



CSS3布局

项目9

项目导读

所谓布局，就是设置网页中各模块及模块中各元素的位置，它是网页制作的基础。本项目将介绍使用CSS3构建各种形式网页布局的方法，以及使网页适配移动端的方法。



学习目标

- ◆ 掌握使用CSS3构建单列和双列布局的方法。
- ◆ 掌握使用CSS3构建三列布局的方法。
- ◆ 掌握使用CSS3的弹性伸缩盒构建局部布局的方法。
- ◆ 掌握使网页适配移动端的方法。



内容概览

任务9.1

CSS3常见布局形式

任务9.2

适配移动端的设置



任务9.1

CSS3常见布局形式

- 单列和双列布局
- 三列布局
- 弹性伸缩盒

9.1.1 单列和双列布局

小型企业宣传网站一般使用简单的单列布局，构建这种布局形式只需按照标准文档流的顺序添加容器标签即可，必要时可通过将个别容器标签左右外边距属性值设置为 auto，使其居中显示。

双列布局一般是指带有左侧或右侧侧边栏的布局形式。构建这种布局形式可以通过设置浮动属性，使主体部分与侧边栏一个向左浮动一个向右浮动来实现。此外，也可以通过设置定位属性实现。



【例9-1】 使用定位属性构建双列布局，页面效果如下图9-1所示。



步骤1



创建HTML5文档，在<body>标签中输入以下代码，标记页面的容器标签。

```
<header>页眉</header>
  <main>
    <aside>侧边栏</aside>
    <div>主体部分</div>
  </main>
<footer>页脚</footer>
```

步骤2



在<head>标签中添加<style>标签，在其中输入代码（详情见教材），使用定位属性设置各容器的位置，构建双列布局。



提示：使用定位属性构建的页面结构更加稳定，但是在使用定位属性构建页面布局时，页眉页脚之间的中心区域必须设置高度，否则它的背景属性及其下方的页面元素都将无法显示。

9.1.2 三列布局

三列布局是指同时有两列侧边栏的布局形式。要使用定位属性实现三列布局，可在例9-1的基础上分别设置左、中、右3个模块绝对定位的左侧（或右侧）偏移量为0%、15%和85%（以两侧侧边栏宽度各占父元素的15%为例）。

使用浮动属性实现三列布局时，可直接设置各模块一同向左或向右浮动。但使用这种方法构建的布局不够稳定，在实际应用中，可通过设置外边距属性或将两个侧边栏模块分别向左和向右浮动来固定侧边栏的位置。

【例9-2】 使用浮动属性构建较为稳定的三列布局，页面效果如下图9-2所示。



步骤1



创建HTML5文档，在<body>标签中输入以下代码，构建三列布局的结构。

```
<header>页眉</header>
  <main>
    <aside class="left">左侧侧边栏</aside>
    <aside class="right">右侧侧边栏</aside>
    <div>主体部分</div>
  </main>
</body>
<footer>页脚</footer>
```



步骤2



在<head>标签中添加<style>标签，在其中输入代码（详情见教材），通过设置浮动与外边距属性构建三列布局。

9.1.3 弹性伸缩盒

除前面介绍的几种整体布局形式外，CSS3中还有一种常用于局部布局的盒子模型，称为弹性伸缩盒。使用弹性伸缩盒构建布局的方式叫作弹性伸缩盒布局（flex布局），它把父元素转换为具有弹性的伸缩盒，将未知大小的子元素以各种形式分布在父元素中，是一种比较简便、灵活的布局方式。

