# day27-前端【html&css】

# 学习目标

01\_能够使用表单form标签创建表单容器

02\_能够使用CSS的基本选择器选择元素

03\_能够使用CSS的扩展选择器选择元素

04\_能够使用常见的CSS属性

05\_能够说出盒子模型的属性

06\_能够制作黑马旅游网的注册页面

# 项目清单

|  |
| --- |
|  |

# 回顾和讲解随堂测试

# 问题引入

HTML技术对一些精确/精细化的样式控制不够好，有的还不能控制样式，如：让字体比<h1>标签更大，比<h6>标签更小，为div标签加边框等等，甚至<font>标签都过式了，我们少用或尽量不用font标签，也要能达到同样的效果。

那么遇到上述问题，我们应用怎么办呢？用什么技术解决这些问题呢？

答案是：CSS技术可以很出色的解决上述我们遇到的问题。

|  |
| --- |
|  |

00\_font.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>问题引入</**title**> </**head**> <**body**>  <**font size="7" color="red"**>这是一个好人</**font**>   <**hr**/>   <**span style="font-size**:50**px**;**color**:**red**;**"**>这是一个好人</**span**> </**body**> </**html**> |

# 第01章 表单

## 1.1表单标签

### 1.1.1表单标签作用

用于将客户端的数据提交给服务器，需要提交大量数据的场景，通常会使用到表单。

表单标签名字：form，注意与from区别。表单在网页上不可见，只是一个容器。

一个网页即可以有一个或多个table，也可以有一个或多个form。

表格是用来定义html布局的，表单是收集客户端数据提交到服务器的。

|  |
| --- |
|  |

### 1.1.2 form常用的属性

|  |  |
| --- | --- |
| **常用属性** | **作用** |
| **action** | 指定服务器提交的地址 |
| **method** | 提交的方法有两种：GET或POST，大小写不敏感 |

|  |
| --- |
| <form action="server" method="post">  用户名：  <input type="text" name="username"><br/>  密码：  <input type="password" name="password"><br/>  <input type="submit" value=”提交”/> </form> |

### 1.1.3 GET与POST在格式上的区别

|  |  |
| --- | --- |
| **提交方法** | **特点** |
| **GET** | 默认是GET，参数会显示在地址栏上，地址与参数之间使用?分隔。参数之间使用&符号分隔 |
| **POST** | 数据提交更加安全，参数不会显示在地址栏上，通常使用POST方式提交数据 |

注意：GET和POST，在书写时，大小写不敏感。

## 1.2表单中的控件

### 1.2.1表单组件的认识

|  |
| --- |
|  |

### 1.2.2常用的表单控件

|  |  |  |  |
| --- | --- | --- | --- |
| **表单项** | **代码** | **属性** | **备注** |
| **文本框** | <input  type="text"  name="名字"> | type：属性默认为text  name：控件的名字，  名字是给服务器使用的  value：文本框中的值  readonly：只读，不能修改  数据能提交给服务器  disabled：不可用，数据不能提交给服务器。 |  |
| **密码框** | <input type="password"  name=”名字”> |  | 输入的字符不可见，  用\*号替代 |
| **单选框** | <input  type="radio"  name="gender"  value="男"  checked="checked"> | checked：默认选中某项  value：必须要指定value才有值 | 名字相同的单选框是同一组，一组中只能选中一个。 |
| **复选框** | <input  type="checkbox" name="hobby"  value="游泳"> | checked：默认是选中  value：选中这项的值 |  |
| **下拉列表** | <select> 容器，  指定name属性，  默认只能选中一项 | multiple：设置为多选  size：显示多少项 |  |
| <option> 其中一项 | value: 建议加上value选中这一项的值  selected: 默认选中的项 |  |
| **隐藏表单域** | <input  type="hidden"  name="id"  value="9324879023842"> | name：表单项的名字  value：表单项的值 | 在表单上不可见，但可以将数据提交给服务器，一般用于一些不需要用户看到的数据。 |
| **文件域** | <input  type="file"  name="photo" accept="image/\*"> | accept: 指定文件的类型，  如：image/gif  如果省略accept，表示可以接收  任意类型的数据 |  |
| **多行文本域** | <textarea  name="intro"  cols="50" rows="5">  默认值  </textarea> | rows：行数  cols：列数 | 注：没有value值，主体部分就是它的值 |
| **提交按钮** | <input  type="submit"  value="注册"/> | value：按钮上的文字 | 将整个表单提交给服务器  从<form>到</form>之间的  所有表单项内容都提交给  服务器 |
| **重置按钮** | <input  type="reset"  value="重置"/> |  | 还原到开始的状态 |
| **普通按钮** | <input  type="button"  value="按钮"/>  <button>按钮文字</button> |  | 没有功能，  用于后期JavaScript程序开发 |
| **图片按钮** | <input  type="image" src="img/regbtn.jpg"> |  | 有submit的功能  按钮上有一个图片 |

