

CSS层叠样式表

一、什么是CSS

CSS是指层叠样式表（Cascading Style Sheets），简称样式表，它是一种制作网页的新技术，定义如何显示HTML元素。如前所述，样式是对网页元素（网页内容）的显示的形式描述。层叠是指当HTML文件引用多个css文件时，如果css文件之间所定义的样式发生冲突，将依据层次的先后来处理其样式对内容的控制。

1、css的作用：

- 1.内容和样式分离，使网页设计更加简洁，明了，美化界面。
- 2.弥补了HTML对标记属性控制的不足，例如，可以任意设置标题的大小，HTML面积指可以用h1-h6来实现
- 3.精确控制网页布局，如行间距，字间距，段落缩进和图像定位等。
- 4.提高网页效率，多个网页可以同时应用一个css样式，既可以减少代码的下载，也提高浏览器速度和网页的更新速度。

不使用css的效果图：



使用css的效果图：



二、CSS的基本语法

```
选择器{  
  样式规则  
}
```

样式规则：

属性名1：属性值1；

属性名2：属性值2；

.....

选择器：是用来选择标签的，选出来以后给标签加样式。

示例代码：

```
div{  
  width:100px;  
  height:100px;  
  background:gold;  
}
```

说明：css是由两个主要的部分组成，选择器和一条或多条样式规则。

注意：样式规则需要放到大括号里面

三、CSS的引入方式

css引入的三种方式

- 1、行内式
- 2、内嵌式（内部样式）
- 3、外链式

行内式：

直接在标签的style属性中添加css的样式

示例代码：

```
<div style="width:100px; height:100px; background:red">
  hello,你好呀
</div>
```

优点：方便、直观

缺点：缺乏可重用性

内嵌式（内部样式）

在< head >标签内加入< style >标签，在< style >标签中编写css代码

示例代码：

```
<head>
  <style type="text/css">
    h3{
      color:red;
    }
  </style>
</head>
```

优点：在同一个页面内部便于复用和维护。

缺点：在多个页面之间的可重用性不高。

外链式：

将css代码写在一个单独的css文件中，在标签中使用标签直接引入该文件到页面中。

示例代码：

```
<link rel="stylesheet" type="text/css" href="css/main.css">
```

优点：使得css样式与html页面分离，便于整个页面系统的规划和维护，可重用性高。

缺点：css代码由于分离到单独的css文件中，容易出现css代码过于集中，若维护不当则容易造成混乱

css引入方式选择：

- 1、行内式几乎不用
- 2、内嵌式在学习css样式阶段用
- 3、外链式在公司开发的阶段使用，可以对css样式和html页面分别进行开发

四、css的选择器

1、css选择器的定义

css选择器是用来选择标签的，选出来以后给标签加样式

2、css选择器的种类

- 1、标签选择器
- 2、类选择器
- 3、层级选择器
- 4、id选择器
- 5、组选择器
- 6、伪类选择器

3、标签选择器

根据标签来选择标签，以**标签开头**，这种选择器影响范围大，一般用来做一些通用的设置。

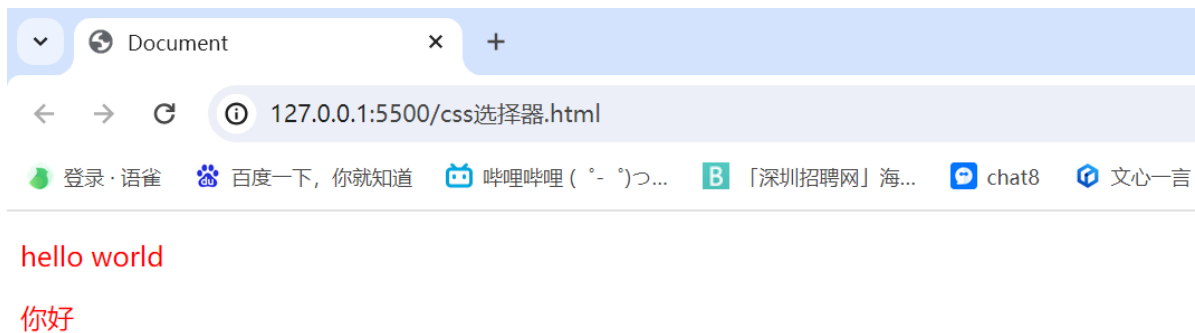
示例代码：

```
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>

  <style>
    p {
      color: red;
    }
  </style>
</head>

<body>
  <!-- 标签选择器，就是以标签名开头，根据标签名来选择html文件中的标签，给标签添加样式 -->
  <p>hello world</p>
  <p>你好</p>
</body>
```

