中南林业科技大学

实验报告

课程名称： J2EE前后端分离

专业班级： 2020级计科二班

姓名： 曾祥文

学号：20202524

2022 年 6月 16 日

实验一 基于CSS+HTML+JS开发简单个人网站

**实验学时：4 实验类型：设计**

1. **目的与任务**

目的：熟悉在静态网页制作基本流程，熟练应用CSS+HTML+JS。

任务：使用CSS+HTML+JS，开发简易个人网站。

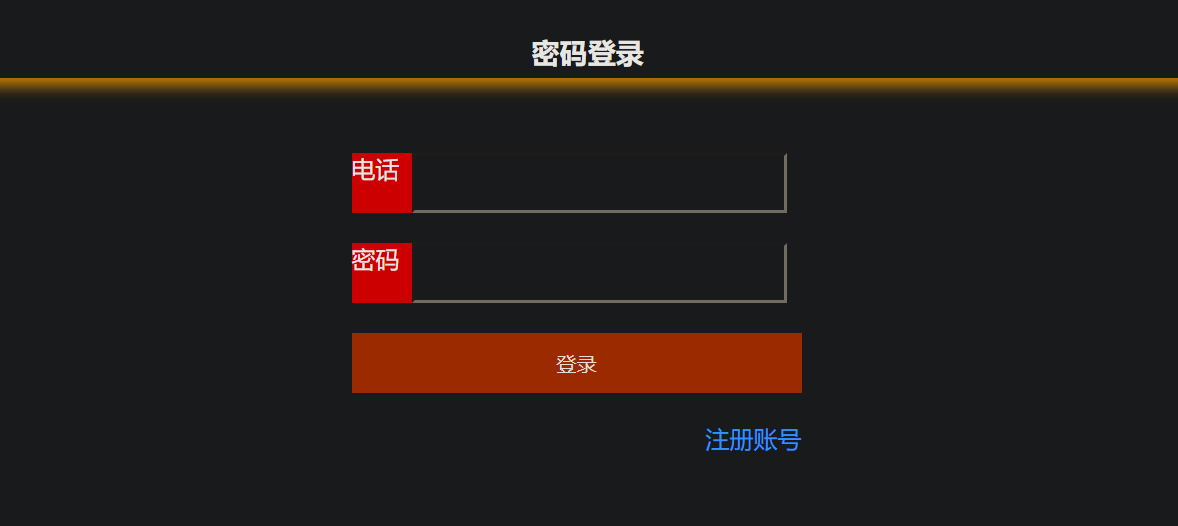
1. **内容、要求与方法步骤**

1、个人网站分为四个部分：注册页面，登录界面，主界面，以及分界面。

1）注册页面需要使用到用户名及密码等信息，用到的标签有表格、文本框、单选按钮、复选按钮、下拉列表框等基本的标签；登录界面需要输入用户名和密码；



2）登陆成功后，可利用javascript判断输入用户名和密码的正确性，输入正确，则利用form标签中action转向主界面，如果输入不正确，可用警告框提醒用户：“输入不正确，再次输入”；