## 1.3案例：注册页面

### 1.3.1案例需求

结合表格布局，制作注册页面。

### 1.3.2案例效果

|  |
| --- |
|  |

### 1.3.3案例分析

1\_整个表单由8行2列组成，第1列显示文本，第2列显示表单控件。

2\_用户名、密码、性别、爱好、照片使用input标签，设置不同的type属性。

3\_学历使用select，个人简介使用textarea。

4\_最后1行跨2列，注册、清空、按钮的type分别是submit、reset、button。

### 1.3.4案例代码

01\_register.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>用户注册</**title**> </**head**> <**body**>  *<!--表单容器  action="表示将表单提交到服务器的什么资源进行处理，可以是今天我们讲的html，也可以是将来我们讲的Servlet/JSP"  method="表示表单提交的方式，常用的GET/POST"，不写的method默认是get，大小不区分   将用户点击submit按钮时，会将整个表单，即就是<form></form>中的所有表单项内容提交action所指明的地址  以post方式。  -->* <**form action="00\_font.html" method="post"**>  *<!--美化表单项的内容-->* <**table align="center" border="0" cellspacing="5" cellpadding="5"**>  <**tr**>  <**td**>用户名：</**td**>  <**td**><**input type="text" name="username" value="赵君" maxlength="20"**/></**td**>  </**tr**>  <**tr**>  <**td**>密码：</**td**>  <**td**><**input type="password" name="password" value="123456"**/></**td**>  </**tr**>  <**tr**>  <**td**>性别：</**td**>  <**td**>  <**input type="radio" name="sex" checked value="男"**/>男  <**input type="radio" name="sex" value="女"**/>女  </**td**>  </**tr**>  <**tr**>  <**td**>爱好：</**td**>  <**td**>  <**input type="checkbox" name="hobby" value="唱歌" checked**/>唱歌  <**input type="checkbox" name="hobby" value="跳舞"**/>跳舞  <**input type="checkbox" name="hobby" value="敲代码" checked**/>敲代码  <**input type="checkbox" name="hobby" value="找Bug"**/>找Bug  </**td**>  </**tr**>  <**tr**>  <**td**>学历：</**td**>  <**td**>  <**select name="edu"**>  <**option value="小学"**>小学</**option**>  <**option value="中学"**>中学</**option**>  <**option value="大学" selected**>大学</**option**>  <**option value="社会"** >社会</**option**>  </**select**>  </**td**>  </**tr**>  <**tr**>  <**td**>相片：</**td**>  <**td**>  <**input type="file" name="photo"**/>  </**td**>  </**tr**>  <**tr**>  <**td**>留言：</**td**>  <**td**>  <**textarea cols="30" rows="5" name="msg"**>好好学习，天天向上</**textarea**>  </**td**>  </**tr**>  <**tr**>  <**td colspan="2" align="center"**>  <**input type="submit" value="注册"**/>  **&nbsp;&nbsp;&nbsp;** <**input type="reset" value="重填"**/>  **&nbsp;&nbsp;&nbsp;** <**input type="button" value="普通按钮，将来和JS配合使用"**/>  </**td**>  </**tr**>  </**table**>  </**form**>  </**body**> </**html**> |

## 1.4 HTML5中新增的type类型

不同的浏览器支持上有差异，有些浏览器依然不支持。

|  |  |
| --- | --- |
| **值** | **描述** |
| **color** | **定义拾色器** |
| **date** | **定义日期字段**  **（带有 calendar 控件）** |
| **datetime-local** | **定义日期字段**  **（带有 calendar 和 time 控件）** |
| **month** | **定义日期字段的月**  **（带有 calendar 控件）** |
| **week** | **定义日期字段的周**  **（带有 calendar 控件）** |
| **time** | **定义日期字段的时、分、秒**  **（带有 time 控件）** |
| **email** | **定义用于 e-mail 地址的文本字段，如果输入不正确的邮箱地址有验证的功能** |
| **number** | **定义带有 spinner 控件的数字字段** |
| **range** | **定义带有 slider 控件的数字字段** |
| **search** | **定义用于搜索的文本字段，当输入内容的时候，搜索框后边会自动出现一个小×，点击这个小×，可以清除输入的内容。** |
| **tel** | **定义用于电话号码的文本字段，在手机上操作会出现输入数字的键盘。** |
| **url** | **定义用于 URL 的文本字段，在手机上操作会出现输入网址的键盘。** |

效果：

|  |
| --- |
|  |

代码：

02\_type.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>HTML5中新增的type类型</**title**> </**head**> <**body**>  <**form action="01\_register.html" method="get"**>  1:拾色器：  <**input type="color"**><**br**/>  2:日期：  <**input type="date"**><**br**/>  3:月份：  <**input type="month"**/><**br**/>  4:邮箱：  <**input type="email"**/><**br**/>  5:数字：  <**input type="number"**/><**br**/>  6:范围：  <**input type="range"**/><**br**/>  7:搜索：  <**input type="search"**/><**br**/>   <**input type="submit"**>  </**form**> </**body**> </**html**> |

