# CSS 做什么的？

上一阶段我们认识了HTML基本知识，也学会如何用HTML来搭一个网页框架了，但是打开一看，一点都不像个样子啊，这有什么用呢？



的确，HTML作用就此而已，它能做出来的效果就是搭骨架，搭出来的结果就如同一个骨头人，没血没肉，也没穿衣打扮，就真的很难看了。但是骨架是基础，没有骨架，后面的工作都没法做。

可以形象比喻，HTML就像一个普通的画笔，只能画出轮廓，却没有美观，如果要想美观看到效果，必须要用其他工具，这些工具可能有多种，比如水墨，彩笔，蜡笔等等，如果把这些工具做一个统称，那就是CSS。

所以再说简单点，CSS就是用来美化的。

## 官方解释

CSS 全称Cascading Style Sheets 翻译中文 **层叠样式表** 是一种用来表现HTML（标准通用标记语言的一个应用）或XML（标准通用标记语言的一个子集）等文件样式的计算机语言。CSS不仅可以静态地修饰网页，还可以 ..... （后面不用了解）

## 引一个例子

为了证明效果，就快速引一个例子更直观。

首先我们创建一个后缀名为html文件

搭建HTML基本框架后，在body标签内添加一个div标签：

<div class="test"></div>

保存后，打开文件看效果，什么都没有吧？

那么接下来在<head> 标签最后添加一个style标签，输入以下内容，看如下所示：



可以直接复制以下代码：

<!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>

<style>

.test {

width: 100px;

height: 100px;

background-color: pink;

}

</style>

</head>

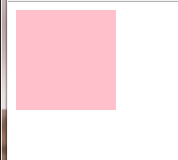
<body>

<div class="test"></div>

</body>

</html>

好了，输入完了，再打开文件看看，会发现有着不一样的效果：



我们的div标签可以具体显示出来了。

这里做了什么呢？和之前有什么不同呢？为何会有那么大的差距？结论可想而知，就是引入了css。但是怎么引入的？引入了哪些东西，客官别急，待下面慢慢揭开面纱。

# CSS 的引入方式

## 标签内引入

每个HTML标签都有一个style属性，这个属性就是用于引入CSS样式的。

  <!-- 标签内style引入 -->

  <div style="width: 100px; height: 100px; background-color: blue;"></div>

## 文件内<style>标签引入

在HTML基本架构中的header标签内可以嵌套一个style属性，这个属性也是引入css 样式的一个标志：

<style>

    /\* style标签内部引入 \*/

  .test {

*width*: 100px;

*height*: 100px;

*background-color*: pink;

  }

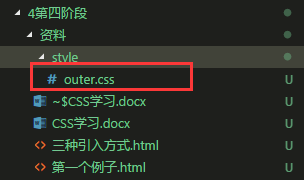
</style>

## 外部文件引入

可以把css 的内容放在一个专有的文件中，这个文件的后缀名为css 放在项目文件中的某个位置，然后再HTML文件中用link标签去引入它，这个标签也是在head标签内嵌套的：

  <!-- 外部文件引入 -->

  <link rel="stylesheet" href="./style/outer.css">



## 总结

第一种首先是不推荐使用的，因为大量的css代码写在标签内部，杂乱不好维护，而且把css和html耦合在一起分不开了。第二种是推荐的，这样就已经把css代码和html代码分开了，第三种是最推荐的，因为它单独把css文件级别隔离了，也是真正项目需要使用的，**但是，但是** 在我们接下来的项目开发中是采用模块化开发的，也就是我们不会一口气把一个页面所有的内容放在一个文件下写，基于模块化开发的思路，我们是需要把css 和 html放在同个文件方便管理和后期整合的，因此在接下来开发过程中可以不需要单独分开css文件引入

