第二次作业

教学班1210 杨骏 202013407085

作业仓库：yangjunyjy/homework2(github.com)

第一题：针对第一次作业的“注册会员”页面，添加样式；将第一次的代码进行部分修改，添加css样式文件完成任务

第一部分代码：

<!DOCTYPE html><html lang="en"><head> <meta charset="UTF-8"> <meta name="viewport" content="width=device-width, initial-scale=1.0"> <title>Document</title> <link rel="stylesheet" href="style.css"></head><body> <form action="data.html" method="GET"> <div class="caption"> <h3> <caption>欢迎注册会员</caption> </h3> </div> <div class="Input"> <label>手机号码：<input type="text" name="telephone" maxlength="11" placeholder="11位手机号码"></label> <span>必填</span> </div> <div class="Input"> <label>创建密码：<input type="password" name="pwd" maxlength="8" placeholder="8位密码"></label> <span>必填</span> </div> <div class="email"> <label>注册邮箱: <input type="test" name="email" placeholder="例如：wustzz@sina.com"></label> <span>必填</span> </div> <div class="Input"> <label>&nbsp;&nbsp;&nbsp;验证码：<input type="test" name="password" placeholder="请输入验证码"></label> <i style="background-color: rgb(165, 42, 149);">&nbsp;<label id="code\_box">UJFg</label>&nbsp;&nbsp;</i> <img src="arrow\_03.png" alt="刷新" height="25px" width="25px"> <br> </div> <div class="sex"> <label>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;性别： <input type="radio" name="sex" value="male" checked="checked">男 <input type="radio" name="sex" value="famale">女 </label> </div> <div class="birthday"> <label>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;生日： <input type="date" name="mydate"></label> </div> <div class="age"> <label>&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;年龄： <input type="test" name="age"></label> </div> <div class="Input"> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;籍贯： <select class="jiguan" id="province" onchange="showCity()"> <option value="0" selected="selected">湖北省</option> <option value="1">河南省</option> <option value="2">山东省</option> <option value="3">广东省</option> </select> <select class="jiguan" id="city"> <option value="0">武汉</option> </select> </div> <div class="Input"> 个人学历： <select name="education"> <option value="0">专科</option> <option value="1" selected="selected">本科</option> <option value="2">硕士研究生</option> <option value="3">博士研究生</option> </select> </div> <div class="Input"> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;月薪： <input type="range" style="width: 150px;" id="salary" name="salary" min="0.0" max="10000.0" step="10.0" value="5000.0"> <span id="msg"></span> </div> <div class="favorite"> 个人爱好： <label> <input type="checkbox" name="like" value="singing">唱歌 <input type="checkbox" name="like" value="running">跑步 <input type="checkbox" name="like" value="swimming">游泳 </label> </div> <div class="photo"> 个人照片： <input type="file" id="test-image-file" name="test"> <div id="test-image-preview" style="border: 1px solid silver; width: 220px; height: 120px; background-repeat: no-repeat; background-position: center;"> </div> <p id="test-file-info"></p> </div> <div class="cv"> <span>个人简历:</span> <textarea name="intro" cols="30" rows="10"></textarea> </div> <div class="button"> <input type="submit" value="提交"> &nbsp;&nbsp;&nbsp;&nbsp;&nbsp; <input type="reset" value="重填"> </div> </form> <script> var msg = document.getElementById("msg"); window.onload = function () { msg.innerHTML = document.getElementById("earnings").value; } document.getElementById("earnings").onmousemove = function () { msg.innerHTML = this.value; } </script> <script> var code\_box = document.getElementById("code\_box"); function refreshCode() { //62个字符 随机选择4位 var code = '0123456789qwertyuiopasdfghjklzxcvbnmQWERTYUIOPASDFGHJKLZXCVBNM', char = '', result = ''; for (var i = 0; i < 4; i++) { //随机选择一位 （0,61） 写出0到61的随机的索引数字 var code\_index = Math.round(Math.random() \* 61); //得到随机的索引 取出随机地字符 var char = code[code\_index]; //随机取出的字符 存在几个相同重复的问题 ，而且对于字母，不能区分大小写。 // 避免重复的思路是：取出字符之后,和最后的result对比一下，看看里边是不是已经存在了，如果存在本次循环就终止，进行下一次 if (result.toUpperCase().indexOf(char.toUpperCase()) > -1) //indexOf() == -1 说明结果里边没有要找的字符 那么 > -1 就是 里边有重复的字符 { i--; //为什么会 --？ 因为如果条件成立，那么本轮循环就结束进行下一轮循环（自然i就加1了），那么本轮本应该取出的字符就没有了 //到最后会少一个字符 缺席 continue;//终止本轮循环 进行下一轮 } result += char; } code\_box.innerHTML = result; } //点击事件 code\_box.onclick = refreshCode; </script> <script> function showCity() { //城市数组要与option标签中的省份相对应，因为第一个选项就是“省份”描述，所以第一个数组用空数组即可 var citys = [[], ["郑州", "南阳", "安阳"], ["济南", "青岛", "崂山"], ["广州", "深圳", "中山"]]; //获取用户选择的省份 var provinceNode = document.getElementById("province"); //获取选择的省份的索引 var index = provinceNode.selectedIndex; //根据省份的索引从二维数组中获取到显示的城市数据，showCity是一个一维数组 var showCity = citys[index]; //我们需要把显示的数据显示到城市这个下拉菜单中，首先需要获取到城市下拉菜单的节点，再获取到它的子节点，也就是具体的城市，将他们删除后，再将获取到的信息显示在城市下拉菜单中 //获取到城市下拉菜单的节点 var cityNode = document.getElementById("city"); //获取到城市下拉菜单的子节点，也就是具体的城市选项，得到的是一个子节点数组 var cityChildNodes = cityNode.childNodes; //将所有的cityChilds删除，也就是城市下拉菜单中的所有元素。 //for循环这里没有变量自增，是因为每当删除一个城市后，下一个城市就自动向前，所以只用一直删除第一个城市即可。 for (var i = 0; i < cityChildNodes.length;) { cityNode.removeChild(cityChildNodes[i]); } //将获取的城市显示到城市下拉菜单的option标签中 for (var i = 0; i < showCity.length; i++) { //创建一个option标签用于存放具体的城市 var optionCityNode = document.createElement("option"); //将查询到的城市添加到option标签中 optionCityNode.innerHTML = showCity[i]; //将optionCityNode标签添加到城市下拉菜单中 cityNode.appendChild(optionCityNode); } } </script> <script> var fileInput = document.getElementById('test-image-file'); var info = document.getElementById('test-file-info'); var preview = document.getElementById('test-image-preview'); fileInput.addEventListener('change', function () { // 监听change事件 preview.style.backgroundImage = ''; // 清除背景图片 if (!fileInput.value) { info.innerHTML = '没有选择文件'; return; } var file = fileInput.files[0]; // 获取File引用 // 获取File信息: info.innerHTML = '文件: ' + file.name + '<br>' + '大小: ' + file.size + '<br>' + '修改: ' + file.lastModifiedDate; if (file.type !== 'image/jpeg' && file.type !== 'image/png' && file.type !== 'image/gif') { alert('不是有效的图片文件!'); return; } var reader = new FileReader(); // 读取文件 reader.onload = function (e) { // 发起一个异步操作来读取文件内容 var data = e.target.result; // data串形如： 'data:image/jpeg;base64,/9j/4AAQSk...(base64编码)...' preview.style.backgroundImage = 'url(' + data + ')'; }; reader.readAsDataURL(file); // 以DataURL的形式读取文件 }); </script> <script> var msg = document.getElementById("msg"); window.onload = function () { msg.innerHTML = document.getElementById("salary").value; } document.getElementById("salary").onmousemove = function () { msg.innerHTML = this.value; } </script></body></html>