# 第02章 CSS选择器

## 2.1 CSS概述

### 2.1.1什么是CSS

CSS称之为层叠样式表（Cascading Style Sheet）。在网页制作时可以有效地对页面的布局、字体、颜色、背景和其它效果实现更加精确的控制。让HTML制作出来的网页更加漂亮，页面更容易维护。

### 2.1.2为什么要CSS

以前我们要完成如下图示：

|  |
| --- |
| fig: |

需要对table中的td标签，设置align=”center”和style=”color:blue”属性，如果td标签过多，非常麻烦，而使用css则方便得多。

以前HTML方式：

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>为什么要CSS</**title**> </**head**> <**body**> <**table border="2" align="center" width="50%"**>  <**tr** >  <**td align="center" style="color**:**blue"**>100</**td**>  <**td align="center" style="color**:**blue"**>200</**td**>  <**td align="center" style="color**:**blue"**>300</**td**>  </**tr**>  <**tr**>  <**td align="center" style="color**:**blue"**>400</**td**>  <**td align="center" style="color**:**blue"**>500</**td**>  <**td align="center" style="color**:**blue"**>600</**td**>  </**tr**>  <**tr**>  <**td align="center" style="color**:**blue"**>700</**td**>  <**td align="center" style="color**:**blue"**>800</**td**>  <**td align="center" style="color**:**blue"**>900</**td**>  </**tr**> </**table**> </**body**> </**html**> |

现在HTML + CSS方式：

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>为什么要CSS</**title**>  <**style type="text/css"**>  /\*只要是本页面中所有td标签，统一使用这里定义的样式\*/  **td**{  **text-align**:**center**;  **color**:**blue**;  }  </**style**> </**head**> <**body**>  <**table border="2" align="center" width="50%"**>  <**tr**>  <**td**>100</**td**>  <**td**>200</**td**>  <**td**>300</**td**>  </**tr**>  <**tr**>  <**td**>400</**td**>  <**td**>500</**td**>  <**td**>600</**td**>  </**tr**>  <**tr**>  <**td**>700</**td**>  <**td**>800</**td**>  <**td**>900</**td**>  </**tr**>  </**table**> </**body**> </**html**> |

### 2.1.3 CSS美化的好处

|  |  |
| --- | --- |
| **CSS的好处** | **特点** |
| **功能上** | **比HTML属性更加强大，可以实现HTML属性中不能实现的功能** |
| **耦合度** | **代码降低了耦合度，分工更加明确，张三编写HTML，李四编写CSS，再将二者合二为一。**  **HTML：用于制作网页的结构。相当于：建房子**  **CSS：用于网页的美化。相当于：装修**  **所以项目中，HTML要配置CSS使用。** |

多个HTML可以使用同一个CSS，从而让HTML和CSS分离开发。

|  |
| --- |
|  |

### 2.1.4 CSS的编写规范

|  |
| --- |
| <**head**>  <**meta charset="UTF-8"**>  <**title**>为什么要CSS</**title**>  <**style type="text/css"**>  /\*文本在td中对齐，且文本蓝色\*/  **td** {  **text-align** : **center** ;  **color** : **blue** ;  }  </**style**> </**head**> |

|  |  |
| --- | --- |
| **规范** | **说明** |
| **大括号** | **所有的样式都写在大括号内部** |
| **样式名** | **字母全部小写，如果样式名有多个单词**  **使用-分隔** |
| **样式值** | **所有样式名和样式值是固定的，**  **名字与值之间使用冒号分隔** |
| **样式结尾** | **每个样式使用分号结尾** |
| **注释** | **/\* 注释 \*/**  **类似于Java中的多行注释** |

## 2.2 CSS的位置

### 2.2.1行内样式

只对某一行起作用，使用style属性。

|  |
| --- |
| <**tr** >  <**td style="text-align**:**left**;**color**:**red"**>100</**td**>  <**td style="text-align**:**left**;**color**:**red"**>200</**td**>  <**td style="text-align**:**left**;**color**:**red"**>300</**td**> </**tr**> |

### 2.2.2内部样式

出现在HTML内部的样式，使用<style>标签。

|  |
| --- |
| <**style type="text/css"**>  **td**{  **text-align** : **center**;  **color** : **green**;  } </**style**>  <**tr**>  <**td**>400</**td**>  <**td**>500</**td**>  <**td**>600</**td**> </**tr**> |