在CSS3中，使用display属性可将元素设置为伸缩盒，具体格式为：

```
display:flex|inline-flex;
```



拓展阅读

其中，flex表示最新版本的伸缩盒，inline-flex表示最新版本的行内伸缩盒。伸缩盒的默认宽度为100%，而行内伸缩盒的宽度取决于其子元素。



【例9-3】 使用弹性伸缩盒布局将块级元素显示在一行中，页面效果如下图9-3所示。



创建HTML5文档，参照以下代码段分别在<style>和<body>标签中输入代码（详情见教材），设置容器元素为弹性伸缩盒，使其块级子元素显示在一行中。

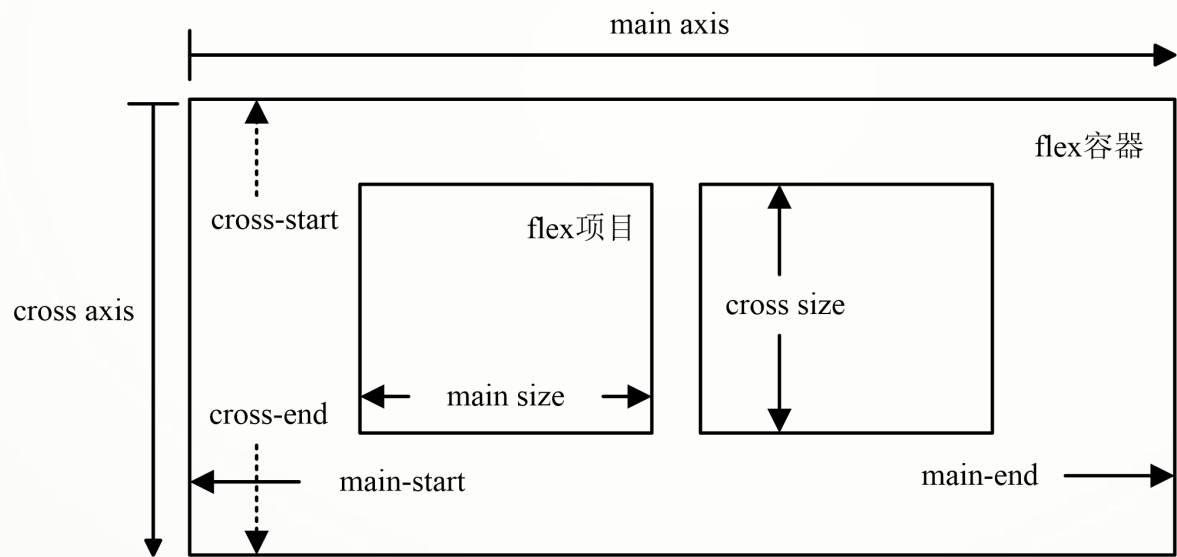


拓展阅读

弹性伸缩盒一直在不断发展，它的发展历程可大致分为3个阶段，分别为老版本阶段、过渡版本阶段与最新版本阶段。2009年的版本为老版本，`display`属性值为`box`与`inline-box`；2011年的版本为过渡版本，`display`属性值为`flexbox`与`inline-flexbox`；2012年的版本为最新版本。

1. flex布局结构

伸缩容器与伸缩项目也称为flex容器 (flex container) 与flex项目 (flex item)。伸缩项目沿伸缩容器的行排列，默认情况下，伸缩容器中只有一个伸缩行，并且与文本方向一致。flex布局的结构如图9-4所示。



1. flex布局结构

相关概念的解释如下。

- (1) 主轴 (main axis) : 伸缩容器的主轴指明伸缩行的方向, 默认为自左向右。
- (2) 主轴起点 (main-start) 与主轴终点 (main-end) : 主轴的开始位置为主轴起点, 结束位置为主轴终点。
- (3) 主轴长度 (main size) : 伸缩项目在主轴方向上的长度。
- (4) 侧轴 (cross axis) : 伸缩容器的侧轴垂直于主轴, 默认方向为自上向下。
- (5) 侧轴起点 (cross-start) 与侧轴终点 (cross-end) : 侧轴的开始位置为侧轴起点, 结束位置为侧轴终点。
- (6) 侧轴长度 (cross size) : 伸缩项目在侧轴方向上的长度。