3）主界面类似图1-1，采用四个盒子分别代表个人简历，我的理想，我的生活，学习内容。每个盒子都采用float浮动。

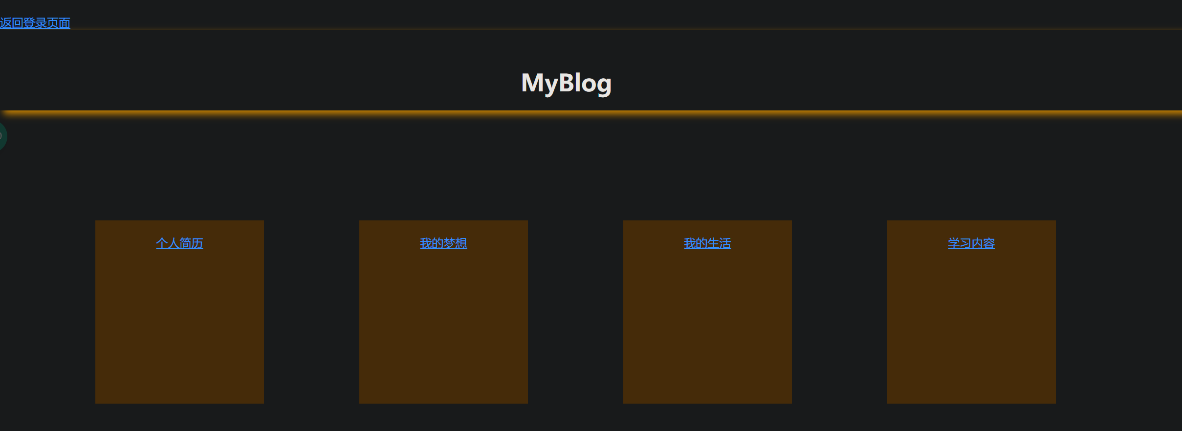


图1-1 主界面

个人简历界面：主要是css3实体样式；同样采用了四个盒子分别介绍四个内容。

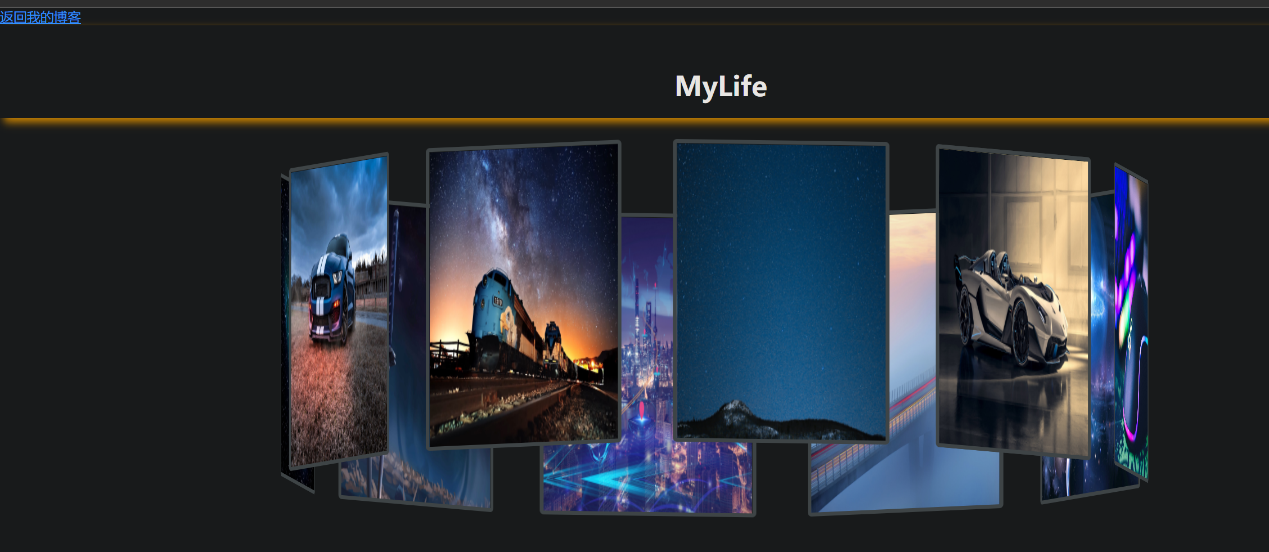


图1-2个人简历界面

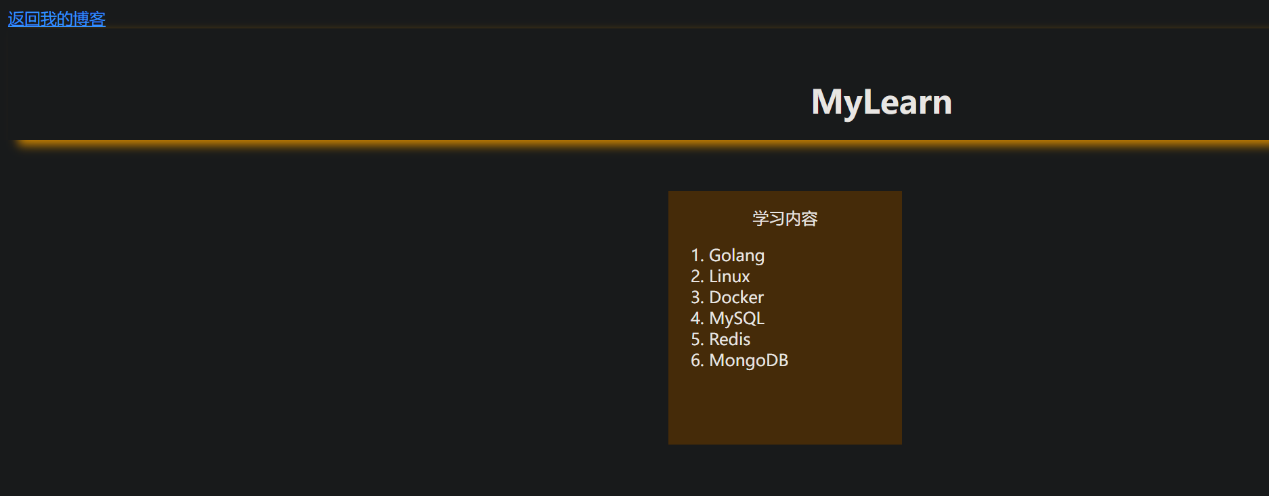
我的理想界面：列举我的理想，可以使用无序列表或者表格排列。



我的生活界面：主要内容采用展示照片墙或者视频来体现。



学习内容界面：采用list-style:none的有序列表。



1. **实验代码**

注册页面 register.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<title>**注册**</title>**
6. **<link** rel="stylesheet" href="../static/css/register.css"**>**
7. **</head>**
8. **<body>**
9. <!--头部-->
10. **<div** id="reg\_header"**>**
11. **<div** class="reg\_h\_center"**>**
12. **<div** class="reg\_h\_left"**>**
13. **<h3>**欢迎注册**</h3>**
14. **</div>**
15. **</div>**
16. **</div>**
17. <!--表单内容-->
18. **<div** class="reg\_back"**>**
19. **<div** class="reg\_left"**>**
20. **<p>**新用户注册**</p>**
21. **<p>**USER REGISTER**</p>**
22. **</div>**
23. **<div** class="reg\_center"**>**
24. **<div** class="reg\_form"**>**
26. **<form** action="http://localhost:9999/regsuccess" method="post"**>**
27. **<table>**
28. **<tr>**
29. **<td** class="td\_left"**><label** for="username"**>**用户名**</label></td>**
30. **<td** class="td\_right"**><input** type="text" name="username" placeholder="请输入用户名"
31. id="username"**></td>**
32. **</tr>**
34. **<tr>**
35. **<td** class="td\_left"**><label** for="password"**>**密码**</label></td>**
36. **<td** class="td\_right"**><input** type="password" name="password" placeholder="请输入密码"
37. id="password"**></td>**
38. **</tr>**
39. **<tr>**
40. **<td** class="td\_left"**><label** for="Email"**>**Email**</label></td>**
41. **<td** class="td\_right"**><input** type="email" name="email" placeholder="请输入Email" id="Email"**>**
42. **</td>**
43. **</tr>**
44. **<tr>**
45. **<td** class="td\_left"**><label** for="rename"**>**姓名**</label></td>**
46. **<td** class="td\_right"**><input** type="text" name="rename" placeholder="请输入真实姓名" id="rename"**>**
47. **</td>**
48. **</tr>**
49. **<tr>**
50. **<td** class="td\_left"**><label** for="Telphone"**>**手机号**</label></td>**
51. **<td** class="td\_right"**><input** type="text" name="telphone" placeholder="请输入您的手机号"
52. id="Telphone"**></td>**
53. **</tr>**
55. **<tr>**
56. **<td** class="td\_left"**><label>**性别**</label></td>**
57. **<td** class="td\_right"**><input** type="radio" name="gender" value="male" checked**>** 男
58. **<input** type="radio" name="gender" value="female"**>** 女
59. **</td>**
60. **</tr>**
62. **<tr>**
63. **<td** class="td\_left"**><label** for="Birthday"**>**出生日期**</label></td>**
64. **<td** class="td\_right"**><input** type="date" name="birthday" id="Birthday"**></td>**
65. **</tr>**
67. **<tr>**
68. **<td** colspan="2" align="center"**><input** type="submit" value="注册" id="btn\_sub" onclick="javascript:jump()"**></td>**
69. **</tr>**
71. **</table>**
72. **</form>**
73. **<script>**
74. function jump(){
75. localStorage.setItem("username",document.getElementById("username").value);
76. localStorage.setItem("password",document.getElementById("password").value);
77. localStorage.setItem("email",document.getElementById("Email").value);
79. var Email = document.getElementById("Email");
80. var User = document.getElementById("username");
81. var Password = document.getElementById("password");
83. if(Email.value === "" || User.value ==="" || Password.value ==="") {
84. alert("邮箱、用户名、或密码不能为空！");
85. return false;
86. }
87. else {
88. alert("注册成功，欢迎进入登录界面!!!");
89. window.open("http://localhost:9999")
90. }
92. }
93. **</script>**
94. **</div>**
95. **<script** type="text/javascript" src="../static/js/register.js"**></script>**
96. **</div>**
97. **<div** class="reg\_right"**>**
98. **<p>**已有账号?**<a** href="/"**>**立即登录**</a></p>**
99. **</div>**
100. **</div>**
101. **</body>**
102. **</html>**