效果图：



4、类选择器

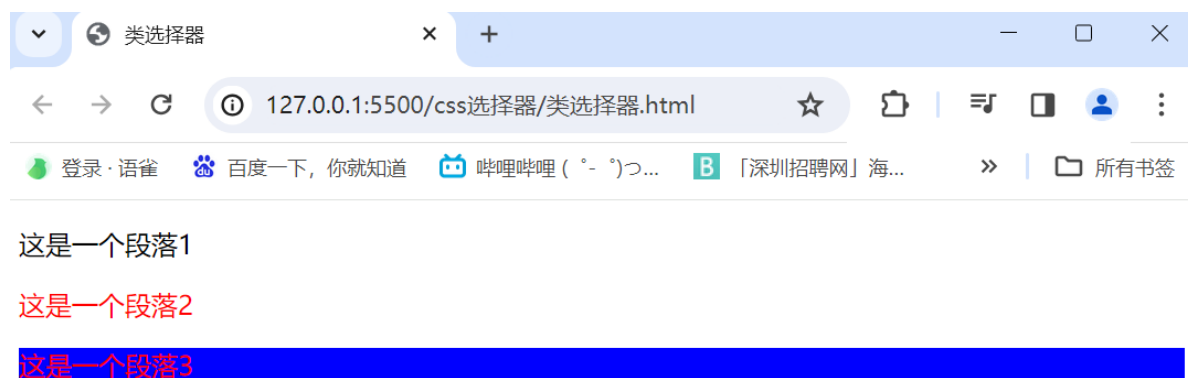
根据类名来选择标签，以**.开头**，一个类选择器可应用于多个标签上，一个标签上也可以使用多个类选择器，多个类选择器需要使用空格分割，应用灵活，可复用，是css中应用最多一个选择器。

示例代码：

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

  <style>
    /* 类选择器，以.开头，根据类名选择标签，给标签添加样式 */
    .duo{
      color: red;
    }
    .do{
      background-color: blue;
    }
  </style>
</head>
<body>
  <p>这是一个段落1</p>
  <p class="duo">这是一个段落2</p>
  <!-- 这里标签使用了两个类选择器 -->
  <p class="duo do">这是一个段落3</p>
</body>
</html>
```

效果图：



5、层级选择器（后代选择器）

根据层级关系选择后代标签，以**选择器1，选择器2**开头，主要应用在标签的嵌套的结构中，减少命名

示例代码：

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

```

<title>层级选择器</title>

<style>
    /* 层级选择器，要有层级关系，根据层级关系获取子标签，给子标签添加样式 */
    div div{
        color: red;
    }
    div .box{
        color: blue;
    }

    /* div div p{
        color: green;
    } */

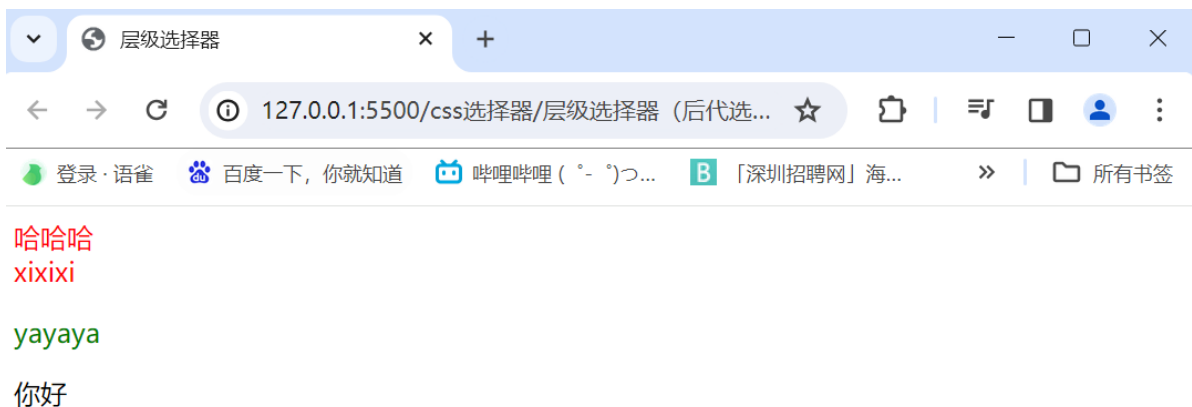
    div p{
        color: green;
    }
</style>
</head>
<body>
    <div>
        <div>哈哈</div>
        <div class="box1">xixixi</div>
        <!-- 层级选择器不一定必须是父子关系，祖孙关系的子标签也可以找到，完成添加样式的操作 -->
    >