这里将大部分的div标签里的类设置为input，便于样式管理，剩下的根据英文名称定义，应为大部分都要输入框所以这样设置

Css样式代码：

body { border: 1px solid black;/\*边框定义，1px实体黑色边框\*/ width: 450px;/\*宽度\*/}form {/\*表单\*/ background-color: rgb(241, 236, 236);/\*背景颜色\*/ background-position: center center;/\*背景位置正中\*/ background-attachment: fixed;/\*其余部分滚动时，背景图像不动\*/}div { margin: 5px 15px;/\*外边距上下5px 左右15px\*/ font-size: 1.1rem;/\*1.1倍根元素字体大小\*/}.caption h3 {/\*最上面的标题：欢迎注册会员\*/ text-align: left;/\*文字居左\*/ margin: 5px 15px;/\*外边距上下5px 左右15px\*/ color: rgb(15, 92, 236);/\*颜色：蓝色\*/}.Input input {/\*input类里的input属性大部分是输入框，手机号码，密码，验证码等\*/ width: 200px;/\*宽度\*/ padding-left: 20px;/\*左内边距\*/ margin-left: 10px;/\*左外边距\*/}.email input{/\*邮件输入框\*/ width: 200px;/\*宽度\*/ padding-left: 20px;/\*左内边距\*/ margin-left: 16px;/\*左外边距\*/}.Input span,.email span {/\*input类和邮件类里的span标签，这里是输入框后的必填字样\*/ color: red;/\*红色\*/}#msg {/\*工资进度条id\*/ color: black;}.sex input {/\*性别单选框\*/ width: 50px;}.birthday input,.age input {/\*生日和年龄输入框\*/ width: 200px;/\*宽度\*/ padding-left: 20px;/\*左内边距\*/ margin-left: 5px;/\*左外边距\*/}.favorite input {/\*喜好复选框\*/ width: 20px;}.photo input {/\*照片框\*/ width: 200px; margin-left: 5px;}#test-image-preview {/\*照片演示框\*/ margin-left: 100px;}select {/\*选择框\*/ width: 220px; padding-left: 20px; margin-left: 5px;}select.jiguan {/\*籍贯选择框\*/ width: 105px;}.cv span {/\*个人简历字体标签\*/ position: relative;/\*相对定位\*/ bottom: 75px;/\*底部距离，这样个人简历四个字就会位于文本框中间\*/}.cv textarea {/\*简历文本框\*/ margin-left: 20px;}.button {/\*底部按钮\*/ padding: 0px 100px;/\*内边距上下0左右100px\*/}

