 属性和属性值之间用英文 “:” 分开
- 多个“键值对”之间用英文 “;” 进行区分

### 2.4 css 语法规范

所有的样式, 都包含在 `<style>` 标签内, 表示是样式表。`<style>` 一般写到 `</head>` 里面。

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>
    body {
      font: 16px/28px 'Microsoft YaHei';
      /* 将body里面的文字全部变成 16px/28px 和微软雅黑 */
    }
  </style>
</head>
<body>
```



## 2.5 选择器的分类

### 2.5.1 标签选择器

**标签选择器**（元素选择器）是指用 **HTML 标签名称** 作为选择器，按标签名称分类，为页面中某一类标签指定统一的 CSS 样式。

#### 作用

标签选择器可以把某一类标签全部选择出来，比如所有的 `<div>` 标签和所有的 `<span>` 标签。

#### 优点

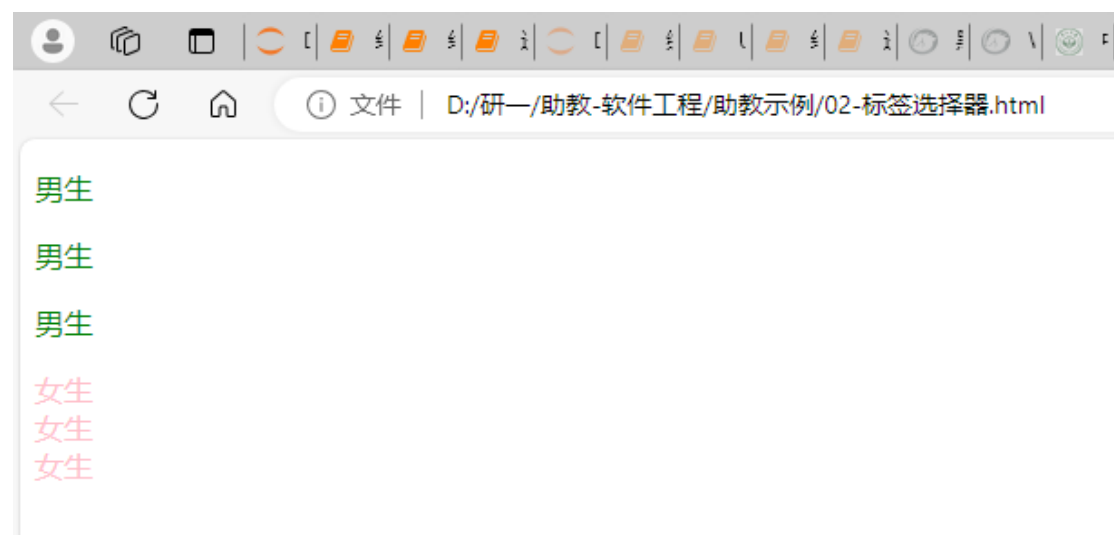
能快速为页面中同类型的标签统一设置样式。

#### 缺点

不能设计差异化样式，只能选择全部当前标签

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <meta http-equiv="X-UA-Compatible" content="ie=edge">
  <title>基础选择器之标签选择器</title>
  <style>
    /* 标签选择器：写上标签名 */
    p {
      color: green;
    }
    div {
      color: pink;
    }
  </style>
</head>
<body>
  <p>男生</p>
  <p>男生</p>
  <p>男生</p>
  <div>女生</div>
  <div>女生</div>
  <div>女生</div>
</body>
</html>
```

只能控制整一类p标签或div标签



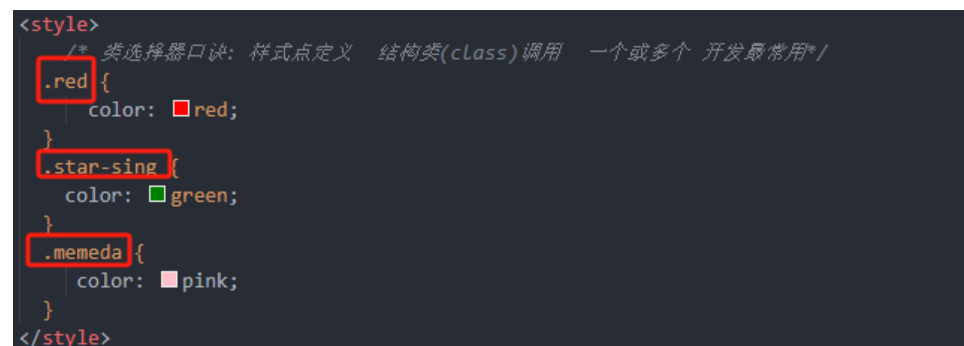
### 2.5.2 类选择器

如果想要差异化选择不同的标签，单独选一个或者某几个标签，可以使用类选择器。

首先，标签中要定义类：结构需要用 class 属性来调用 class 类的意思



其次在 css 中对类进行声明：




注意，是要加符号“.”再加上类名。

### 2.5.3 多类名



和前面一样，但是用空格隔开两个类名。

在 css 代码中就可以分别声明每个类。

```
.box {
  width: 150px;
  height: 100px;
  font-size: 30px;
}
.red {
  /* 背景颜色 */
  background-color:  red;
}
.green {
  background-color:  green;
}
```


