# CSS基础

# 1. 表单

# 作用：用来收集用户信息的;

# <form ></form>

# name 属性：规定表单的名称，form 元素的 name 属性提供了一种在脚本中引用表单的方法

# action属性：规定当提交表单时，向何处发送表单数据。

# method属性：规定如何发送表单数据（表单数据发送到 action 属性所规定的页面）浏览器使用method属性设置的方法将表单中的数据传送给服务器进行处理。共有两种方法：POST 方法和 GET 方法。

# 1）文本框

# <input type="text" />

# type属性： 规定 input 元素的类型；。

# value属性：为 input 元素设定值；

# 对于不同的输入类型，value 属性的用法也不同

# type="text", "password", "hidden" - 定义输入字段的初始值

# name属性：规定 input 元素的名称。

# name 属性用于对提交到服务器后的表单数据进行标识，或者在客户端通过 JavaScript 引用表单数据，只有设置了name属性的表单元素才能在提交表单时传递它们的值。

# placeholder属性：提供可描述输入字段预期值的提示信息，该提示会在输入字段为空时显示，并会在字段获得焦点时消失。

# maxlength属性：规定输入字段中的字符的最大长度

# 2 ) 密码框

# <input type="password" placeholder="密码" name=" " value= " " />

# 3）提交按钮

# <input type="submit" value="按钮内容" />

# 4）按钮

# <input type="button" name="" value="按钮内容" />

# 5）重置按钮

# <input type="reset" value="按钮内容" name=“”/>

# value属性在type= "button ", "reset”, "submit " 的标签上，表示定义按钮上的显示的文本

# button和submit的区别：

# submit是提交按钮起到提交信息的作用，button只是一个按钮；

# 二、CSS简介

# CSS(cascading style sheets)WEB标准中的表现标准语言,表现标准语言在网页中主要对网页信息的显示进行控制，简单说就是如何修饰网页的显示样式。---级联样式表、层叠样式表

# 目前推荐遵循的是W3C发布的CSS3.0

# css3在包含了所有css2.0所支持的基础上改变了改进，css遵循的是模块式开发，发布的时间不是一个点而是一个时间段。

# 1998年5月21日由w3C正式推出的css2.0

# (在css1的基础上增加了高级功能，浮动，定位，高级选择器等（子选择器，相邻选择器，通用选择器）

# 1996年推出CSS1

# 包含非常基本的属性，比如字体，颜色、空白边

# 历史

**在使用CSS之前，Web开发人员依赖于一些特定的HTML标签和属性来增强网页的视觉体验，而依赖表格来进行布局设计。**

和样式有关的一些HTML标签：

* <basefont> 定义整个网页文档的字体
* <font> 定义一个具体的字体类型，颜色和它所包含的文本大小
* <center> 水平居中它所包含的元素
* <big> 使文本变大
* <strike> 为文本添加删除线

和样式有关的一些HTML属性：

* bgcolor 定义元素的背景色
* text 定义文本颜色
* margin 定义元素的边距

在页面整体布局的设计上曾经广泛使用表格 <table> 元素。因为表格提供了一个可视化的网格（grid）使得创建列、对齐和定位页面中的各个元素要相对直观一些；

**这些样式和布局方法有很多缺点：**

（1）内置的HTML元素和属性，使得表现和内容绑定在一起，如果要提供多样式的网页，必然要提供多份HTML文档内容

（2）HTML表格的代码相当繁琐，需要很多公式化标签，吃力不讨好

（3）HTML表格标记用法是语义错误的： 表格应该只是用来组织多维数据的，而不是用来布局的

（4）改变布局需要更改标记：如果我们想把左边的列移到右边，我们得修改HTML文档结构

（5）HTML表格容易出现语法错误：行和列需要遵循严格的排序和嵌套关系

（6）HTML表格标记用法可读性差：需要使用嵌套表格来支持列中有列的布局，结构将变得非常复杂

这就是为什么内置HTML样式和使用表格进行布局在实践中已经逐渐被抛弃，转而使用div+CSS来代替的原因。

**内容和表现分离的好处：**

(1)同一份文档可以拥有不同的呈现，提高了文档的可复用性和设计灵活性

(2) 内容和样式分离使得HTML文档精简清晰，可读性高

(3) 由于浏览器会缓存静态文件如CSS/JS，这将有助于提高网站性能和SEO评估

(4) 代码的可维护性好

**三、css语法**

选择符{属性:属性值; 属性:属性值;}

选择符：表示要定义样式的对象

1）每个CSS样式由两部分组成，即选择符和声明，声明又分为属性和属性值；

2）属性必须放在{}内；

3）属性与属性值用冒号连接

4）当一个选择符有多个属性时，用分号隔开；

5）当一个属性有多个属性值时，用空格分隔；

6）使用空格、换行不影响css样式的显示。

**简单样式**

width:value+px；设置标签的宽度

height:value+px;设置标签的高度

background:#colorValue; 设置标签的背景颜色

color:#colorValue；设置文本颜色

border:1px solid #colorValue; 设置标签的边框

边框类型：solid实线边框 double双线边框 dashed虚线 dotted点状线

**样式表分类**

样式:内部样式 外部样式 和 内联样式

**内部样式**

语法:

<style type="text/css">

css语句

</style>

注：使用style标记创建样式时，最好将该标记写在<head></head>里面;

**外部样式**

外部样式的建立及调用（@charset "utf-8";在外部样式表文件内使用。指定该样式表使用的字符编码。该规则后面的分号是必需的，如果省略了此分号，会生成错误信息。）

a:外部样式表的创建。 b:外部样式表的导入。

\*方法 一

<link rel="stylesheet" type="text/css" href="目标文件的路径及文件名全称" />

说明：使用link元素导入外部样式表时，需将该元素写在文档头部，即<head>与</head>之间。

\*方法 二

<style type="text/css">

@import url(目标文件的路径及文件名全称);

</style>

注：@和import之间没有空格 url和小括号之间也没有空格；必须结尾以分号结束；

**两种导入样式的区别**

link和import导入外部样式的区别：

差别1：

link属于XHTML标签，而@import完全是CSS提供的一种方式。 link标签除了可以加载CSS外，还可以做很多其它的事情，比如定义RSS，定义rel连接属性等，@import就只能加载CSS。

差别2：

加载顺序的差别：当一个页面被加载的时候（就是被浏览者浏览的时候），link引用的CSS会同时被加载，而@import引用的CSS会等到页面全部被下载完再被加载。所以有时候浏览@import加载CSS的页面时开始会没有样式。

差别3：

兼容性的差别：@import是CSS2.1提出的，所以老的浏览器不支持，@import只在IE5以上的才能识别，而link标签无此问题。

差别4：

使用dom控制样式时的差别：当使用javascript控制dom去改变样式的时候，只能使用link标签，因为@import不是dom可以控制的

**内联样式表**（行间样式，行内样式，嵌入式样式）

语法：<标签 style="属性:属性值 ; 属性:属性值;"></标签>

**样式表的优先级**

内联样式表的优先级别最高

内部样式表与外部样式表的优先级和书写的顺序有关，后书写的优先级别高

**四、选择符（选择器）**

**1） 元素选择符/类型选择符（element选择器 )**

语法：元素名称{属性：属性值；}

说明：元素选择符就是以文档语言对象类型作为选择符，即使用结构中元素名称作为选择符。例如body、div、p,img,em,strong,span等。

用法：当统一或者改变文档某类元素的显示效果时，可以使用类型选择符；

**2) id选择器**

语法：#id名{属性：属性值;}

说明：

1）使用id选择符时，应该为每个元素定义一个id属性

2）id选择符的语法格式是：#id名{}

3） 起名时要取英文名，不能用关键字(所有的标记和属性都是关键字)

4）一个id名称只能对应文档中一个具体的元素对象，因为id只能定义页面中某一个唯一的元素对象。

5) 最大的用处：创建网页的外围结构。

**3）class选择器**

语法：.class名{属性：属性值;}

说明： 使用类选择符时，为每个元素定义一个类名称，语法格式是：

<div class=“top”></div>在css中使用：.top{}

用法：1）class选择符更适合定义一类样式；

div.top{} p.top{}

2 ） 一个标记可以设置多个class名

语法：<标记 class="名称1   名称2" ></标记>

3）class名可以重复

**4）群组选择器**

语法：选择符1，选择符2，选择符3{属性：属性值;}

说明：当有多个选择符应用相同的样式时，可以将选择符用“，”分隔的方式，合并为一组。

**5） 包含选择器/后代选择符**

语法：选择符1 选择符2{属性：属性值;}

说明：

选择符1和选择符2用空格隔开，含义就是选择符1中包含的所有选择符2;

**6）子选择器**

语法：选择符1 > 选择符2{属性：属性值;}

说明：

选择符1和选择符2用>链接，选择符2必须是选择符1的子类

**7） 伪类选择器(伪类选择符)**

语法 ：

:link{属性：属性值;}超链接的初始状态;

:visited{属性：属性值;}超链接被访问后的状态;

:hover{属性：属性值;}鼠标悬停，即鼠标划过超链接时的状态;

:active{属性：属性值;}超链接被激活时的状态，即鼠标按下时超链接的状态;

说明：  
1）当这4个超链接伪类选择符联合使用时，应注意他们的顺序，正常顺序为：a:link,a:visited,a:hover,a:active,错误的顺序有时会使超链接的样式失效；

2）使用与超链接相关的伪类选择符时，应为a元素定义href属性;

**8）通配符**

语法：\*{属性：属性值；}

说明：通配选择符的写法是“\*”，其含义就是所有元素。

用法：常用来重置样式

**全局样式**

\*{margin:0;padding:0;}

margin:0 auto; 让子元素相对父元素水平居中显示