运行结果截图：

第一题整体没有问题，就是边框的对齐有些小问题，对每个框设置一样的宽度，邮箱框不一样，通过细微调整基本对齐了。

第二题：对music.css的解读

:root {/\*伪类控制全局\*/

  --primary-color: #31c27c;/\*最初的颜色\*/

}

body {      /\*身体\*/

  margin: 0;/\* 外边距设置为0\*/

  background: #f5f5f5;/\*背景色\*/

  font-family: Microsoft Yahei;/\*字体风格：微软雅黑\*/

  font-size: 14px;  /\*字体大小\*/

  /\* min-width: 1000px; \*/

}

这一部分是全局的控制

a {/\*超链接类\*/

  text-decoration: none;/\*字体装饰：无\*/

  color: #000;/\*颜色\*/

}

超链接标签的控制，页面文字都是用的a标签

header {  /\*头部标签\*/

  height: 90px;/\*高度\*/

  background: white;/\*背景颜色\*/

  display: flex;/\*弹性布局方式，默认横向\*/

  align-items: center;/\*布局内容在中心位置\*/

}

这里是对顶部的控制，对图1中箭头位置QQ音乐图标的控制

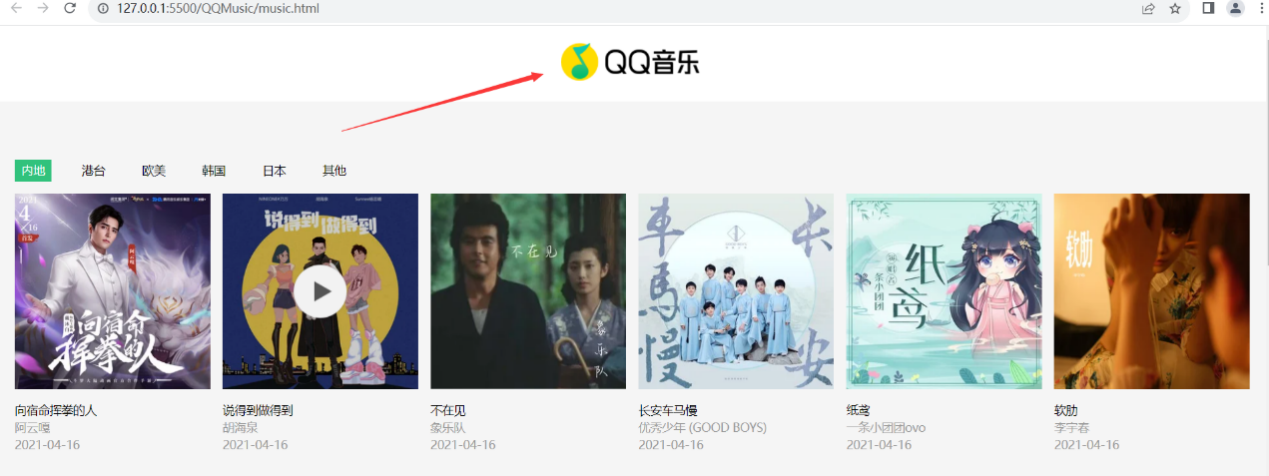


图1

.header-inner, main {

  /\* width: 1200px; \*/

  padding: 20px;/\*内边距\*/

  margin: auto;/\*外边距：自适应\*/

}

Hade-inner里面包含的是图1QQ音乐图标，main是对下面卡片整体的控制

.tag-list {

  margin-top: 50px;/\*顶部外边距\*/

  margin-bottom: 15px;/\*底部外边距\*/

}

.tag-item {

  display: inline-block;/\*行内块级元素\*/

  padding: 0 8px;/\*内边距，上下是0，左右是8px\*/

  margin-right: 24px;/\*右外边距\*/

  line-height: 26px;/\*行高\*/

  color: #000;/\*颜色\*/

  text-decoration: none;/\*无字体装饰\*/

}

.tag-active {

  background: var(--primary-color);/\*背景通过var（）函数调用root里的颜色\*/

  color: white;

}

.tag-item:not(.tag-active):hover {/\*鼠标在tag-item而不是tag-active上悬浮时为绿色\*/

  color: var(--primary-color);

}

Html代码：

<div class="tag-list">

      <a href="#" class="tag-item tag-active">内地</a>

      <a href="#" class="tag-item">港台</a>

      <a href="#" class="tag-item">欧美</a>

      <a href="#" class="tag-item">韩国</a>

      <a href="#" class="tag-item">日本</a>

      <a href="#" class="tag-item">其他</a>

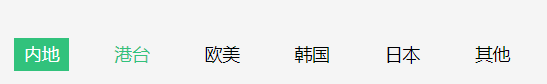


图2.

如图2，上面的样式就是控制它的，.tag-active控制背景色绿色，.tag-item:not(.tag-active):hover表示鼠标悬浮在非内地标签上时文字变成绿色

.nowrap {/\*nowrap强制不换行\*/

  white-space: nowrap;/\*文本不会换行，文本会在在同一行上继续，直到遇到标签为止\*/

  overflow: hidden;/\*隐藏溢出部分\*/

  text-overflow: ellipsis;/\*溢出部分以...形式显示\*/

}

.playlist {

  display: grid;/\*网格布局\*/

  grid-template-columns: repeat(auto-fill, minmax(200px,1fr));/\*尽可能多地放下这些二百像素的网格\*/

  gap: 45px 15px;/\*网格间隙\*/

  margin-bottom: 45px;/\*底部外边距\*/

}

Playlist控制的整个播放页面卡片

.cover-wrap {

  display: flex;/\*弹性布局方式，默认横向\*/

  position: relative;/\*相对定位\*/

  overflow: hidden;/\*隐藏溢出部分\*/

  width: 100%;/\*宽度\*/

  margin-bottom: 15px;/\*底部外边距\*/

  align-items: center;/\*布局内容在中心位置\*/

  justify-content: center;/\*将子元素在主轴上居中对齐\*/

  cursor: pointer;/\*在计算机中将光标呈现为指示链接的指针（一只手）\*/

}

.cover-img {

  width: 100%;/\*宽度\*/

  transition: all 0.75s;/\*过渡时间0.75s\*/

}

.cover-mask {

  position: absolute;/\*绝对定位\*/

  left: 0;

