

1)

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Document</title>
7 </head>
8 <body>
9   <h1>
10    | Hello Welcome
11   </h1>
12   <p id="essay"></p>
13   <button onclick="updateCount()">Click me</button>
14
15   <script src="script.js"></script>
16 </body>
17 </html>
```

```
function getCount(){
    if(localStorage.getItem('count')){
        | return parseInt(localStorage.getItem('count'))
    }

    else{
        | localStorage.setItem('count','0');
        | return 0;
    }
}

function updateCount() {
    var count = getCount();

    count++;
    localStorage.setItem('count',count.toString()); /// localStorage handles all data in Strings

    document.getElementById('essay').textContent = "you have clicked the button" + " " + count + " " + "times"
}

function clearCount(){
    | localStorage.setItem('count', '0')
}

window.onload = clearCount;
```

2)

```
<!DOCTYPE html>
<html lang="en">

<head>
  <script src="data.js"></script>
  <script src="view.js"></script>
  <script src="controller.js"></script>

  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Shopping List MVC</title>
</head>

<body>
  <h1>Shopping List</h1>

  <div>
    <input type="text" id="itemInput" placeholder="Enter item">
    <button>Add Item</button>
  </div>

  <ul id="shoppinglist"></ul>

  <script src="view.js">
    const model = new ShoppingListModel();
    const view = new ShoppingListView();
    const controller = new ShoppingListController(model, view);

    model.loadItemsFromStorage();
    controller.updateView();
  </script>
</body>

</html>
```

```

class ShoppingListView {
  constructor() {
    this.itemInput = document.getElementById('itemInput');
    this.shoppingList = document.getElementById('shoppingList');
    this.addItemButton = document.querySelector('button');

    this.addItemButton.addEventListener('click', () => this.controller.addItem());
  }

  bindRemoveItem(handler) {
    this.shoppingList.addEventListener('click', (event) => {
      if (event.target.tagName === 'LI') {
        const index = Array.from(this.shoppingList.children).indexOf(event.target);
        handler(index);
      }
    });
  }

  updateItemList(items) {
    this.shoppingList.innerHTML = '';
    items.forEach(item => {
      const listItem = document.createElement("li");
      listItem.textContent = item;
      this.shoppingList.appendChild(listItem);
    });
  }

  getItemInputValue() {
    return this.itemInput.value.trim();
  }

  clearItemInput() {
    this.itemInput.value = '';
  }
}

```