### 2.2.3外部样式

把CSS文件放在HTML的外部，使用的时候需要引入进来才可以使用。

引入方式一：推荐

|  |
| --- |
| **<link type="text/css" rel="stylesheet" href="css/my.css"/>** |
| **type：指定文本的类型为CSS，固定写法** |
| **rel：指定当前的HTML与CSS文件之间的关系是样式表的关系 relation，固定写法** |
| **href：指定样式文件地址** |

|  |
| --- |
| <**link href="css/my.css" rel="stylesheet" type="text/css"**>  <**tr**>  <**td**>700</**td**>  <**td**>800</**td**>  <**td**>900</**td**> </**tr**> |

引入方式二：

import指令导入。其中： url函数，样式文件地址

|  |
| --- |
| <**style type="text/css"**>  **@import url**(**"css/my.css"**); </**style**> |

### 2.2.4优先级

就近原则，哪个最近，就会覆盖前面的同名的样式。

行内样式 > （内部或外部样式）。

如果内部或外部样式同时出现，且又针对同一个标签，后者覆盖前者。

### 2.2.5案例：CSS三种方式的优先级

#### 2.2.5.1案例需求

分别采用行内，内部，外部，三种方式将<td>标签中的文本居中且变色显示。

#### 2.2.5.2案例效果

|  |
| --- |
|  |

#### 2.2.5.3案例分析

1\_采用行内样式，将第一行td中的文本居中且蓝色。

2\_采用内部样式，将所有行td中的文本居中且红色。

3\_采用外部样式，将所有行td中的文本居中且绿色。

#### 2.2.5.4案例代码

03\_css.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>CSS三种方式的优先级</**title**>  <**style type="text/css"**>  **td**{  **text-align** : **center**;  **color** : blue;  }  </**style**>  *<!--同名样式，后者覆盖前者-->* <**link href="css/my.css" rel="stylesheet" type="text/css"**> </**head**> <**body**> <**table border="2" align="center" width="50%"**>  <**tr** >  <**td style="text-align**:**left**;**color**:**red"**>100</**td**>  <**td style="text-align**:**left**;**color**:**red"**>200</**td**>  <**td style="text-align**:**left**;**color**:**red"**>300</**td**>  </**tr**>  <**tr**>  <**td**>400</**td**>  <**td**>500</**td**>  <**td**>600</**td**>  </**tr**>  <**tr**>  <**td**>700</**td**>  <**td**>800</**td**>  <**td**>900</**td**>  </**tr**> </**table**> </**body**> </**html**> |

## 2.3基本选择器

### 2.3.1选择器的作用

使用设置好的样式之前，先要选中哪些元素/标签使用这些样式。

选中的方式通过选择器来操作。

### 2.3.2语法格式

|  |
| --- |
| 选择器 {  样式名 ：样式值 ;  样式名 ：样式值 ;  } |

### 2.3.3基本选择器

|  |  |  |  |
| --- | --- | --- | --- |
| **选择器分类** | **作用** | **语法** | **细节** |
| **标签选择器** | **通过标签名选择同名的所有的标签** | **标签名 { }** |  |
| **类选择器（常用）** | **通过class属性的值选择元素** | **. 类名 { }** | **前提：先给标签进行分类，**  **使用class属性分类**  **类名：不能以数字开头** |
| **ID选择器** | **通过属性ID选择元素** | **# ID { }** | **前提：先给标签指定ID属性**  **唯一：建议ID在同个网页中唯一，不要重复** |
| **通用选择器** | **选中整个网页中所有的元素** | **\* {**  **}** |  |

### 2.3.4基本选择器之间的优先级

通用选择器（\*） < 标签选择器（div） < 类选择器（./class） < ID选择器（#/id）

### 2.3.5案例：使用CSS语法和四个基本选择器

#### 2.3.5.1案例需求

使用CSS语法和四个基本选择器。

#### 2.3.5.2案例效果

|  |
| --- |
|  |

#### 2.3.5.3案例分析

1\_在<head>标签中，定义四种基本选择器。

2\_在<body>标签中，使用四种基本选择器。

3\_“你好”使用标签选择器。

4\_“他好”使用类选择器。

5\_“我好”使用ID选择器。

6\_“大家好”使用标签选择器。

7\_“大家也好”使用通用选择器。

#### 2.3.5.4案例代码

04\_selector.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>基本选择器</**title**>  <**style**>  */\*标签选器器\*/* **div**{  **color**:**red**;  }  */\*类选择器\*/* .**classSelector**{  **color**:**green**;  }  */\*ID选择器\*/* **#idSelector**{  **color**:**blue**;  }  */\*通用选择器，针对网页中所有标签\*/* **\***{  **color**:**purple**;  **font-size**: 20**px**;  }  .**a**{  **color**:**yellow**;  }  .**b**{  **color**:**orange**;  }  .**c**{  **color**:**pink**;  }  </**style**> </**head**> <**body**>  *<!--使用标签选择器-->* <**div**>你好</**div**>   *<!--使用类选择器，注意使用类时，不加.号-->* <**div id="idSelector" class="classSelector"** >他好</**div**>   *<!--使用ID选择器,ID确保在这个html中唯一-->* <**div class="classSelector"**>我好</**div**>   *<!--使用标签选择器-->* <**div**>大家好</**div**>   *<!--使用通用选择器-->* <**span**>大家也好</**span**>   <**hr**/>   <**h1 class="a"**>大家好</**h1**>  <**h1 class="b"**>大家好</**h1**>  <**h1 class="c"**>大家好</**h1**>  </**body**> </**html**> |