  top: 0;

  right: 0;

  bottom: 0;/\*相对父元素距离都是0\*/

  background: #000;/\*背景色\*/

  opacity: 0;/\*DOM 结构：透明度为 100%，不会让元素从渲染树消失，渲染元素继续占据空间，只是内容不可见\*/

  transition: all 0.5s;/\*过渡时间0.5s\*/

}

.cover-play {

  position: absolute;/\*绝对定位\*/

  left: 50%;

  top: 50%;/\*绝对定位，相对于父元素距离左边和顶部占50%，相当于在中间位置\*/

  margin-left: -35px;/\*外边距向左移动35px\*/

  margin-top: -35px;/\*外边距向上移动35px\*/

  opacity: 0;/\*DOM 结构：透明度为 100%，不会让元素从渲染树消失，渲染元素继续占据空间，只是内容不可见\*/

  transform: scale(0.7);//\*元素缩放为0.7倍\*/

  transition: all 0.5s;/\*过渡时间0.5s\*/

}

.cover-wrap:hover .cover-mask {

  opacity: 0.2;/\*透明度0.2\*/

}

.cover-wrap:hover .cover-play {

  opacity: 1;/\*透明度1，即不透明\*/

  transform: scale(1);/\*缩放为原来的1倍\*/

}

.cover-wrap:hover .cover-img {

  transform: scale(1.07);/\*缩放为原来的1。07倍\*/

}

.album-title:hover {/\*鼠标悬浮在标题上时\*/

  color: var(--primary-color);/\*绿色\*/

}

.album-author {/\*作者\*/

  color: #999;

  line-height: 1.6;/\*行内高度\*/

}

.album-author:hover {/\*鼠标悬浮在作者上时\*/

  color: var(--primary-color);

}

.album-time {

  color: #999;

一部分代码：

<div class="album">

        <a href="#" class="cover-wrap">

          <img src="https://y.gtimg.cn/music/photo\_new/T002R300x300M000000hNAwr4dbutH\_1.jpg" class="cover-img">

          <div class="cover-mask"></div>

          <img src="imgs/cover\_play.png" class="cover-play">

        </a>

        <div>

          <a href="#" class="nowrap album-title">向宿命挥拳的人</a>

        </div>

        <div>

          <a href="#" class="nowrap album-author">阿云嘎</a>

        </div>

        <div class="nowrap album-time">2021-04-16</div>

      </div>

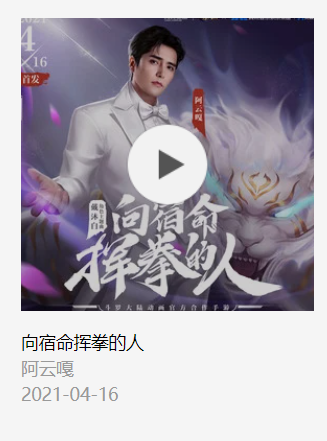




图3.

如图3，这部分代码主要控制播放卡片的样式，当鼠标放在图片位置，图片经过过渡时间后放大一点然后显示出一个播放按钮，鼠标移动到文字时，颜色发生变化，时间的颜色不会变

.page {

  display: flex;/\*弹性布局\*/

  align-items: center;/\*布局内容在中心位置\*/

  justify-content: center;/\*将子元素在主轴上居中对齐\*/

}

.page-item {

  display: inline-block;/\*行内块级元素\*/

  color: #a2a2a2;/\*颜色\*/

  line-height: 50px;/\*行内高度\*/

  padding: 0 20px;/\*内边距上下为0，左右为20px\*/

  margin: 10px;/\*外边距\*/

  font-size: 16px;/\*字体大小\*/

}

.page-current, .page-item:not(.page-more):hover {

  background: var(--primary-color);/\*背景色\*/

  color: white;

