LAB ASSIGNMENT-7

1. Develop prototype 3 continuing with the last lab.

Confirm that the app now remembers your list even after a page refresh.

. .

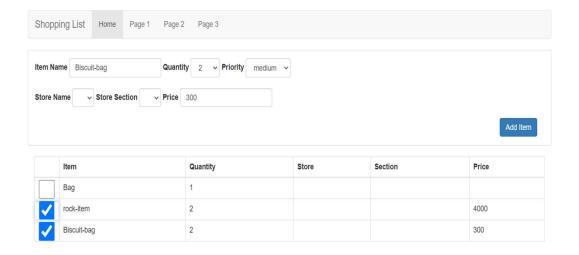
CODE:-

```
♦ lab7.html ×
                           37
38
39
40
41
42
43
44
45
46
47
48
49
50
                                   <option value="5">5</option>
<option value="6">6</option>
                                   <option value="9">9</option>
<option value="10">10</option>
                           <option value="high">high</option>
                                           ♦ lab6.html
                                                          lab7.html X
                            C 1.c
🜣 lab7.html > 🔗 html > 🤣 body > 🔗 div.container > 🤡 div.panel.panel-default.col-md-12 > 🤣 div.panel-body > 🛠 div.row.form-inline > 🚱 div.form-group > 🤡 label
                    var stores = ['Fareway', 'Ace Hardware', 'Caseys', 'The Hatchery', 'Amundsens']
             var sections = ['Produce', 'Meats', 'Cereal', 'Canned Goods', 'Frozen Foods', 'Dairy', 'Liquor', 'Tools', 'Clothing']
             function clickedon() {
                 let rowcolids = ['itemname', 'qty', 'store', 'category', 'price']
                 let vals = []
                 for (let cid of rowcolids) {
                    vals.push(document.getElementById(cid).value)
                 makeRow(vals, document.getElementById('shoppinglist'))
                 saveList();
             function makeRow(valueList, parent) {
                 let row = document.createElement("tr")
                 row.classList.add(document.getElementById("priority").value)
                 let cb = document.createElement("input")
```

```
♦ lab6.html
                                                               lab7.html X
🜣 lab7.html 🗦 🤣 html 🗦 🥩 body 🖒 🤡 div.container 🗦 🤡 div.panel-panel-default.col-md-12 🕻 Ø div.panel-body 🕻 Ø div.row.form-inline 🖒 🍪 div.form-group 🕻 Ø label
              function makeRow(valueList, parent) {
                  let row = document.createElement("tr")
                  row.classList.add(document.getElementById("priority").value)
                  let cb = document.createElement("input")
                  cb.classList.add("form-control")
                  row.appendChild(cb)
                  for (let val of valueList) {
                      let td = document.createElement("td")
                      td.innerHTML = val
                      row.appendChild(td)
                  parent.appendChild(row)
              function populateSelect(selectId, sList) {
                  let sel = document.getElementById(selectId)
                  for (let s of sList) {
                      let opt = document.createElement("option")
                      opt.innerHTML = s
                      sel.appendChild(opt)
              function saveList() {
                  let table = document.getElementById('shoppinglist');
                  let items = [];
```

```
| Isabel | I
```

Output:-



MVC Shopping List with localStorage

Enter item... Add Item

Items:

- rrr
- nine pointer
- sudhanshu
- 2. Create a local storage that saves the number of times you have accessed the page and displays it.

Code:-

```
lab7.html >
     <html lang="en">
      <meta charset="UTF-8">
       <meta name="viewport" content="width=device-width, initial-scale=1.0">
      <title>Shopping List</title>
        body {
          font-family: Arial, sans-serif;
         #shopping-list {
          margin: 20px;
          list-style-type: none;
           padding: 0;
         #items li {
          margin-bottom: 5px;
     <div id="shopping-list">
      <h1>Shopping List</h1>
      <input type="text" id="item" placeholder="Enter item">
<button id="addItemBtn">Add Item</button>
      You have accessed this page <span id="pageAccessCount"></span> times.
```

```
class Model {
    constructor() {
        this.items = JSON.parse(localStorage.getItem('shoppingList')) || [];
        this.accessCount = localStorage.getItem('accessCount') || 0;

        this.accessCount = localStorage.getItem('accessCount') || 0;

        this.accessCount() {
        this.items.push(item);
        this.save();

        }

        getItems() {
            return this.items;
        }

        save() {
            localStorage.setItem('shoppingList', JSON.stringify(this.items));
        }

        incrementAccessCount() {
            this.accessCount++;
            localStorage.setItem('accessCount', this.accessCount);
        }

        getAccessCount() {
            return this.accessCount;
        }

        getAccessCount() {
            return this.accessCount;
        }

    }
}
```

```
this.itemsList = document.getElementById('items');
  this.itemInput = document.getElementById('item');
  this.addItemBtn = document.getElementById('addItemBtn');
 this.pageAccessCount = document.getElementById('pageAccessCount');
getNewItem() {
return this.itemInput.value.trim();
clearInput() {
this.itemInput.value = '';
renderItems(items) {
 this.itemsList.innerHTML = '';
 items.forEach(item => {
   const li = document.createElement('li');
   li.textContent = item;
   this.itemsList.appendChild(li);
renderAccessCount(count) {
this.pageAccessCount.textContent = count;
```

```
constructor(model, view) {
    this.model = model;
    this.view = view;
    this.view.addItemBtn.addEventListener('click', () => this.addItem());
    this.initialRender();
  initialRender() {
    this.view.renderItems(this.model.getItems());
    this.view.renderAccessCount(this.model.getAccessCount());
    this.model.incrementAccessCount();
  addItem() {
    const newItem = this.view.getNewItem();
      this.model.addItem(newItem);
      this.view.renderItems(this.model.getItems());
      this.view.clearInput();
const model = new Model();
const view = new View();
const controller = new Controller(model, view);
```

Output:-

Shopping List



You have accessed this page 1 times.

Shopping List



car

truck

You have accessed this page 4 times.