    <div>
        <p>yayaya</p>
    </div>
    <div>你好</div>
</body>
</html>

```

注意点：这个层级关系不一定是父子关系，也有可能是祖孙关系，只要有后代关系都适用于这个层级选择器。

效果图：



6、id选择器

根据id选择标签，以#开头，元素的id名称不能重复，所有id选择器只能对应于页面上一个元素，不能复用，id名一般给程序使用，所以不推荐使用id作为选择器。

示例代码：

```

<!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>
    /* id选择器，以#开头，根据id名选择html中的标签，给标签添加样式 */
    #myid1{
      color: red;
    }
  </style>
</head>
<body>
  <p id="myid1">世界你好</p>
  <!-- 错误的实例，因为在html文件中标签的id是唯一的，不能重复，后续js会根据id获取对应的标签对象 -->
  <p id="myid1">css真好玩</p>
</body>
</html>

```

注意点：虽然给其他标签设置id="box"也可以设置样式，但是不推荐这样做，因为id是唯一的，以后js通过id只能获取一个唯一的标签对象。

效果图：



世界你好

css真好玩

7、组选择器

根据组合的选择器选择不同的标签，以分割开，如果有公共的样式设置，可以使用组选择器

示例代码：

```

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

  <style>
    /* 组选择器就是多个选择器的组合，把功能的代码可以放在组选择器中 */
    /* 这里表达的是定义三个类选择器，他们都有相同的样式规则 */
    .box1, .box2, .box3{
      width: 100px;
      height: 100px;
    }
  </style>