登录页面 login.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<title>**登录**</title>**
6. <!--    <link rel="shortcut icon" href="../static/images/camera.ico">-->
7. **<link** rel="stylesheet" href="../static/css/login.css"**>**
8. **</head>**
10. **<body>**
12. <!--头部-->
13. **<div** id="login\_header"**>**
14. **<div** class="login\_h\_center"**>**
15. **<div** class="login\_h\_left"**>**
16. **<h3>**密码登录**</h3>**
17. **</div>**
18. **</div>**
19. **</div>**

22. <!--中部-->
23. **<div** id="login\_body"**>**
24. **<div** class="login\_b\_center"**>**
25. **<div** class="login\_bg"**>**
26. **<form** action="http://localhost:9999/logsuccess" id="login" method="post"**>**
27. **<div** class="userName"**>**
28. **<span>**电话**</span><input** type="text" id="username\_id" name="tel"**>**
29. **</div>**
31. **<div** class="password"**>**
32. **<span>**密码**</span><input** type="password" id="password\_id" name="psw"**>**
33. **</div>**
35. **<div** class="login\_btn"**>**
36. **<input** type="submit" id="login\_id" value="登录" onclick="javascript:jump()"**>**
37. **</div>**
39. **<div** class="forgot\_password"**>**
40. **<a** href="/register"**>**注册账号**</a>**
41. **</div>**
43. **</form>**
45. **<script>**
46. function jump(){
47. localStorage.setItem("password",document.getElementById("password\_id").value);
48. var Password = document.getElementById("password\_id");
49. if(Password.value ==="") {
50. alert("密码不能为空！");
51. return false;
52. }
53. else {
54. alert("登录成功，欢迎进入博客界面!!!");
55. window.open("http://localhost:9999/index")
56. // window.location.href="http://localhost:9999/index";
57. }
58. }
59. **</script>**
61. **</div>**
62. **</div>**
63. **</div>**
65. **</body>**
66. **</html>**

主界面 index.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<title>**MyBlog**</title>**
6. **<link** rel="stylesheet" href="../static/css/index.css"**>**
7. **</head>**
9. **<body** **>**
10. **<a** href="http://localhost:9999"**>**返回登录页面**</a>**
11. <!--        <button id="re">返回登陆界面</button>-->
12. <!--头部-->
13. **<div** id="index\_header"**>**
14. **<div** class="index\_h\_center"**>**
15. **<div** class="index\_h\_left"**>**
16. **<h3** style="margin-top: 50px; text-align: center; font-size: xx-large;"**>**MyBlog**</h3>**
17. **</div>**
18. **</div>**
19. **</div>**
21. **<div** class="box"**>**
22. **<p** style="text-align: center;"**><a** href="resume.html"**><a** href="http://localhost:9999/resume"**>** 个人简历**</a></a></p>**
23. <!--            <a href="page1.html">查看个人简历</a>-->
24. **</div>**
26. **<div** class="box1"**>**
27. **<p** style="text-align: center;"**><a** href="http://localhost:9999/mydream"**>**我的梦想**</a></p>**
28. <!--            <a href="page2.html"><img src="o2.jpg" width="190px" height="170"></a>-->
29. **</div>**
31. **<div** class="box2"**>**
32. **<p** style="text-align: center;"**><a** href="http://localhost:9999/mylife"**>**我的生活**</a></p>**
33. <!--            <a href="page3.html"><img src="o3.jpg" width="190px" height="170"></a>-->
34. **</div>**
36. **<div** class="box3"**>**
37. **<p** style="text-align: center;"**><a** href="http://localhost:9999/learn"**>**学习内容**</a></p>**
38. <!--            <a href="page4.html"><img src="o4.jpg" width="190px" height="170"></a>-->
39. **</div>**
41. **<script>**
42. var re=document.getElementById('re');
43. re.onclick=function(){
44. location.href='login.html';
45. }
46. **</script>**
48. **</body>**
49. **</html>**

个人简历界面 resume.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<link** rel="stylesheet" href="../static/css/resume.css"**>**
6. **<title>**resume**</title>**
7. **</head>**
8. **<body>**
9. **<a** href="http://localhost:9999/index"**>**返回我的博客**</a>**
10. <!--    <button id="re">返回上一页面</button>-->
11. <!--头部-->
12. **<div** id="resume\_header"**>**
13. **<div** class="resume\_h\_center"**>**
14. **<div** class="resume\_h\_left"**>**
15. **<h3** style="margin-top: 50px; text-align: center; font-size: xx-large;"**>**Resume**</h3>**
16. **</div>**
17. **</div>**
18. **</div>**
20. **<div** class="box"**>**
21. **<p** style="text-align: center;"**>**个人信息**</p>**
22. **<ul** type="square" style="margin-left:23px;"**>**
23. **<li>**姓名：曾祥文**</li>**
24. **<li>**出生年月：2002-08**</li>**
25. **<li>**地址：湖南长沙**</li>**
26. **<li>**学历：本科**</li>**
27. **<li>**民族：汉族**</li>**
28. **</ul>**
29. **</div>**
31. **<div** class="box1"**>**
32. **<p** style="text-align: center;"**>**联系方式**</p>**
33. **<ul** type="square" style="margin-left:23px;"**>**
34. **<li>**qq:2768817839**</li>**
35. **<li>**微信:15211434981**</li>**
36. **<li>**电话:15211434981**</li>**
37. **<li>**学校:**<a** href="https://www.csuft.edu.cn" target="\_blank"**>**中南林业科技大学**</a></li>**
38. **<li>**地址:长沙市天心区**</li>**
39. **</ul>**
40. **</div>**
42. **<div** class="box2"**>**
43. **<p** style="text-align: center;"**>**兴趣爱好**</p>**
44. **<ul** type="square" style="margin-left:23px;"**>**
45. **<li>**打乒乓球**</li>**
46. **<li>**敲代码**</li>**
47. **<li>**看电影**</li>**
48. **<li>**听音乐**</li>**
49. **</ul>**
50. **</div>**
52. **<div** class="box3"**>**
53. **<p** style="text-align: center;"**>**特殊技能**</p>**
54. **<ul** type="square" style="margin-left:23px;"**>**
55. **<li><a** href="https://golang.google.cn/" target="\_blank"**>**Golang**</a></li>**
56. **<li><a** href="https://www.docker.com/" target="\_blank"**>**Docker**</a></li>**
57. **<li><a** href="https://kubernetes.io/" target="\_blank"**>**K8s**</a></li>**
58. **</ul>**
59. **</div>**
60. **</body>**
61. **</html>**