#### 2.5.4 id 选择器

id 选择器和类选择器的区别

- ① 类选择器（class）好比人的名字，一个人可以有多个名字，同时一个名字也可以被多个人使用。
- ② id 选择器好比人的身份证号码，全中国是唯一的，不得重复。
- ③ id 选择器和类选择器最大的不同在于使用次数上。
- ④ 类选择器在修改样式中用的最多，id 选择器一般用于页面唯一性的元素上，经常和 JavaScript 搭配使用。

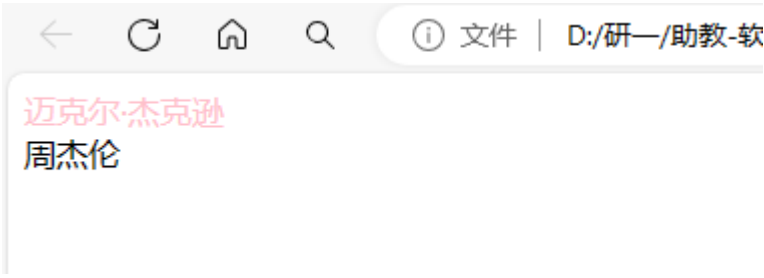
```
<body>
  <div id="pink">迈克尔·杰克逊</div>
  <div>周杰伦</div>
</body>
```

在 css 中调用，语法如下：

```
<style>
  /* id选择器的口诀：样式#定义，结构id调用，只能调用一次，别人切勿使用 */
  #pink {
    color:  pink;
  }
</style>
```

注意和类 class 不同这是符号“#”+id 名称。

上面的代码运行结果如下：



## 2.6 练习

根据 2.6.html 文件，结合上面的内容增加 css 部分，目前的文件已经写了如下：

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <style>

  </style>
</head>
<body>
  <p>男生</p>
  <p>男生</p>
  <div>女生</div>
  <div>女生</div>
  以上是改标签选择器
  <li class="red">来生缘</li>
  <li class="memeda">生僻字</li>
  <li class="star-sing">江南style</li>
  以上是改类选择器
  <div id="pink">迈克尔·杰克逊</div>
  以上是改id选择器

</body>
</html>
```

## 2.7 文本属性

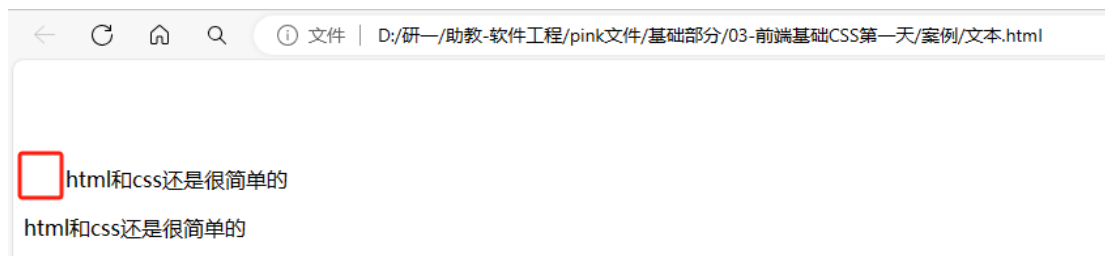
在这一块，文本属性是可以选择他的字体（微软雅黑）、字体大小（多少 px）、字体颜色、对齐文本（居中等）、文本缩进（开头空两格）

属性	表示	注意点
color	文本颜色	我们通常用 十六进制 比如 而且是简写形式 #fff
text-align	文本对齐	可以设定文字水平的对齐方式
text-indent	文本缩进	通常我们用于段落首行缩进2个字的距离 text-indent: 2em;
text-decoration	文本修饰	记住 添加下划线 underline 取消下划线 none
line-height	行高	控制行与行之间的距离

### 2.7.1 文本缩进

```
<style>
  p {
    /* 首行缩进2个字的距离 */
    text-indent: 2em;
  }
</style>
</head>
<body>
  <p>html和css还是很简单的</p>
  <div>html和css还是很简单的</div>
```

上面的代码给 p 标签定义了 2 个缩进，em 表示缩进的字数，div 没定义，具体效果如下



### 2.7.2 文本粗细、居中对齐、颜色、大小

下面的代码给 h1 标签加了几个定义。

```
<title>Document</title>
<style>
  h1 {
    /* 文字不加粗 */
    font-weight: 400;
    /* 让h1里面的文字水平居中对齐 */
    text-align: center;
    /* 修改颜色 */
    color: #eb0c0c;
    font-size: 30px;
  }
</style>
</head>
<body>
  <h1>祝大家第三次实验顺利完成</h1>
</body>
```

效果如下图：



那么到这里，我们第一次实验课需要讲的内容就差不多了，希望大家先回去消化一下，同时课下可以多去了解 html 和 css 还有很多功能是本节课没有讲到的。大家自己课下学学。

## 本堂实验课作业：用今天讲到的方法制作一页或多页的新闻界面。

具体设计可以考虑：

- 文字部分，注意字体大小，标题、一级标题的合理运用
- 字体颜色、大小等更看重用 css 语言来写而不是放在 html 标签里面写
- 想制作多页的，可以在关键词做超链接（可以设置锚点或者超链接到下一页）
- 图片文字的排版要合理得当
- 尽量使用学过的内容，比如鼠标放在图片上的时候显示文字等

文件（第三周实验课作业.html）中，先给好了开头的部分，大家可以根据这个进行内容填充，题目自己想哈。



上图为简单示例，大家可以发挥想象力，创新和丰富界面。

## 本堂课需要提交的文件：

### 1、1.5.8 锚点链接练习

### 2、1.6 综合案例练习

### 3、新闻界面练习