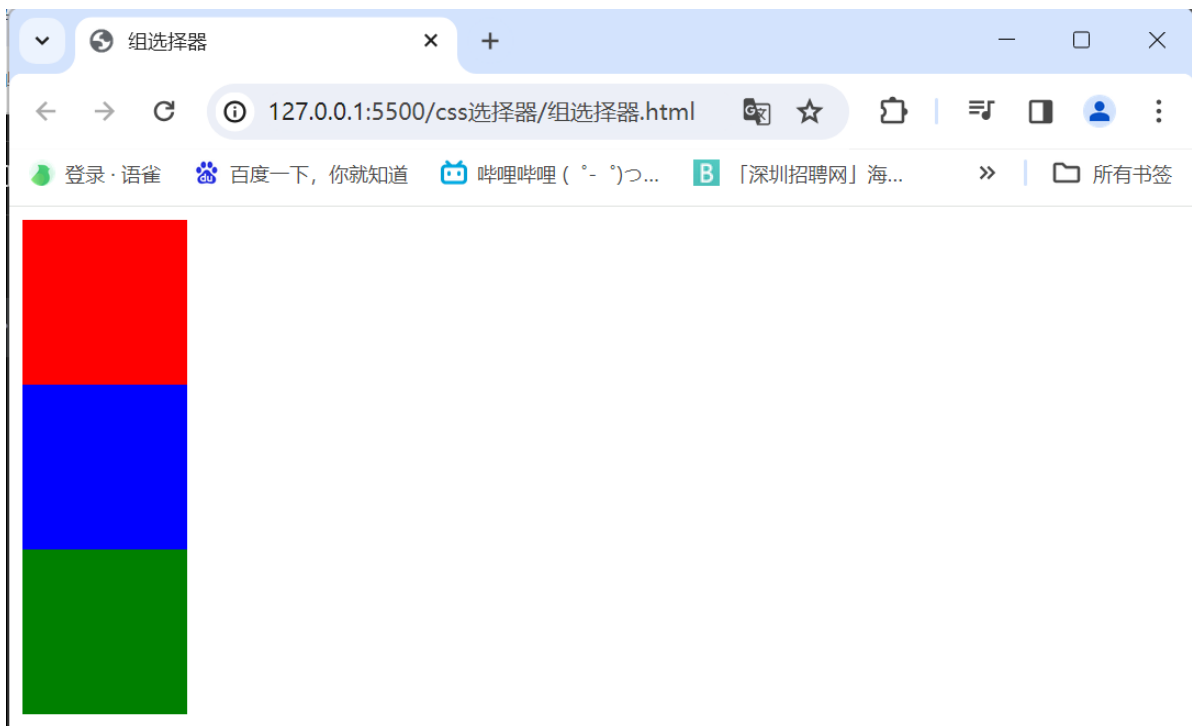
```

```

    }
    /* 在css里面出现同样的选择器，后面的不会覆盖前面的，后面的选择器会在前面选择器的基础上
    进行追加样式 */
    .box1{
        background: red;
    }
    .box2{
        background: blue;
    }
    .box3{
        background: green;
    }
</style>
</head>
<body>
    <!-- div.box1*3:创建三个div并指定class属性的名字是box1 -->
    <div class="box1"></div>
    <div class="box2"></div>
    <div class="box3"></div>
</body>
</html>

```

效果图：



8、伪类选择器

用于向选择器添加特殊的效果，以:分割开，当用户和网站交互的时候改变显示效果可以使用类选择器

示例代码：

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>伪类选择器</title>

```

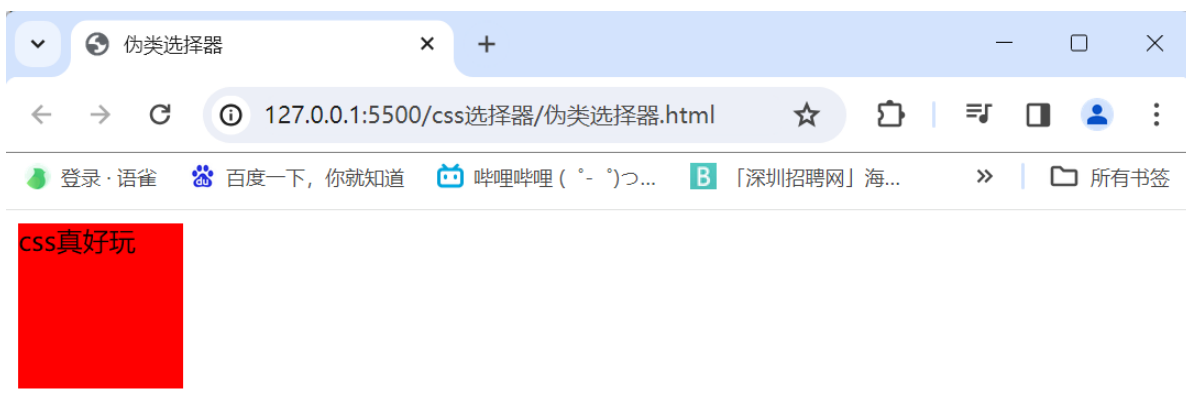


```

<style>
  div{
    width: 100px;
    height: 100px;
    background: red;
  }
  /* 伪类选择器就是给其他选择器添加特殊的效果，表现形式选择器后面加上冒号在跟上伪类的名字 */
  div:hover{
    width: 200px;
    height: 200px;
    background: green;
  }
</style>
</head>
<body>
  <div >css真好玩</div>
</body>
</html>

```

效果图：



五、CSS的属性

了解一下就行，不需要全部记住

1、布局常用样式属性

width: 设置元素（标签）的宽度，如：width: 100px

height: 设置元素或标签的高度。如：height: 100px

background: 设置元素背景色或背景图片。如：设置元素的背景色，background:gold;