我的梦想界面 mydream.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<link** rel="stylesheet" href="../static/css/mydream.css"**>**
6. **<title>**mydream**</title>**
7. **</head>**
8. **<body>**
9. **<a** href="http://localhost:9999/index"**>**返回我的博客**</a>**
10. **<div** id="mydream\_header"**>**
11. **<div** class="mydream\_h\_center"**>**
12. **<div** class="mydream\_h\_left"**>**
13. **<h3** style="margin-top: 50px; text-align: center; font-size: xx-large;"**>**MyDream**</h3>**
14. **</div>**
15. **</div>**
16. **</div>**
18. **<div** class="box"**>**
19. **<p** style="text-align: center;"**>**我的梦想**</p>**
20. **<ul>**
21. **<li>**梦想，可以天花乱坠**</li>**
22. **<li>**心有多宽，舞台就有多大**</li>**
23. **<li>**心有多高，梦想就有多远**</li>**
24. **</ul>**
25. **</div>**
26. **</body>**
27. **</html>**

我的生活界面 mylife.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<link** rel="stylesheet" href="../static/css/mylife.css"**>**
6. **<title>**mylife**</title>**
7. **</head>**
8. **<body>**
9. **<a** href="http://localhost:9999/index"**>**返回我的博客**</a>**
10. **<div** id="mylife\_header"**>**
11. **<div** class="mylife\_h\_center"**>**
12. **<div** class="mylife\_h\_left"**>**
13. **<h3** style="margin-top: 50px; text-align: center; font-size: xx-large;"**>**MyLife**</h3>**
14. **</div>**
15. **</div>**
16. **</div>**
18. **<div** id="div2"**>**
19. **<div** id="photo\_box"**>**
20. **<img** src="../static/images/001.jpg"**>**
21. **<img** src="../static/images/002.jpg"**>**
22. **<img** src="../static/images/003.jpg"**>**
23. **<img** src="../static/images/004.jpg"**>**
24. **<img** src="../static/images/005.jpg"**>**
25. **<img** src="../static/images/006.jpg"**>**
26. **<img** src="../static/images/007.jpg"**>**
27. **<img** src="../static/images/008.jpg"**>**
28. **<img** src="../static/images/009.jpg"**>**
29. **<img** src="../static/images/010.jpg"**>**
30. **<img** src="../static/images/011.jpg"**>**
31. **</div>**
32. **</div>**
33. **</body>**
35. **</html>**

我的学习界面 learn.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<link** rel="stylesheet" href="../static/css/learn.css"**>**
6. **<title>**learn**</title>**
7. **</head>**
8. **<body>**
9. **<a** href="http://localhost:9999/index"**>**返回我的博客**</a>**
10. **<div** id="learn\_header"**>**
11. **<div** class="learn\_h\_center"**>**
12. **<div** class="learn\_h\_left"**>**
13. **<h3** style="margin-top: 50px;  font-size: xx-large;"**>**MyLearn**</h3>**
14. **</div>**
15. **</div>**
16. **</div>**
18. **<div** class="box"**>**
19. **<p** style="text-align: center;"**>**学习内容**</p>**
20. **<ol>**
21. **<li>**Golang**</li>**
22. **<li>**Linux**</li>**
23. **<li>**Docker**</li>**
24. **<li>**MySQL**</li>**
25. **<li>**Redis**</li>**
26. **<li>**MongoDB**</li>**
27. **</ol>**
28. **</div>**
30. **</body>**
31. **</html>**

# 实验二 基于Vue的图书购物车

**实验学时：4 实验类型：设计**

**一、目的与任务**

目的：熟悉Vue的基本指令，进一步熟练应用CSS+HTML+JS。

任务：使用Vue+CSS+HTML+JS，设计图书购物车。

**二、内容、要求与方法步骤**

1、设计如图1的图书购物车。



图1



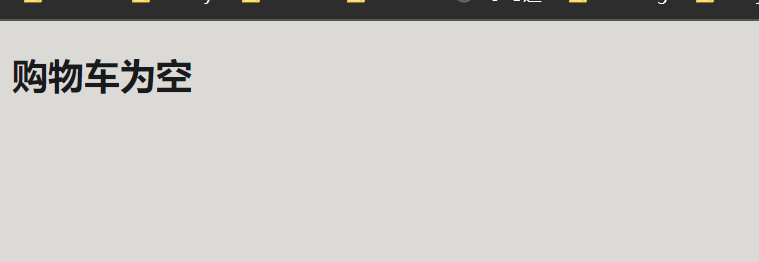
1. 点击移除按钮，则移除一行记录的显示；



1. 点击“+”或者“-”按钮，则购买数量发生相应的变化，且总价也随之发生变化；



1. 当页面所有记录移除完，则在页面上显示“购物车为空”。



4）当购买数量为“0”时，“-”按钮变成不可使用。

具体操作步骤：

1）设计一个main.js文件保存图1中4本书信息，编号，书籍名称，出版日期，价格，购买数量默认值为1。在main.js中已保存基本信息，需要在method中添加购买数量发生加1和减1以及移除（可使用数组的删除方法：.splice()）这三个操作方法。添加computed中的方法，实现总价格计算(可采用传统写法或者利用高阶函数reduce()实现)。

2）表格样式可设计一个.css文件，在主页面index.html中引入进来。

3）在主页面index.html中合适位置引入外部main.js文件，利用v-for指令实现表格内容的展示。

4）在“+”或者“-”按钮添加相应的监听指令，同时在“-”按钮中绑定属性：v-bind: disabled=“购买数量<=0”

5）在移除按钮中添加相应的监听指令。

6）打开浏览器，观察各个页面运行结果。

2、实验安排方式：上机编码。

3、实验结果展示，包括代码（可分栏展示）和效果图。

**三、实验代码**

**index.html**

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<title>**图书购物车**</title>**
6. **<link** rel="stylesheet" href="main.css"**>**
7. **<script** src="https://cdn.jsdelivr.net/npm/vue@2.6.14/dist/vue.js"**></script>**
8. **</head>**
9. **<body>**
10. **<div** id="app" v-cloak**>**
11. **<div** class="ui container" v-if="books.length>0"**>**
12. **<table** class="ui table"**>**
13. **<thead>**
14. **<tr>**
15. **<th></th>**
16. **<th>**书籍名称**</th>**
17. **<th>**出版日期**</th>**
18. **<th>**价格**</th>**
19. **<th>**购买数量**</th>**
20. **<th>**操作**</th>**
21. **</tr>**
22. **</thead>**
24. **<tbody>**
25. **<tr** v-for="(item,index) in books"**>**
26. **<th** **>**{{item.id}}**</th>**
27. **<th** **>**{{item.name}}**</th>**
28. **<th** **>**{{item.date}}**</th>**
29. **<th** **>**{{item.price | getFinalPrice}}**</th>**
30. **<th** **>**
31. **<button** type="button" @click="sub(index)" :disabled="item.count<=1"**>**-**</button>**
32. {{item.count}}
33. **<button** type="button" @click="add(index)"**>**+**</button>**
34. **</th>**
35. **<th><button** type="button" @click="remove(index)"**>**移除**</button></th>**
36. **</tr>**
37. **</tbody>**
39. **</table>**
40. **<h2** class="ui header"**>**总价格:{{finalPrice}}**</h2>**
41. **</div>**
42. **<div** class="ui container" v-else**><h2** class="ui header"**>**购物车为空**</h2></div>**
43. **</div>**
44. **<script** src="main.js"**></script>**
45. **</body>**
46. **</html>**

