# 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



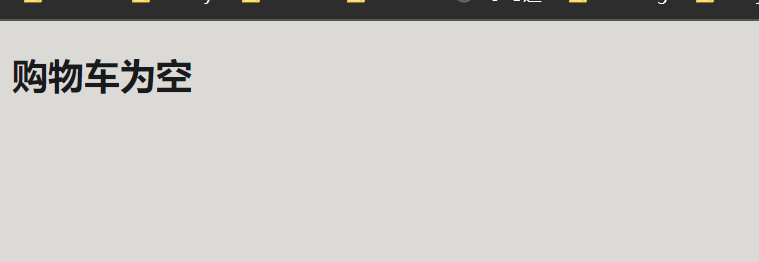
1. 点击移除按钮，则移除一行记录的显示；



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. }
13. th {
14. background-color: #f7f7f7;
15. color: #f60202;
16. font-weight: 600;
17. }

**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: {
37. add:function(i){
38. this.books[i].count++
39. },
40. sub:function(i){
41. this.books[i].count--;
42. },
43. remove:function(i){
44. this.books.splice(i,1);
45. }

48. },
49. filters:{//过滤器
50. getFinalPrice:function(price){
51. return '￥'+price.toFixed(2);
52. }
53. },
54. computed: {
55. finalPrice:function(){
56. let sum=0;
57. for(let i=0;i**<this.books.length**;i++){
58. sumsum=sum+this.books[i].price\*this.books[i].count
59. }
60. return sum
61. }
62. }
63. })