设置元素背景图片，background:url(img/xxx);

border:设置元素四周的边框。如：border:1px solid black;设置元素四周边框是1像素宽的黑色实线

以上也可以拆分四个边的写法，分别设置四个边：

border-top 设置顶边边框，如：border-top:10px solid red;

border-left 设置左边边框，如：border-left:10px solid blue;

border-right 设置右边边框，如：border-right:10px solid green;

border-bottom 设置底边边框，如：border-bottom:10px solid pink;

padding 设置元素包含的内容和元素边框的距离，也叫内边距，如padding:20px;padding是同时设置4个边的，也可以像border一样拆分成分别设置四个边：padding-top,padding-left,padding-right,padding-bottom.

2、文本常用样式属性

color 设置文字的颜色，如：color:red;

font-size 设置文字的大小，如：font-size:12px;

font-family 设置文字的字体，如:font-family:'微软雅黑'; 为了避免中文字不兼容，一般写成：font-family:'Microsoft Yahei';

font-weight 设置文字是否加粗，如：设置加粗 font-weight:bold; 设置不加粗 font-weight:normal;

line-height 设置文字的行高，如：line-height:24px;表示文字高度加上文字上下的间距是24px，也就是每行占有的高度是24px

text-decoration 设置文字的下划线，如：text-decoration:none;将文字下划线去掉

text-align 设置文字水平对齐方式，如：text-align:center;设置文字水平居中

text-indent 设置文字首行缩进，如：text-indent:24px;设置文字首行缩进24px

3、布局常用样式属性代码

```
<head>
  <!-- 网页采用utf-8的编码格式 -->
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <!-- 网页标题 -->
  <title>Document</title>
  <!-- 以内部样式为例 -->
  <style>
    #name {
      /* font-size 设置字体大小 */
      font-size: 20px;
      /* font-size 设置字体粗细 */
      font-weight: 400;
      /* 设置字体 */
      font-family: '仿宋';
    }
  </style>
</head>
<!-- body包裹的称为内容区域 -->

<body>
  <div id="name">我是div标签</div>
</body>
```

```
<!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>
  p{
    /* 设置文字颜色 */
    color: red;
    /* 设置文字大小 */
    font-size: 30px;
    /* 设置文字加粗 */
    font-weight: bold;
    /* 设置文字的字体 */
    font-family: "Microsoft Yahei";
    background: blue;
    /* text-decoration: underline; */
    /* text-decoration: line-through; */
    /* 设置下划线 */
    text-decoration: overline;
    /* 设置行高，可以让文字垂直方向居中 */
    line-height: 100px;
    /* 设置水平方向的对象方式 */
    text-align: left;
    /* 文本缩进 */
    text-indent: 30px;

  }

  span {
    color: green;
  }

  a {
    /* 取消下划线 */
    text-decoration: none;
  }
</style>

</head>
<body>
  <!-- span标签可以给文本中的一小段内容设置样式 -->
  <p>听说下雨天音乐和<span>辣条</span>更配哟~</p>
  <a href="http://www.baidu.com">百度</a>
</body>
</html>

```

六、CSS元素溢出

1、什么是CSS的元素溢出？

当子元素（标签）的尺寸超过父元素（标签）的尺寸时，此时需要设置父元素显示溢出的子元素的方式，设置的方式是通过overflow属性来完成的。

overflow的设置项：

- 1.visible默认值，显示子标签溢出部分
- 2.hidden隐藏子标签溢出部分

3.auto如果子标签溢出，则可以滚动查看其余的内容

示例代码：

```
<!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>
    .box1{
      width: 100px;
      height: 100px;
      background: red;
      /* 设置元素溢出需要在父标签选择器上进行设置 */
      /* 超出部分隐藏 */
      /* overflow: hidden; */
      /* 超出部分滚动显示 */
      overflow: auto;
    }
    .box2{
      width: 200px;
      height: 50px;
      background: green;
    }
  </style>
</head>
<body>
  <!-- div.box1>div.box2:生成下面的代码 -->
  <div class="box1">
    <div class="box2">hahahha</div>
  </div>
</body>
</html>
```

效果图：