**main.css**

1. table {
2. border: 10px solid #e9e9e9;
3. border-collapse: collapse;
4. border-spacing: 0;
5. }
7. th, td {
8. padding: 8px 16px;
9. border: 10px solid #e9e9e9;
10. text-align: left;
11. }
12. th {
13. background-color: #f7f7f7;
14. color: #f60202;
15. font-weight: 600;
16. }

**main.js**

1. const app = new Vue({
2. el: '#app',
3. data: {
4. books: [
5. {
6. id: 1,
7. name: '《算法导论》',
8. date: '2006-9',
9. price: 85.00,
10. count: 1
11. },
12. {
13. id: 2,
14. name: '《UNIX编程艺术》',
15. date: '2006-2',
16. price: 59.00,
17. count: 1
18. },
19. {
20. id: 3,
21. name: '《编程珠玑》',
22. date: '2008-10',
23. price: 39.00,
24. count: 1
25. },
26. {
27. id: 4,
28. name: '《代码大全》',
29. date: '2006-3',
30. price: 128.00,
31. count: 1
32. },
33. ]
34. },
35. methods: {
36. add:function(i){
37. this.books[i].count++
38. },
39. sub:function(i){
40. this.books[i].count--;
41. },
42. remove:function(i){
43. this.books.splice(i,1);
44. }
45. },
46. filters:{//过滤器
47. getFinalPrice:function(price){
48. return '￥'+price.toFixed(2);
49. }
50. },
51. computed: {
52. finalPrice:function(){
53. let sum=0;
54. for(let i=0;i**<this.books.length**;i++){
55. sumsum=sum+this.books[i].price\*this.books[i].count
56. }
57. return sum
58. }
59. }
60. })

**实验三 学习Vue-router的使用**

**实验学时：4 实验类型：设计**

**一、目的与任务**

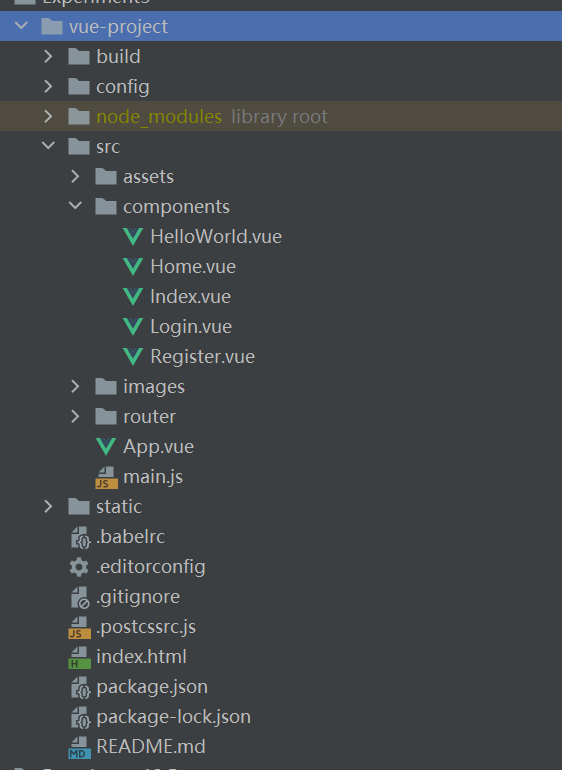
目的：学习基于Vue-cli中的Vue-router的使用

任务: 完成基于Vue-cli中的Vue-router的使用

**二、内容、要求与安排方式**

1、实验内容与要求

创建vue-cli的项目



项目构建成功之后通过运行指令使用项目



在views下创建test.vue页面添加到App.vue通过router的语法

然后通过router文件夹下的index.js配置路由然后在也面点击跳转到路由test页面



要求通过该实验的练习，学习并会使用Vue-router的基本语法，以及路由跳转和路由嵌套





具体操作步骤：

1)通过Vue-cli构建创建项目

2)通过指令运行Vue-cli脚手架

3)创建test.vue的页面  
 4)配置路由设置在主页面进行跳转看是否跳转成功，至少设计3个组件，组件内容自定。

1. 实验安排方式：上机编码。

**三、实验代码**

主界面 index.vue

1. **<template>**
2. **<div>**
3. **<img** src="../assets/logo.png"**>**
4. **<h1** style="position: absolute;left: 730px; top: 215px;"**>**首页**</h1>**
5. <!--  <router-view></router-view>-->
6. **<ul** style="position: absolute;left: 655px;top: 310px"**>**
7. **<span><router-link** to="/index/login"**>**登录页面**</router-link></span>**
8. **<span><router-link** to="/index/register"**>**注册页面**</router-link></span>**
9. **</ul>**
10. **<router-view></router-view>**
11. **</div>**
13. **</template>**
15. **<script>**
16. export default {
17. name: "Index"
18. }
19. **</script>**
21. **<style** scoped**>**
22. .router-link-active{
23. color: red;
24. }
25. **</style>**

嵌套组件一login.vue

1. **<template>**
2. **<div** id="login"**>**
3. **<h3>**{{msg}}**</h3>**
4. **<br>**
5. **<img** src="../images/login.png"**>**
6. **</div>**
7. **</template>**
9. **<script>**
10. export default {
11. name: "Login",
12. data(){
13. return {
14. msg:'密码登录'
15. }
16. }
17. }
18. **</script>**
20. **<style** scoped**>**
22. **</style>**

嵌套组件二 register.vue

1. **<template>**
2. **<div** id="register"**>**
3. **<h3>**{{msg}}**</h3>**
4. **<br>**
5. **<img** src="../images/register.png"**>**
6. **</div>**
7. **</template>**
9. **<script>**
10. export default {
11. name: "Login",
12. data(){
13. return {
14. msg:'欢迎注册'
15. }
16. }
17. }
18. **</script>**
20. **<style** scoped**>**
22. **</style>**