## 2.4扩展选择器

### 2.4.1什么是扩展选择器

由基本选择器组合而成，可以有更加灵活的选择方式。

### 2.4.2常用的扩展选择器

|  |  |  |  |
| --- | --- | --- | --- |
| **扩展选择器** | **格式** | **作用** | **语法符号** |
| **层级选择器** | **父选择器 子孙选择器{ }** | **选择某个元素下面的所有子孙元素** | **中间至少一个空格** |
| **属性选择器** | **标签名[属性名]**  **标签名[属性名="属性值"]** | **包含属性名即可**  **属性名=属性值才可以**  **通过属性名和属性值选中符合条件的元素** | **[ ] 中括号** |
| **伪类选择器** | **链接：**  **a:link 正常状态**  **a:visited 访问过的**  **a:hover 鼠标悬停**  **a:active 正在激活**  **:focus 文本/密码框得到焦点** | **元素在操作的过程中，不同的状态呈现不同的样式**  **正在操作的元素表示得到了焦点** | **: 冒号** |
| **并集选择器** | **选择器1 , 选择器2** | **同时使用2个选择器的和** | **, 逗号** |

### 2.4.3案例：使用CSS语法和四个扩展选择器

#### 2.4.3.1案例需求

使用CSS语法和四个扩展选择器。

#### 2.4.3.2案例效果

|  |
| --- |
|  |

#### 2.4.3.3案例分析

1\_在<head>标签中，定义四种扩展选择器。

2\_在<body>标签中，使用四种扩展选择器。

3\_“用户登录”使用层次选择器。

4\_文本框使用属性选择器。

5\_“注册”超链接使用伪类选择器，密码框使用伪类选择器。

6\_”姓名”，“密码”，“提交按钮”使用并集选择器。

#### 2.4.3.4案例代码

05\_selector.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>扩展选择器</**title**>  <**style type="text/css"**>  */\*层级选择器\*/* **form div**{  **font-size**: 18**px**;  **font-weight**: **bold**;  }   */\*属性选择器\*/* **input**[**type**=**"text"**]{  **background-color**:**gainsboro**;  }   */\*伪类选择器\*/* **a**:**link**{  **color**:**black**;  }  **a**:**hover**{  **color**:**red**;  }  **input**[**type**=**"password"**]:**focus**{  **background-color**:**green**;  }   */\*并集选择器\*/* **#un**,**#pw**,**form input**[**type**=**"submit"**]{  **font-weight**: **bold**;  **color**: **purple**;  }  </**style**> </**head**> <**body**>  <**form**>  <**div**>用户登录</**div**>  <**span id="un"**>姓名</**span**>：<**input type="text" name="username"**/><**br**/>  <**span id="pw"**>密码</**span**>：<**input type="password" name="password"**/><**br**/>  <**input type="submit" value="提交"**/>  **&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;** <**a href="http://www.baidu.com"**>注册</**a**>  </**form**> </**body**> </**html**> |

# 第03章 常见的CSS样式

## 3.1背景样式

|  |  |  |
| --- | --- | --- |
| **功能** | **属性名** | **属性取值** |
| **背景色** | background-color | 颜色常量，如：red  使用十六进制，如：#123，#112233  使用rgb(红,绿,蓝)，如：rgb(255,200,150) |
| **背景图片** | background-image | url( “图片文件”)，如：url(“img/girl.jpg”) |
| **平铺方式** | background-repeat | repeat 默认。背景图像将在垂直方向和水平方向重复。  repeat-x 背景图像将在水平方向重复。  repeat-y 背景图像将在垂直方向重复。  no-repeat 背景图像将仅显示一次。 |
| **背景位置** | background-position | left top，如：background-position:10px 20px |
| **背景大小** | background-size | width height，如：background-size: 100px 100px |

效果：

|  |
| --- |
|  |

代码：

06\_backStyle.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>背景样式</**title**>  <**style type="text/css"**>  **body**{  */\*background-color:#112233;\*/* **background-image**: **url**(**"img/girl.jpg"**);  **background-repeat**: **repeat**;  **background-position**:10**px** 20**px**;  **background-size**: 100**px** 100**px**;  }  </**style**> </**head**> <**body**> </**body**> </**html**> |