2. 伸缩方向

在flex布局中，使用flex-direction属性设置伸缩容器中主轴的方向，也就是伸缩项目的排列方向，具体格式为：

```
flex-direction:row|row-reverse|column|column-reverse;
```

其中，row是默认值，表示自左向右排列；row-reverse表示自右向左排列；column表示自上向下排列；column-reverse表示自下向上排列。



拓展阅读

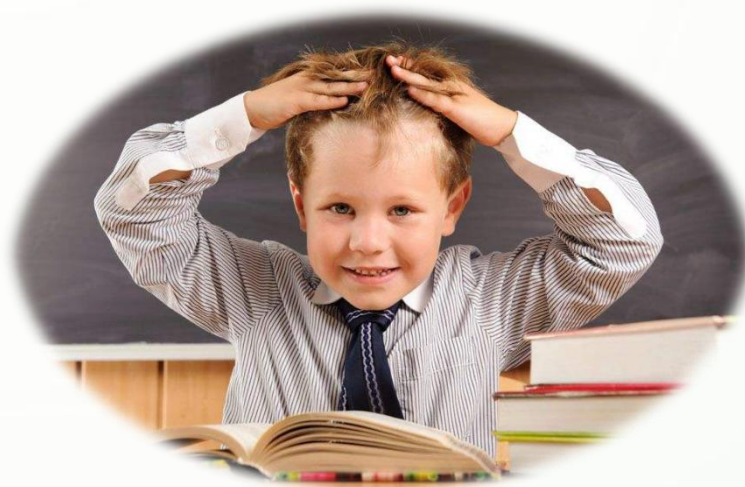
伸缩项目的排列方向受文字书写方向属性direction影响，其属性值为rtl（自右向左）时，flex-direction属性值row与row-reverse的排列顺序与原来相反；属性值column与column-reverse保持原排列顺序并自右侧向左侧排列。

3. 单行或多行显示

在flex布局中，使用flex-wrap属性设置伸缩容器是否多行显示，具体格式为：

```
flex-wrap:nowrap|wrap|wrap-reverse;
```

其中，nowrap是默认值，表示单行显示；wrap表示多行显示；wrap-reverse表示在wrap的基础上反向排列。



4. 对齐方式

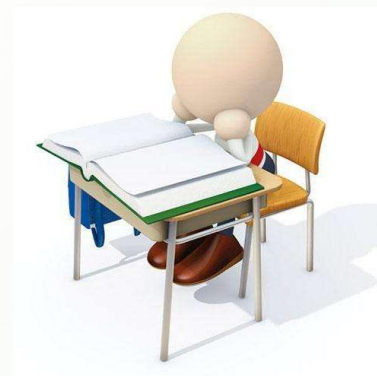
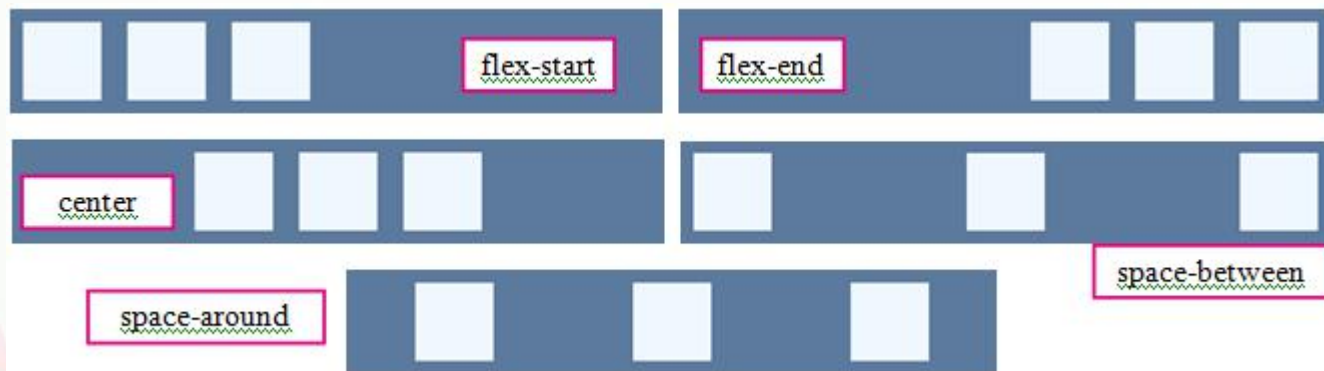
在flex布局中，设置伸缩容器的相关属性，能够使伸缩项目以不同的对齐方式显示。