# 基础属性和颜色表示法

这里为了接下来介绍方便，先介绍一下一些基本属性的含义：

首先是width 字面翻译就是宽度的意思 px 表示像素单位

height 字面翻译就是高度

background-color 字面翻译就是背景色

综合以上样式的总和，就是我们打开文件看到的效果了，宽100像素，高100像素，背景色粉色。

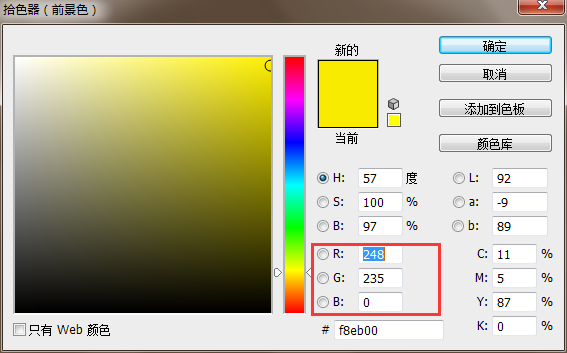
## 颜色表示法

### 英文字母表示法

凡是非英语专业能认识的表示颜色的英文单词均可以使用，诸如 red blue green yellow orange 等，但是一般不常用，因为不够用。

### rgb表示法

在PS中颜色面板就已经有显示每种颜色的红绿蓝的数值：



这时候要想根据PS上的具体颜色在页面上一模一样呈现，就需要用rbg属性

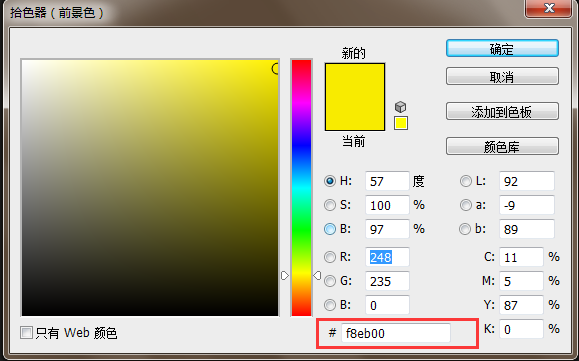
*background-color*: rgb(248, 235, 0);

### 十六进制表示法

上述rgb方法对人认识比较直观，但是对计算机不是很友好（计算机是专注于二进制的）于是我们更多的是把人认识的世界翻译过去，就有了十六进制表示法：

简单介绍一下，就是 r g b 的值大小都是 0 ~ 255 以内，翻译到十六进制数就只需要两位， 00 ~ ff 于是我们每两位代表一种颜色的数值大小，总共就需要六位即可表示所有rgb能表示的颜色了。

在PS颜色选择面板中也有这个具体的数值

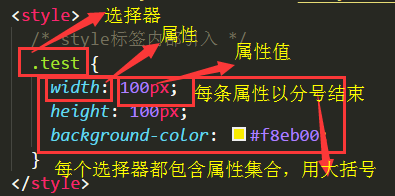


使用十六进制表示时必须签名带上# 在css内就如下书写：

*background-color*: #f8eb00;

# CSS 语法结构

关于语法结构，就用一张图表示



首先是选择器，这里具体内容下一章去了解

语法元素就是 大括号，冒号，分号

这几个分开的，需要掌握的概念，就是属性和属性值了

## 注释

在css中，注释的结构为 /\* \*/ 和大多数编程语言一致

# CSS 选择器

## id 选择器

这个选择器尽量不再css用。

每个HTML标签都有一个id属性，这个属性就是用来命名的，但是这个名字就像身份证一样，每个HTML文件内id 的名字不能重复

在css 中 使用 # 开头加上id 名即可选中该元素。

## class 类选择器

每个HTML标签都有一个class属性，这个属性就是用来命名的，通常就是给css的选择器使用，让css知道接下来的属性集合应该应用于哪个标签下

在css 中 使用. 开头加上class 名即可选中该元素。

## 后代选择器

用于嵌套关系的额HTML标签的选中，HTML结构有嵌套时就有继承关系，就是父子孙等后代关系了，如果我们想要选择孙子元素，通常需要顺着父元素class名到子元素的class名到孙子元素的class名来选中它，这些都是使用class选择器，用空格隔开。

每一代的选择，可以使用任何一种选择器方式，不局限于class选择器

## 复合选择器

就是更为了严格的选中某个元素，避免其他元素冲突。根据我们

### 标签和类名复合

### 多类名复合

## 子选择器

## 结构性伪类选择器

## 兄弟选择器

## 总结

重点掌握二、三种，以下的以后会用到，再来了解

# 经典盒模型

# 常用文本属性

# 学习资料