路由配置 index.js

1. //引入Vue框架
2. **import** Vue from "vue";
3. //引入vue-router
4. **import** Router from 'vue-router'
6. **import** HelloWorld from "../components/HelloWorld";
7. **import** Home from "../components/Home";
8. **import** Index from "../components";
9. **import** Login from "../components/Login";
10. **import** Register from "../components/Register";
12. //全局使用router
13. Vue.use(Router)
15. //定义路由配置
16. **export** **default** **new** Router ({
17. mode:'history',
18. routes:[
19. {
20. path:'/hello',
21. name:'HelloWorld',
22. component:HelloWorld
23. },
24. {
25. path: '/',
26. name: "Home",
27. component: Home
28. },
29. {
30. path:'/index',
31. name:'Index',
32. component:Index,
33. children:[
34. {
35. path:'login',
36. name:'Login',
37. component:Login
38. },
39. {
40. path:'register',
41. name:'Register',
42. component:Register
43. }
44. ]
45. }
46. ]
47. })

**实验四 基于IDEA+SpringBoot+Mave+Thymeleaf的系统实现**

**实验学时：2 实验类型：设计**

**一、目的与任务**

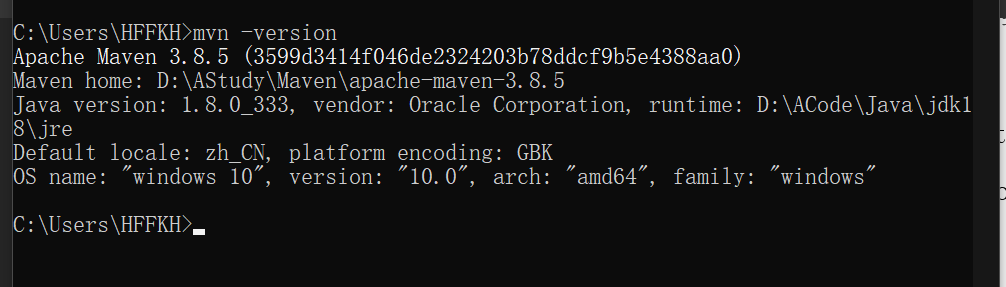
目的：学习IDEA+SpringBoot+maven的系统实现

任务 搭建好IDEA+SpringBoot+maven的系统环境

**二、内容、要求与安排方式**

1、实验内容与要求：

下载好IntelliJ IDEA开发工具并安装配置。安装成功后输入mvn -version命令查看是否安装成功。



下载maven并配置maven环境，添加阿里云镜像配置。将IDEA中的maven配置为本地已安装好的maven。

掌握SpringBoot项目的三种创建方式。

pom.xml中引入thymeleaf依赖，在classpath:/templates/下新建HTML页面(可采用实验一创建的页面)，新建controller进行测试，访问页面后返回success成功数据。



要求通过该实验的练习，理解掌握idea工具的基本使用和SpringBoot框架的基本内容；同时掌握和thymeleaf的基本语法。

2、实验安排方式：上机编码。

**三、实验代码**

controller测试 **maincontroller.java**

1. **package** test.test.com.exp4.Controller;
3. **import** org.springframework.stereotype.Controller;
4. **import** org.springframework.ui.Model;
5. **import** org.springframework.web.bind.annotation.GetMapping;
7. **import** java.util.ArrayList;
8. **import** java.util.List;
10. @Controller
11. **public** **class** MainController {
12. @GetMapping("/main")
13. **public** String main(Model model){
14. List<Content> list = **new** ArrayList<>();
15. String[] strings = {
16. "Golang",
17. "Linux",
18. "Docker",
19. "MySQL",
20. "Redis"
21. };
22. **for**(**int** i=0;i<strings.length;++i){
23. Content content = **new** Content();
24. content.setGoal(strings[i]);
25. list.add(content);
26. }
27. model.addAttribute("list",list);
28. **return** "main";
29. }
30. }
31. **class** Content{
32. **private** String content;
34. **public** String getContent() {
35. **return** content;
36. }
38. **public** **void** setGoal(String content) {
39. **this**.content = content;
40. }
41. }

**返回成功的界面 main.html**

1. <!DOCTYPE html**>**
2. <!-- suppress ThymeleafVariablesResolveInspection -->
3. **<html** xmlns:th="http://www.thymeleaf.org"**>**
4. **<html** lang="en"**>**
5. **<head>**
6. **<meta** charset="UTF-8"**>**
7. **<link** rel="stylesheet" href="/css/learn.css"**>**
8. **<title>**learn**</title>**
9. **</head>**
10. **<body>**
11. **<a** href="http://localhost:8080/index"**>**返回我的博客**</a>**
12. **<div** id="learn\_header"**>**
13. **<div** class="learn\_h\_center"**>**
14. **<div** class="learn\_h\_left"**>**
15. **<h3** style="margin-top: 50px;  font-size: xx-large;"**>**MyLearn**</h3>**
16. **</div>**
17. **</div>**
18. **</div>**
20. **<div** class="box"**>**
21. <!--    <p style="text-align: center;">学习内容</p>-->
22. **<ol>**
23. <!--/\*@thymesVar id="list" type="test/"\*/-->
24. **<li** th:each="item:${list}" th:text="${item.content}"**></li>**
25. **</ol>**
26. **</div>**
28. **</body>**
29. **</html>**