(1) 主轴对齐

justify-content属性可以设置伸缩项目在主轴方向上的对齐方式，具体格式为：

```
justify-content: flex-start|flex-end|center|space-between|  
space-around;
```

其中，flex-start是默认值，表示向主轴起点方向对齐；flex-end表示向主轴终点方向对齐；center表示居中对齐；space-between表示两端对齐；space-around表示平均分布，它们的效果如图9-5所示。



(2) 侧轴对齐

`align-items`属性可以设置伸缩项目在侧轴方向上的对齐方式，具体格式为：

```
align-items: stretch|flex-start|flex-end|center|baseline;
```



其中，`stretch`是默认值，表示将伸缩项目拉伸至填充整个伸缩容器；`flex-start`表示向侧轴起点方向对齐；`flex-end`表示向侧轴终点方向对齐；`center`表示居中对齐；`baseline`表示根据元素中文本的基线对齐，它们的效果如下图9-6所示。



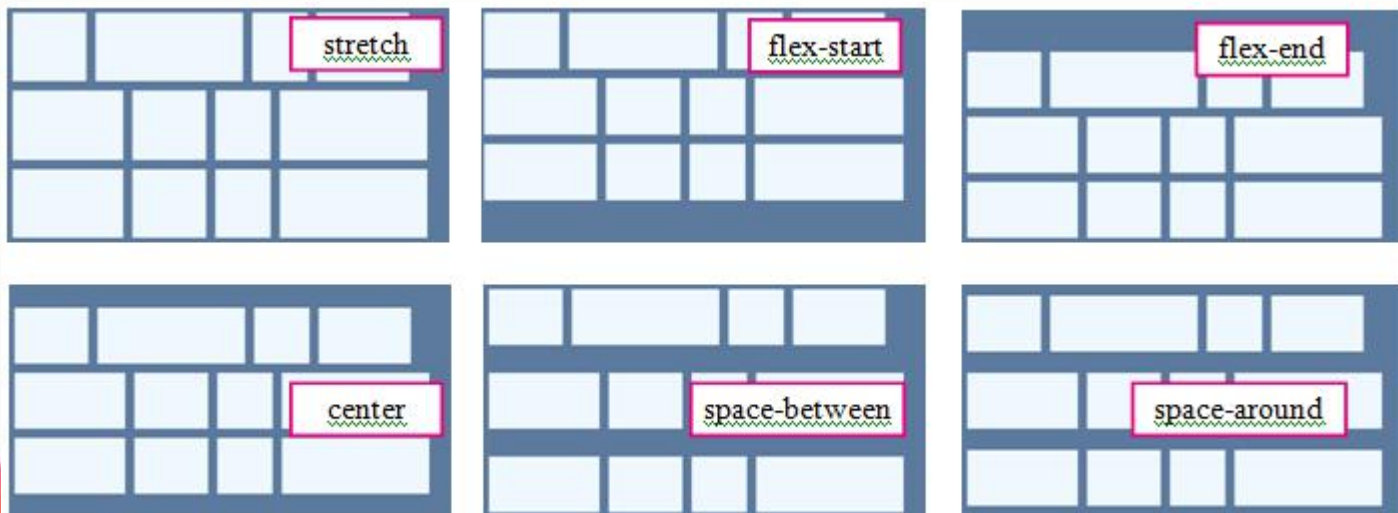
(3) 行对齐

`align-content`属性可以设置伸缩行在侧轴方向上的对齐方式，具体格式为：

```
align-content:stretch|flex-start|flex-end|center|
space-between|space-around;
```