}

footer {/\*页脚\*/

  margin-top: 60px;/\*顶部外边距\*/

  background: #333;/\*背景色\*/

  height: 200px;/\*高度\*/

}

Html代码：

.page {

  display: flex;/\*弹性布局\*/

  align-items: center;/\*布局内容在中心位置\*/

  justify-content: center;/\*将子元素在主轴上居中对齐\*/

}

.page-item {

  display: inline-block;/\*行内块级元素\*/

  color: #a2a2a2;/\*颜色\*/

  line-height: 50px;/\*行内高度\*/

  padding: 0 20px;/\*内边距上下为0，左右为20px\*/

  margin: 10px;/\*外边距\*/

  font-size: 16px;/\*字体大小\*/

}

.page-current, .page-item:not(.page-more):hover {

  background: var(--primary-color);/\*绿色

  color: white;

}

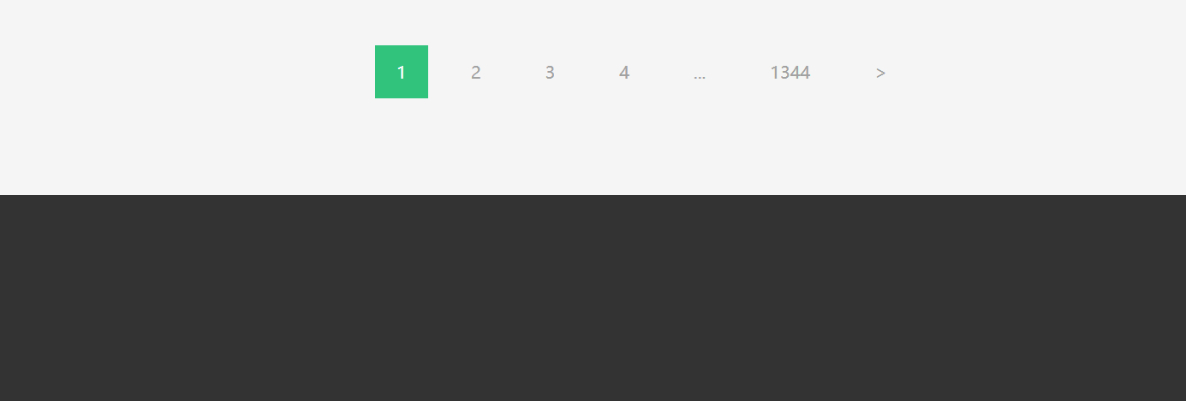
footer {/\*页脚\*/

  margin-top: 60px;/\*顶部外边距\*/

  background: #333;/\*黑色\*/

  height: 200px;/\*高度\*/

}



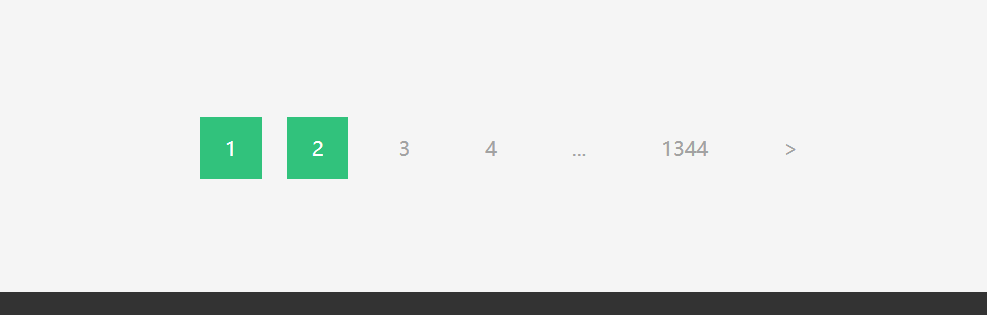


图4.

如图4，这是对底部的设计，鼠标悬停背景会变色，还有页脚也就是最底部是黑色背景色。

本题看代码时还有许多的属性样式没有见过，通过网上查阅资料，对相关属性进行补充，学到了许多东西，css样式还需要继续学习，课上学到的只有一点点东西，需要学习的还有很多。