## 3.2文本样式

|  |  |  |
| --- | --- | --- |
| **功能** | **属性名** | **属性取值** |
| **颜色** | color | 颜色常量，如：red  使用十六进制，如：#123，#112233  使用rgb(红，绿，蓝)函数 |
| **设置行高** | line-height | 单位是像素，px |
| **文字修饰** | text-decoration | underline 下划线  overline 上划线  line-through 删除线 |
| **文本缩进** | text-indent | 用于缩进文本，可以使用em单位，表示缩进2个字符，  无论字符的大小。 |
| **文本对齐** | text-align | left 把文本排列到左边。  right 把文本排列到右边。  center 把文本排列到中间。  默认值：由浏览器决定。 |

效果：

|  |
| --- |
|  |

代码：

07\_textStyle.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>文本样式</**title**>  <**style type="text/css"**>  .**firstDiv**{  **color**:**red**;  **line-height**:20**px**;  **text-decoration**:**line-through**;  **text-indent**:2**em**;  }  .**secondDiv**{  **color**:**green**;  **line-height**:20**px**;  **text-decoration**:**line-through**;  **text-indent**:0**em**;  **text-align**:**left**;  }  </**style**> </**head**> <**body**>  <**div class="firstDiv"**>  好好学习，天天向上。  </**div**>  <**div class="secondDiv"**>  好好工作，天天进取。  </**div**> </**body**> </**html**> |

## 3.3字体样式

|  |  |  |
| --- | --- | --- |
| **功能** | **属性名** | **作用** |
| **字体名** | font-family | 设置字体，本机必须要有这种字体 |
| **设置大小** | font-size | 单位：像素 |
| **设置样式** | font-style | 字体设置为斜体  italic 浏览器会显示一个斜体的字体样式。  normal 默认值。浏览器显示一个标准的字体样式。 |
| **设置粗细** | font-weight | bolder加粗 |

效果：

|  |
| --- |
|  |

代码：

08\_fontStyle.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>字体样式</**title**>  <**style type="text/css"**>  .**firstDiv**{  **font-family**: **幼圆**;  **font-size**:22**px**;  **font-style**: **italic**;  **font-weight**: **bold**;  }  .**secondDiv**{  **font-family**: **黑体**;  **font-size**:44**px**;  **font-style**: **normal**;  **font-weight**: **lighter**;  }  </**style**> </**head**> <**body**> <**div class="firstDiv"**>  好好学习，天天向上。 </**div**> <**div class="secondDiv"**>  好好工作，天天进取。 </**div**> </**body**> </**html**> |

## 3.4盒子模型

### 3.4.1什么是盒子模型

任何一个网页元素包含由这些属性组成：

内容（content）、边框（border）、内边距（padding）、边界（margin）。

这些属性我们可以用日常生活中的常见事物——盒子作一个比喻来理解，所以叫它盒子模型。整个网页由各种小盒子组成。

1\_内容（content）就是盒子里装的东西。

2\_内边距（padding）就是怕盒子里装的东西（贵重的）损坏而添加的泡沫或者

其它抗震的辅料。

3\_边框（border）就是盒子本身了。

4\_外边界（margin）则说明盒子摆放的时候的不能全部堆在一起，要留一定空隙

保持通风，同时也为了方便取出。

### 3.4.2盒子模型的属性

|  |
| --- |
|  |

|  |  |
| --- | --- |
| **属性** | **作用** |
| **width** | 宽度 |
| **height** | 高度 |
| **margin** | 外边距 |
| **padding** | 内边距 |
| **border** | 边框：粗细 线型 颜色 |

## 3.5盒子模型边框样式

一条语句设置四边值写法：

|  |  |
| --- | --- |
| **外边距的写法** | **含义** |
| **margin:10px** | 四边相同 |
| **margin:10px 20px;** | 上下 左右 |
| **margin:10px 20px 30px;** | 上 左右 下 |
| **margin:10px 20px 30px 40px;** | 上 右 下 左 （按顺时针方向） |

每条语句设置一边值写法：

|  |  |
| --- | --- |
| **外边距的写法** | **含义** |
| **margin-top:10px;** | 上外边距 |
| **margin-left:10px;** | 左外边距 |
| **margin-bottom: 10px;** | 下外边距 |
| **margin-right:10px;** | 右外边距 |

|  |  |
| --- | --- |
| **内边距的写法** | **含义** |
| **padding-top:10px;** | 上内边距 |
| **padding-left:10px;** | 左内边距 |
| **padding-bottom: 10px;** | 下内边距 |
| **padding-right:10px;** | 右内边距 |