其中，`stretch`是默认值，表示伸缩行拉伸至尽量占用伸缩容器的空间；`flex-start`表示向伸缩容器顶部对齐；`flex-end`表示向伸缩容器底部对齐；`center`表示向伸缩容器中心对齐；`space-between`表示向伸缩容器两端对齐；`space-around`表示在伸缩容器中平均分布，它们的效果如下图9-7所示。



提示： 设置了伸缩项目的宽度与高度后，`align-items`与`align-content`属性的默认值`stretch`不会拉伸伸缩项目，显示效果与`flex-start`类似。

5. 排列顺序

在flex布局中，设置伸缩项目的order属性可以调整其在伸缩容器中的排列顺序，属性值可以为正整数、负整数或0，默认值为0。order属性值越大的伸缩项目显示在伸缩容器中越靠后的位置。

【例9-4】 使用弹性伸缩盒制作水平导航条，页面效果如下图9-8所示。



步骤1



创建HTML5文档，在<body>标签中输入以下代码，构建导航条的结构。

```
<nav class="head">  
  <a href="#">秒杀</a>  
  <a href="#">优惠券</a>  
  ..... (省略部分代码，具体内容如图9-8所示)  
</nav>
```

步骤2



在<head>标签中添加<style>标签，在其中输入代码（详情见教材），设置容器元素为弹性伸缩盒，并设置伸缩项目的对齐方式，制作水平导航条。

任务实施

构建“书店简介”页面布局





任务9.2

适配移动端的设置

- 视口
- 响应式适配移动设备

9.2.1 视口

在PC端，视口是指浏览器的可视区域，视口宽度与浏览器窗口宽度一致。在CSS标准文档中，视口也称为初始包含块，它的宽度是所有CSS百分比宽度推算的根源，也为网页布局限制了最大宽度。

移动端浏览器的宽度受到屏幕设备的影响，一般为300 px左右，而大多数PC端网页的宽度都在800 px以上，如果仍然以浏览器窗口作为视口，将无法正常显示页面内容。





知识库

此处提到的移动端屏幕宽度一般为300 px左右，指的是逻辑像素，也就是设置样式时常用的单位。生活中常说的手机屏幕像素是指物理像素，物理像素是手机屏幕上的最小显示单位，像素大小是固定不变的。例如，iPhone X手机的物理像素为1125×2436，而因为其应用了视网膜三倍屏幕，以3×3个物理像素显示一个逻辑像素的图像，所以它的逻辑像素为375×812。

为使移动端视口与浏览器窗口不再关联，引入了布局视口（layout viewport）、视觉视口（visual viewport）与理想视口（ideal viewport）的概念，下面分别介绍。



1. 布局视口

在早期PC端与移动端网页共用一套代码时，为了能够在移动端正常浏览PC端网页，移动设备的浏览器都默认设置了一个虚拟的布局视口，IOS和Android基本都将这个视口的宽度设置为980 px。这样PC端的网页基本都能在手机上呈现，只是元素看上去小很多，可以手动缩放网页查看局部效果（本节中各示意图皆以文旌财经网校官网的页面为例），如下图所示。



2. 视觉视口

视觉视口是用户在屏幕上所能够看到的网页区域，当用户放大网页时，屏幕上能看到的网页区域变小，也就相当于视觉视口变小，如下图9-14所示。同理，当用户缩小网页时，屏幕上能看到的网页区域变大，也就相当于视觉视口变大。