**实验五 SpringBoot整合Redis点赞功能的简单实现**

**实验学时：2 实验类型：设计**

**一、目的与任务**

目的：学习SpringBoot整合redis简单点赞功能

任务 通过给一篇文章点赞，点赞成功+1,取消点赞-1，数据都放在reids缓存里

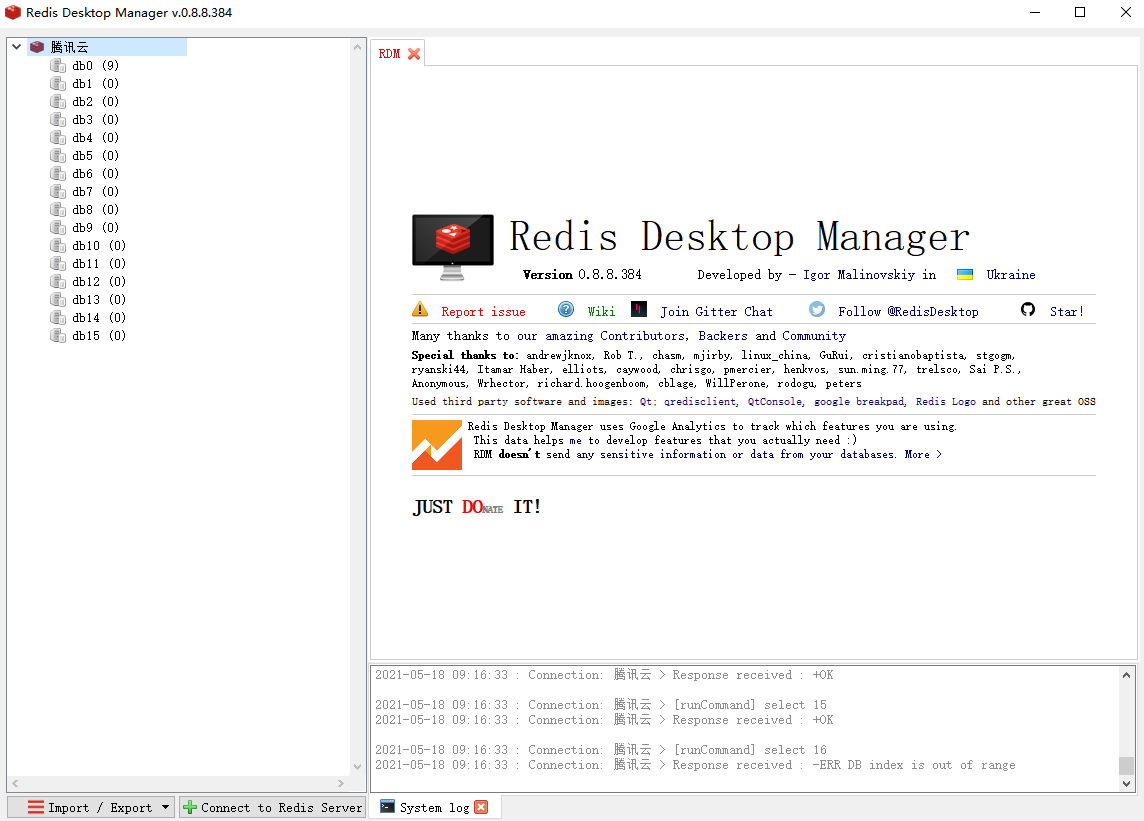
**二、内容、要求与安排方式**

1、实验内容与要求：

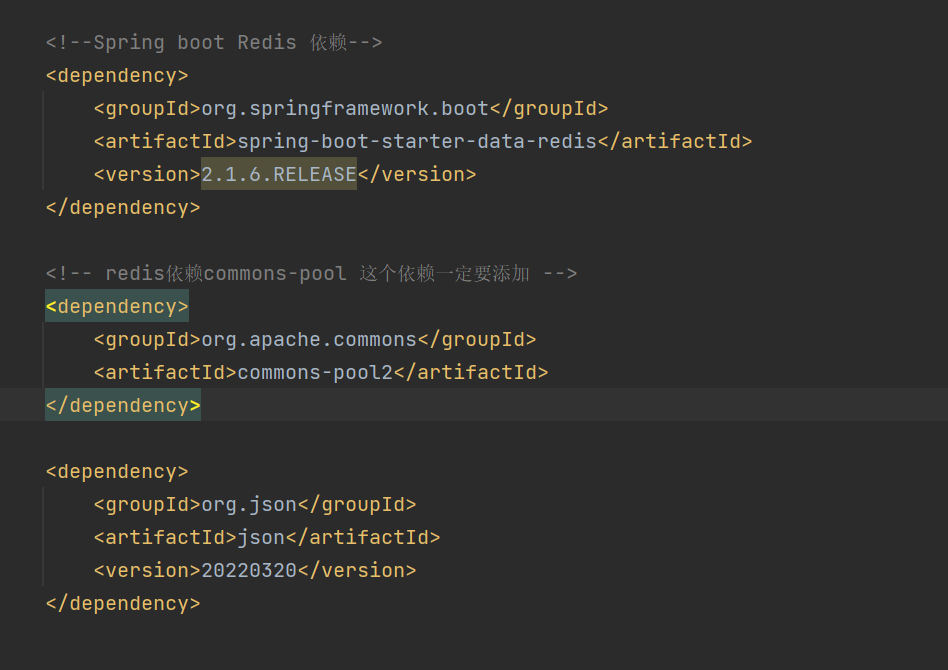
下载redis可以使用windows版也可以是linux版的(如果有虚拟机或者云服务器的同学可以尝试一下)。