效果：

|  |
| --- |
|  |

代码：

09\_borderStyle.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>边框样式</**title**>  <**style type="text/css"**>  .**firstDiv**{  **border-style**: **dotted**;  **border-color**: **red**;  **margin**:50**px** 50**px** 50**px** 50**px**;  **padding**: 20**px** 20**px** 20**px** 20**px**;  }  .**secondDiv**{  **border-style**: **groove**;  **border-color**: **blue**;  }  </**style**> </**head**> <**body**> <**div class="firstDiv"**>  好好学习，天天向上。 </**div**> <**div class="secondDiv"**>  好好工作，天天进取。 </**div**> </**body**> </**html**> |

# 第04章 案例：黑马旅游注册页面

## 4.1案例需求

运用今天学过的CSS，并结合table布局和div布局的方式制作黑马旅游注册页面。

## 4.2案例效果

|  |
| --- |
|  |

## 4.3案例分析

|  |
| --- |
|  |

**一：外面大的div**

1. 设置所有元素样式的内，外边距为0px，设置box-sizing：border-box，

即：边框的宽和高就是盒子的宽和高，不会因为内边距或内容的变大而撑大。

1. 创建外面的白色div容器，类名rg\_form，宽986px，高534px，背景色为白色，

上下外边距24px，左右自动auto，边框为8px，实线，颜色为#eee。

**二：里面左边div**

1. 左边的的容器，类名为rg\_form\_left，宽度为256px，左浮动，上内边距20px，

左内边距20px。

1. “新用户注册”汉字，第1行使用p标签，

使用伪类样式，p:first-child，字体大小20px，颜色：#ffcd26。

1. “新用户注册”第2行英文使用p标签，内容：USER REGISTER，

使用伪类样式，p:last-child，字体大小20px，颜色：#a6a6a6。

**三：里面中间div**

1. 中间注册窗体的容器类名rg\_form\_center，宽458px，左浮动，上内边距10px，

字体大小14px。

1. 中间容器中的表格类名rg\_table，上外边距25px。
2. 表格中左边的文字所在单元格，类名td\_left，宽65px，文本左对齐。
3. 表格中右边的输入框单元格类名td\_right，宽293px，高50px，

其它表单项相同。

1. 表格中的文本框，日期框，密码框使用属性选择器，宽256px，高32px，

行高line-height：32px，内边距上下6px，左右12px，圆角border-radius:4px，

边框：1px 实线solid 颜色#a6a6a6，右浮动。

1. 性别输入框类名sex，左内边距40px。
2. 验证码输入框类名check，宽118px，左浮动，左外边距35px，右外边距0px。
3. 提交按钮类名submit，宽120px，高36px，行高36px，背景色#ffc900，

字体大小14px，边框类型为none。

**四：里面右边div**

1. 右边“已有账号”的容器类名 rg\_form\_right，宽156px，左浮动。
2. 右边的文字p标签，右浮动，字大小14px，内边距，上20px，右12px，下0px，左0px。