3. 理想视口

当布局视口的宽度等于移动设备的屏幕宽度时，称其为理想视口。在理想视口下，用户不用缩放就能看到网页的最佳显示效果，如下图9-15所示。



设置理想视口后，网页响应式应用适配移动端的样式代码，得到更合适的页面效果

4. 设置布局视口



在网页文档的头部标签中添加<meta>标签，可以重新设置移动端的布局视口，具体格式为：

```
<meta name="viewport" content="属性=值,....." />
```

其中，name属性值为viewport，表示该标签用于设置布局视口；content属性用于设置布局视口的相关属性，如宽度与缩放比例等，常用属性及说明如下。



(1) width用于设置布局视口的宽度，默认值为980 px，值为device-width表示布局视口宽度等于当前移动端屏幕宽度（即理想视口）。

(2) user-scalable用于设置页面能否手动缩放，属性值yes是默认值，表示可以缩放；no表示不可缩放。



(3) initial-scale用于设置屏幕宽度与视口宽度之间的缩放比例，属性值通常设置为1，表示原大小。



(4) maximum-scale用于设置缩放的最大比例，属性值通常设置为1，表示原大小。

(5) minimum-scale用于设置缩放的最小比例，属性值通常设置为1，表示原大小。

(6) viewport-fit。viewport-fit用于适配iPhone X及其他型号的刘海屏iPhone手机（IOS系统），属性值cover表示将页面内容填充至顶部。



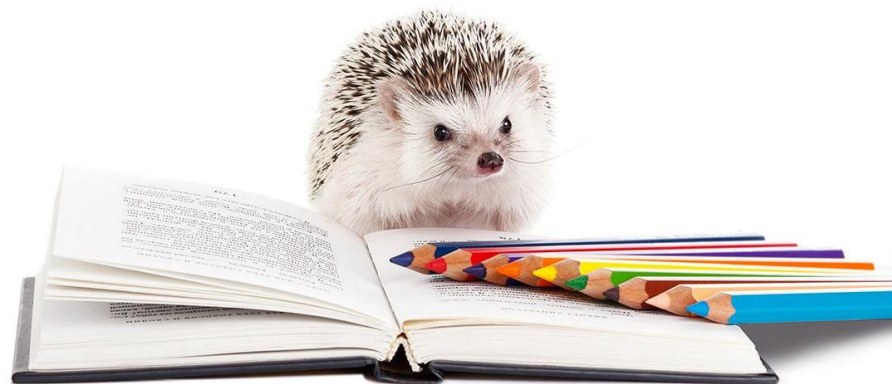
知识库

刘海屏一词最早来源于2017年9月苹果公司发布的iPhone X手机，iPhone X为实现全面屏将前置摄像头、听筒等元件放置在了屏幕上方的中心位置，因其形似刘海儿而得名。其后发布的安卓手机也纷纷效仿，刘海屏逐渐流行起来。但是在Android中，适配刘海屏手机需要使用JavaScript脚本，感兴趣的读者可查阅相关资料。