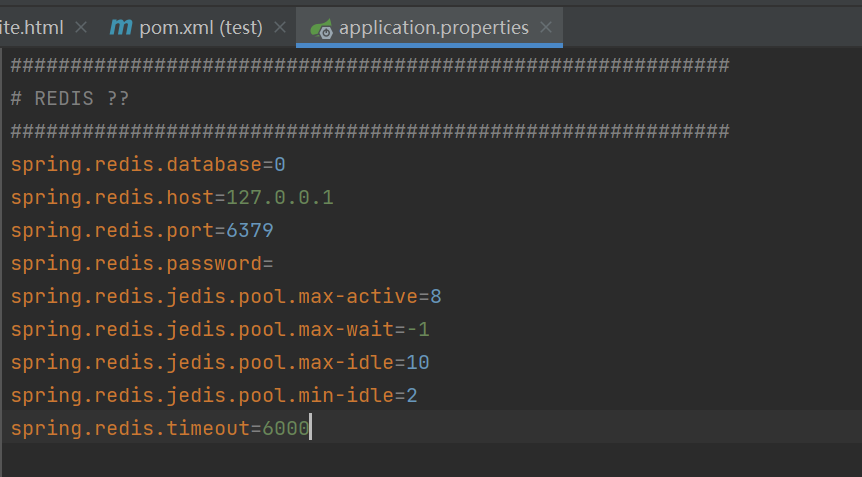
安装redis并掌握redis的基本使用。

可以下载RedisDesktopManager去直接操作redis数据。RedisDesktopManager界面如下：

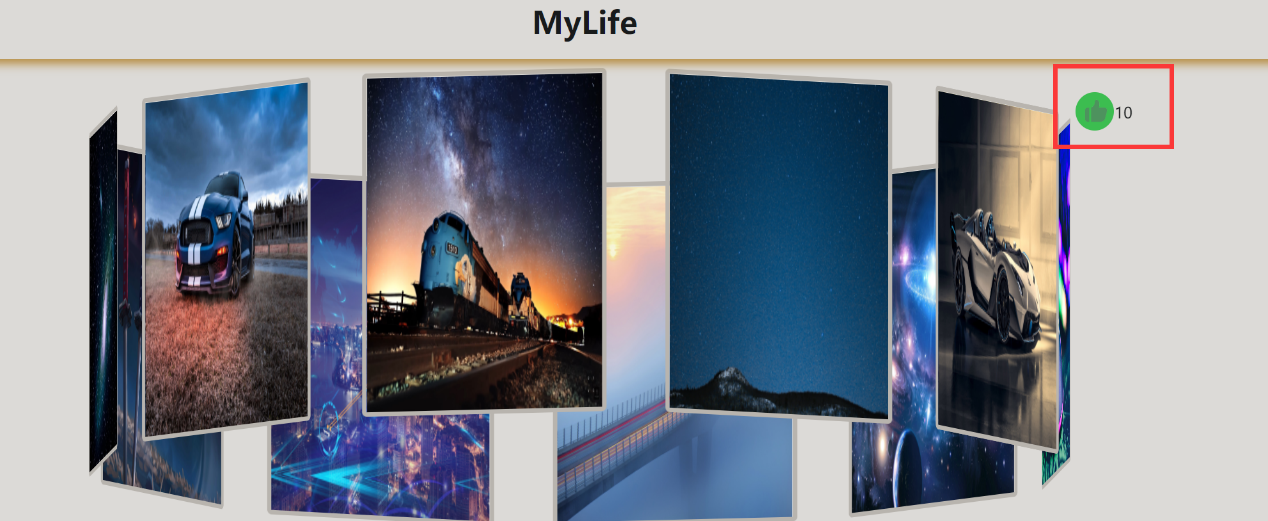


引入Springboot整合redis的依赖，并在配置文件中配置redis相关属性。

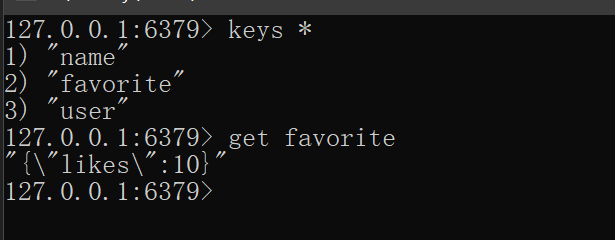




模拟实现文章点赞功能，选用合适的redis数据类型，判断如果当前未点赞，则存入缓存，如果已经存在点赞，那么就删除这条缓存，从而实现前端的点赞功能，根据后端返回1(true)或0(false)去控制。



redis的数据变化



2、实验安排方式：上机编码。

**三、实验代码**

container 层的favoritecontroller.java

1. package test.test.com.exp4.Controller;
3. import org.json.JSONObject;
4. import org.springframework.boot.SpringApplication;
5. import org.springframework.boot.autoconfigure.SpringBootApplication;
6. import org.springframework.data.redis.core.RedisTemplate;
7. import org.springframework.web.bind.annotation.GetMapping;
8. import org.springframework.web.bind.annotation.RequestParam;
9. import org.springframework.web.bind.annotation.RestController;

12. @SpringBootApplication
13. @RestController
14. **public** **class** Application {
16. **public** **static** **void** main(String[] args) {
17. SpringApplication.run(Application.**class**, args);
18. }
19. @GetMapping("/favorite")
20. **public** String favorite(@RequestParam(value = "webpage") String webpage) {
21. RedisTemplate redisTemplate = Redis.redis;
22. Object object = redisTemplate.opsForValue().get(webpage);
23. JSONObject node = **new** JSONObject(object.toString());
24. node.put("favorites",node.getInt("favorites")+1);
25. redisTemplate.opsForValue().set(webpage,node.toString());
26. **return** "Done";
27. }
28. @GetMapping("/unfavorite")
29. **public** String unfavorite(@RequestParam(value = "webpage") String webpage) {
30. RedisTemplate redisTemplate = Redis.redis;
31. Object object = redisTemplate.opsForValue().get(webpage);
32. JSONObject node = **new** JSONObject(object.toString());
33. node.put("favorites",node.getInt("favorites")-1);
34. redisTemplate.opsForValue().set(webpage,node.toString());
35. **return** "Done";
36. }
37. @GetMapping("/Seefavorite")
38. **public** **int** Seefavorite(@RequestParam(value = "webpage") String webpage) {
39. RedisTemplate redisTemplate = Redis.redis;
40. Object object = redisTemplate.opsForValue().get(webpage);
41. **if**(object == null){
42. JSONObject newJson = **new** JSONObject();
43. newJson.put("favorites",0);
44. redisTemplate.opsForValue().set(webpage,newJson.toString());
45. **return** 0;
46. }
47. JSONObject node = **new** JSONObject(object.toString());
48. System.out.println(node.get("favorites"));
49. **return** node.getInt("favorites");
50. }
51. }

点赞页面 favorite.html

1. <!DOCTYPE html**>**
2. **<html** lang="en"**>**
3. **<head>**
4. **<meta** charset="UTF-8"**>**
5. **<link** rel="stylesheet" href="/css/favorite.css"**>**
6. **<title>**mylife**</title>**
7. **</head>**
8. **<body>**
9. **<a** href="http://localhost:8080/index"**>**返回我的博客**</a>**
10. **<div** id="mylife\_header"**>**
11. **<div** class="mylife\_h\_center"**>**
12. **<div** class="mylife\_h\_left"**>**
13. **<h3** style="margin-top: 50px; text-align: center; font-size: xx-large;"**>**MyLife**</h3>**
14. **</div>**
15. **</div>**
16. **</div>**
18. **<div** id="div2"**>**
19. **<div** id="photo\_box"**>**
20. **<img** src="/images/001.jpg"**>**
21. **<img** src="/images/002.jpg"**>**
22. **<img** src="/images/003.jpg"**>**
23. **<img** src="/images/004.jpg"**>**
24. **<img** src="/images/005.jpg"**>**
25. **<img** src="/images/006.jpg"**>**
26. **<img** src="/images/007.jpg"**>**
27. **<img** src="/images/008.jpg"**>**
28. **<img** src="/images/009.jpg"**>**
29. **<img** src="/images/010.jpg"**>**
30. **<img** src="/images/011.jpg"**>**
31. **</div>**
32. **<div** style=" position:absolute;width:100px;height:50px;left:80%;top:20%;"**>**
33. **<div** id="likeDiv" class="like unliked"**></div>**
34. **<img** id="likeIcon" onclick="btn\_like()" class="unlikedIcon"
35. src="/images/favoriteicon.png"
36. style="margin-left: 4px;width:32px;height:32px;vertical-align: middle;"**>**
37. **<p** id="likeCount" style="display:inline-block;vertical-align: middle;"**>**---**</p>**
38. **</div>**
39. **</div>**
40. **</body>**
41. **<script** src="/js/LikeButton.js"**></script>**
42. **<script>**
43. var currentPage = "favorite"
45. function btn\_like() {
46. onClickLike(currentPage);
47. }
48. window.onload=function(){
49. var collection = document.getElementsByClassName("box");
50. var i;
51. for(i=0;i**<collection.length**;i++){
52. collection[i].classList.add("box"+(i+1));
53. }
54. // 加载点赞数
55. LoadLikeCount(currentPage);
56. }

59. **</script>**
60. **</html>**

redis配置 application.properties

1. ############################################################
2. # REDIS ??
3. ############################################################
4. spring.redis.database=0
5. spring.redis.host=127.0.0.1
6. spring.redis.port=6379
7. springspring.redis.password=
8. spring.redis.jedis.pool.max-active=8
9. spring.redis.jedis.pool.max-wait=-1
10. spring.redis.jedis.pool.max-idle=10
11. spring.redis.jedis.pool.min-idle=2
12. spring.redis.timeout=6000