p内部有个a标签，颜色#fc8989。

## 4.4案例代码

10\_register.html

|  |
| --- |
| <!DOCTYPE **html**> <**html lang="en"**> <**head**>  <**meta charset="UTF-8"**>  <**title**>黑马旅游注册页面</**title**>  <**style**>  */\*1.设置所有元素样式的内，外边距为0px，设置box-sizing：border-box，  即：边框的宽和高就是盒子的宽和高，不会因为内边距或内容的变大而撑大。\*/* **\***{  **padding**: 0**px**;  **margin**: 0**px**;  **box-sizing**: **border-box**;  }  */\*2.创建外面的白色div容器，类名rg\_form，宽886px，高534px，背景色为白色，  上下外边距24px，左右自动auto，边框为8px，实线，颜色为#eee。\*/* .**rg\_form**{  **width**: 886**px**;  **height**: 534**px**;  **background-color**: **white**;  **margin**: 24**px auto**;  **border**: 8**px solid #eee**;  }   */\*3.左边的容器，类名为rg\_form\_left，宽度为256px，左浮动，上内边距20px，  左内边距20px。\*/* .**rg\_form\_left**{  **width**:256**px**;  **float**: **left**;  **padding-top**: 20**px**;  **padding-left**: 20**px**;  }   */\*4."新用户注册"汉字，第1行使用p标签，  使用伪类样式，p:first-child，字体大小20px，颜色：#ffcd26。\*/* **p**:**first-child**{  **font-size**: 20**px**;  **color**:**#ffcd26**;  }   */\*5."新用户注册"第2行英文使用p标签，内容：USER REGISTER，  使用伪类样式，p:last-child，字体20px，颜色：#a6a6a6。\*/* **p**:**last-child**{  **font-size**: 20**px**;  **color**:**#a6a6a6**;  }   */\*6.中间注册窗体的容器类名rg\_form\_center，宽458px，左浮动，上内边距10px，  字体大小14px。\*/* .**rg\_form\_center**{  **width**:458**px**;  **float**: **left**;  **padding-top**: 10**px**;  **font-size**: 14**px**;  }   */\*7.中间容器中的表格类名rg\_table，上外边距25px。\*/* .**rg\_table**{  **margin-top**: 25**px**;  }   */\*8.表格中左边的文字所在单元格，类名td\_left，宽65px，文本左对齐。\*/* .**td\_left**{  **width**: 65**px**;  **text-align**: **left**;  }   */\*9.表格中右边的输入框单元格类名td\_right，宽293px，高50px。其它表单项相同\*/* .**td\_right**{  **width**: 293**px**;  **height**: 50**px**;  }   */\*10.表格中的文本框，日期框，密码框使用属性选择器，宽256px，高32px，  行高line-height：32px，内边距上下6px，左右12px，圆角border-radius:4px，  边框：1px 实线solid 颜色#a6a6a6，右浮动。\*/* **input**[**type**=**"text"**],**input**[**type**=**"date"**],**input**[**type**=**"password"**]{  **width**: 256**px**;  **height**: 32**px**;  **line-height**: 32**px**;  **padding**: 6**px** 12**px**;  **border-radius**:4**px**;  **border**: 1**px solid #a6a6a6**;  **float**:**right**;  }   */\*11.性别输入框类名sex，左内边距40px。\*/* .**sex**{  **padding-left**: 40**px**;  }   */\*12.验证码输入框类名check，宽118px，左浮动，左外边距35px，右外边距0px。\*/* .**check**{  **width**:118**px**;  **float**: **left**;  **margin-left**: 35**px**;  **margin-right**: 0**px**;  }   */\*13.提交按钮类名submit，宽120px，高36px，行高36px，背景色#ffc900，  字体大小14px，边框类型为none。\*/* .**submit**{  **width**:120**px**;  **height**: 36**px**;  **line-height**: 36**px**;  **background-color**: **#ffc900**;  **font-size**: 14**px**;  **border-style**: **none**;  }   */\*14.右边"已有账号"的容器类名 rg\_form\_right，宽156px，左浮动。\*/* .**rg\_form\_right**{  **width**:156**px**;  **float**: **left**;  }   */\*15.右边的文字p标签，右浮动，字大小14px，内边距，上20px，右12px，下0px，左0px。  p内部有个a标签，颜色#fc8989。\*/* .**rg\_form\_right p**{  **font-size**: 14**px**;  **padding**: 20**px** 12**px** 0**px** 0**px**;  }   .**rg\_form\_right p a**{  **color** : **#fc8989**;  }   </**style**> </**head**> <**body**>  *<!--外面大的div-->* <**div class="rg\_form"**>   *<!--里面左边div-->* <**div class="rg\_form\_left"**>  <**p**>新用户注册</**p**>  <**p**>USER REGISTER</**p**>  </**div**>   *<!--里面中间div-->* <**div class="rg\_form\_center"**>  <**table class="rg\_table"**>  <**tr**>  <**td class="td\_left"**>帐号</**td**>  <**td class="td\_right"**><**input type="text" name="userid" placeholder="请输入帐号"**/></**td**>  </**tr**>  <**tr**>  <**td class="td\_left"**>密码</**td**>  <**td class="td\_right"**><**input type="password" name="password"**/></**td**>  </**tr**>  <**tr**>  <**td class="td\_left"**>邮箱</**td**>  <**td class="td\_right"**><**input type="text" name="email"**/></**td**>  </**tr**>  <**tr**>  <**td class="td\_left"**>姓名</**td**>  <**td class="td\_right"**><**input type="text" name="username"**/></**td**>  </**tr**>  <**tr**>  <**td class="td\_left"**>手机号</**td**>  <**td class="td\_right"**><**input type="text" name="phone"**/></**td**>  </**tr**>  <**tr**>  *<!--性别既要满足td\_left要求，又要满足sex要求-->* <**td class="td\_left"**>性别</**td**>  <**td class="td\_right sex"**>  <**input type="radio" name="sex" value="男"**/>男  <**input type="radio" name="sex" value="女" checked**/>女  </**td**>  </**tr**>  <**tr**>  <**td class="td\_left"**>生日</**td**>  <**td class="td\_right"**>  <**input type="date" name="birthday"**/>  </**td**>  </**tr**>  <**tr**>  <**td class="td\_left"**>验证码</**td**>  <**td class="td\_right"**>  <**input type="text" name="checkcode" class="check"**/>  </**td**>  <**td**>  <**img src="img/checkcode.jpg"**/>  </**td**>  </**tr**>  <**tr**>  <**td**></**td**>  <**td**>  <**input type="submit" value="注册" class="submit"**/>  </**td**>  </**tr**>  </**table**>  </**div**>   *<!--里面右边div-->* <**div class="rg\_form\_right"**>  <**p**>  已有帐号？  <**a href=""**>立即登录</**a**>  </**p**>  </**div**>   </**div**> </**body**> </**html**> |

# 随堂测试