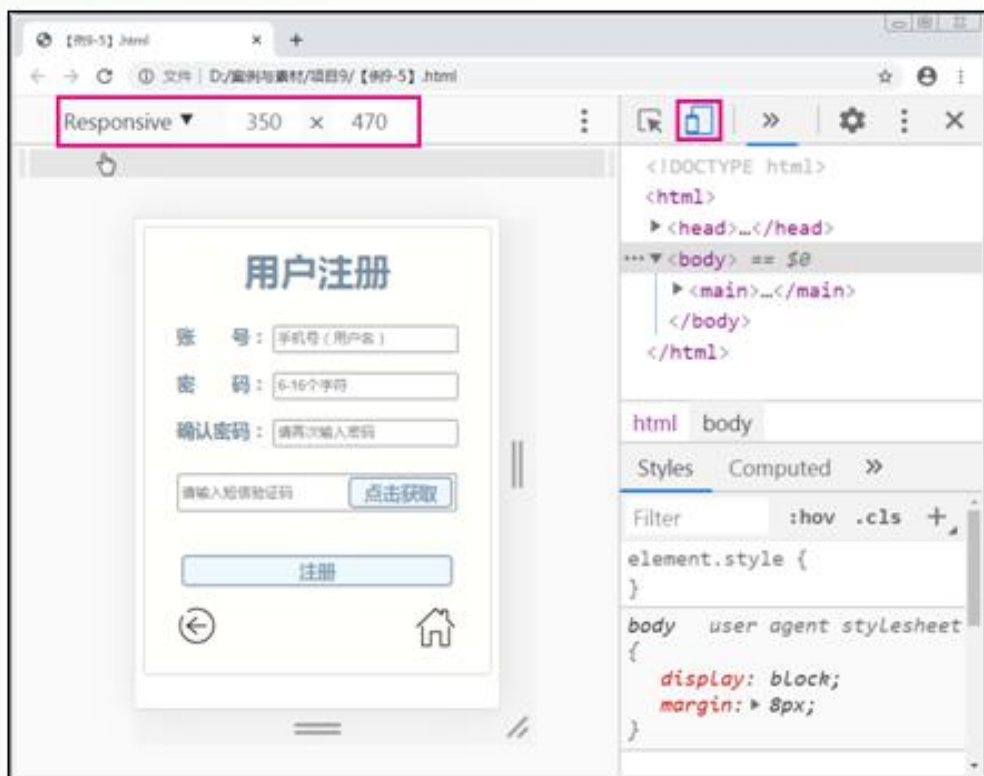
一般设置移动端网页视口的代码为：

```
<meta name="viewport" content="width=device-width,  
user-scalable=no,initial-scale=1,maximum-scale=1,  
minimum-scale=1,viewport-fit=cover" />
```

上述代码的含义为设置视口宽度为屏幕宽度、无法缩放（页面为原大小）且适配IOS系统的刘海屏手机。



【例9-5】 设置用户注册表单的视口，使其在移动设备中正常浏览，页面效果如下图9-16所示。

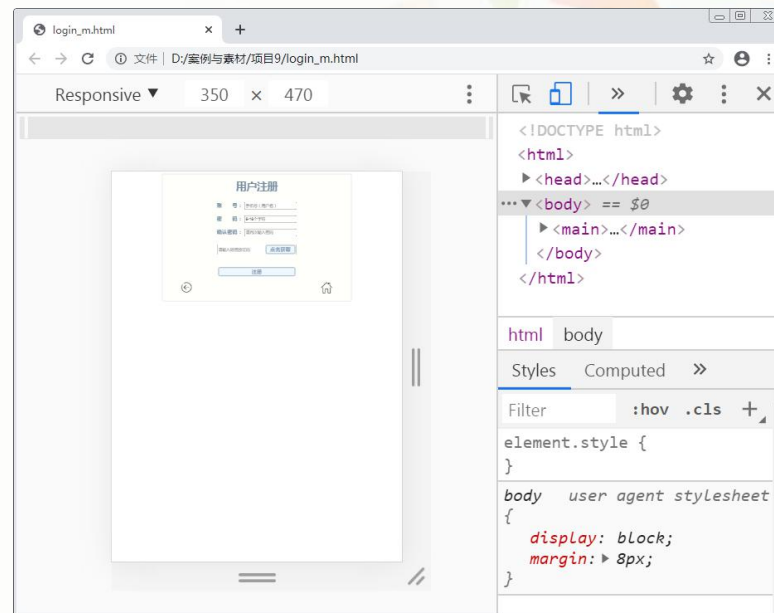


提示：多数浏览器支持在PC端查看移动端网页效果。以谷歌浏览器为例，在浏览器中按“F12”键打开“开发者工具”窗格，单击窗格左上角的按钮，打开移动端模拟器。单击页面左上角的黑色三角按钮，在展开的下拉列表中选择“Responsive”选项，然后在右侧的编辑框中输入合适的屏幕尺寸即可，此处设置屏幕尺寸为350×470。

步骤1



在DW中打开本书配套素材“项目9” → “login_m.html”文档，该文档中已经创建好表单，此时“login_m.html”的移动端页面效果如下图9-17所示。



步骤2



在<head>标签中添加以下代码，设置网页的视口宽度与缩放比例。

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

9.2.2 响应式适配移动设备

1. 使用百分比设置元素宽度

对于一些简单的网页来说，可以通过将其中元素的宽度设置为百分比来适应用移动设备。因为这样一来，当屏幕尺寸发生变化时，各元素的大小也能够同步等比变化。

2. 设置多个样式表文件

在项目6中提到过媒体查询的概念，使用媒体查询可以让不同宽度的浏览器窗口应用不同的页面样式，同样地，也可以让不同宽度的移动设备应用不同的样式。

在实际的网页制作中，通常会根据不同的屏幕宽度创建不同的样式表文件，然后在网页文档的头部标签中添加以下代码，使网页通过媒体查询功能根据屏幕宽度响应式地加载不同的样式表文件。

```
<link rel="stylesheet" type="text/css" media="screen and (max-device-width:300px)" href="css300.css" />
```

其中，media用于设置媒体查询的条件。这段代码的含义是当最大屏幕宽度为300 px时，加载名为“CSS300.css”的样式表文件。



【例9-6】 制作可响应式适配移动端的大学生职业规划调查问卷页面，页面效果如下图所示9-18所示。



The screenshot shows a mobile browser displaying a questionnaire titled "大学生职业规划调查问卷" (University Student Career Planning Questionnaire). The page is titled "个人信息" (Personal Information) and contains two input fields for "姓名" (Name) and "班级" (Class), both marked with an asterisk. Below this is a section for "一、单选题" (I. Single Choice Questions). The first question is "1. 你是否收到学校发的《规划手册》?" (1. Have you received the 'Planning Handbook' issued by the school?). It has two options: "A: 收到" (A: Received) and "B: 未收到" (B: Not received). The second question is "2. 你是否阅读了该手册?" (2. Have you read the handbook?). It has three options: "A: 仔细阅读, 并做详细记录" (A: Read carefully and make detailed records), "B: 粗略阅读" (B: Skimmed), and "C: 没有阅读过" (C: Never read). The third question is "3. 你是否满足自己的专业?" (3. Are you satisfied with your major?). It has four options: "A: 非常满意" (A: Very satisfied), "B: 一般" (B: Average), "C: 不满意" (C: Dissatisfied), and "D: 没想过" (D: Never thought). The fourth question is "4. 你是否有职业方向的规划?" (4. Do you have a career direction plan?).



步骤1



在DW中打开本书配套素材“项目9” → “survey.html”文档，该文档中已经创建好调查问卷的内容，此时的移动端页面效果（如无特殊说明，本节中皆使用iPhone X手机的屏幕尺寸显示移动端页面效果）如图9-19所示（详情见教材）。



步骤2



在<head>标签中添加以下代码，设置页面的视口。

```
<meta name="viewport" content="width=device-width,  
user-scalable=no,initial-scale=1,maximum-scale=1,  
minimum-scale=1,viewport-fit=cover" />
```

步骤3



继续在<head>标签中添加以下代码，设置最大屏幕宽度为375 px时加载“msurvey.css”文档。

```
<link rel="stylesheet" type="text/css" media="screen and  
(max-device-width:375px)" href="msurvey.css" />
```

步骤4



在“survey.html”文档的存储目录下创建“msurvey.css”文档，在其中输入代码（详情见教材），设置移动端页面的样式。

提示：使用<meta>标签的
media属性让网页加载新的样式表文件时，可能需要对某些元素重复设置
样式，此时应尽量使用更高优先级的
选择器。



任务实施

使“反馈意见”页面适配移动端

