

1.1\*\*ไม่มีการแก้ไขอะไรในหน้า index.html

script.js ใช้เพิ่ม element node ที่ไม่มีใน index.html + เพิ่ม Event ให้แสดงผลที่ต้องการ

item.js เป็น object ที่เก็บข้อมูลของสินค้า และมี function ที่จัดการ สร้าง ลบ แก้ไข ตัว localStorage ให้

// source code ทั้งหมด จากไฟล์ script.js

```
index.html JS script.js X JS items.js
63130500106 > final_materials > JS script.js > showProduct
1  import { product } from "./items.js";
2
3  //1.2.1 Add event from input
4  const inputText = document.getElementById('item');
5  const inputUnit = document.getElementById('qty');
6  const buttonSubmit = document.getElementById('btn-submit');
7
8  buttonSubmit.addEventListener('click',()=>{
9      let productName = inputText.value;
10     let productUnit = inputUnit.value;
11     if(productName.length !== 0){
12         product.add(productName,productUnit);
13         product.save();
14         location.reload();
15         inputText.value="";
16     }
17 });
18
19 //1.2.1 show Count
20 const countPendingItems = document.getElementById('countPendingItems');
21 countPendingItems.textContent = `${product.checkCountPending()}`
22 const countDoneItems = document.getElementById('countDoneItems');
23 countDoneItems.textContent = `${product.checkCountDone()}`
24
```

```
25
26 //1.2.1 add show product-list
27 const orderList = document.getElementById('buying-list');
28
29 let productList = product.list();
30 productList.forEach((item)=>{
31     showProduct(item);
32 })
33 function showProduct(item){
34     let{itemName,itemUnit,itemStatus} = item;
35     const listItem = document.createElement('li');
36
37     const imgStatus = document.createElement('img');
38     if(itemStatus == 'Pending'){
39         imgStatus.src = "./images/pending.svg";
40     }else if(itemStatus == 'Done'){
41         imgStatus.src = "./images/done.svg";
42         listItem.style = "text-decoration: line-through; color:grey;";
43     }
44     //1.2.3 Add event icon
45     imgStatus.addEventListener('click',()=>{
46         product.buy(itemName);
47         product.save();
48         location.reload();
49     });
50
51     listItem.textContent = `${itemName}, ${itemUnit} unit,`
52     listItem.appendChild(imgStatus);
53     orderList.appendChild(listItem);
54 }
55
56 //1.2.2 add Clear
57 const buttonClear = document.getElementById('btn-clear');
58 buttonClear.addEventListener('click',()=>{
59     product.removeAll();
60     location.reload();
61 })
```

// source code ทั้งหมด จากไฟล์ item.js

```

index.html JS script.js JS items.js X
63130500106 > final_materials > JS items.js > [?] product > save
1 //1.1
2 export let product = {
3   items:[],
4   //1.2.1
5   add: function(name,unit){
6     let foundIndex = -1;
7     foundIndex = product.items.findIndex(
8       (item) => item.itemName === name
9     );
10    if (foundIndex === -1) {
11      product.items.push({
12        itemName: name,
13        itemUnit: Number(unit),
14        itemStatus: 'Pending'
15      });
16    } else {
17      product.items[foundIndex].itemUnit = unit;
18    }
19    //1.2.4
20  }, save: function () {
21    localStorage.setItem('ItemsList', JSON.stringify(product.items));
22  },
23   //1.2.1
24   load: function(){
25     product.items = localStorage.getItem('ItemsList');
26     if (product.items === undefined || product.items === 0) {
27       product.items = [];
28     } else {
29       product.items = JSON.parse(product.items);
30     }
31  },
32   //1.2.1
33   list: function () {
34     this.load();
35     let listProduct = [];
36     product.items.forEach((item) => {
37       listProduct.push({ ...item });
38     });
39     return listProduct;
40  },
41   //1.2.1
42   checkCountPending : function(){
43     this.load();
44     let itemsPending = product.items.filter((item)=>{return item.itemStatus == 'Pending'});
45     return itemsPending.length;
46  },
47   //1.2.1
48   checkCountDone : function(){
49     this.load();
50     let itemsPending = product.items.filter((item)=>{return item.itemStatus == 'Done'});
51     return itemsPending.length;
52  },
53
54   //1.2.2
55   removeAll: function(){
56     if (confirm('Emptry List')) {
57       product.items = [];
58       localStorage.removeItem('ItemsList');
59     }
60  },
61
62   //1.2.3
63   buy: function(itemId){
64     this.load()
65     let indexToFind = product.items.findIndex((item)=>{
66       return item.itemName.includes(itemId);
67     });
68     product.items[indexToFind].itemStatus = 'Done';
69  },
70
71
72 }

```

1.2

1.2.1

ก่อนกด submit

## Buying Items

Add items you want to buy.

unit (s)

## Your Buying List

(Pending: 0 items, Done: 0 items)

หลังกด submit

## Buying Items

Add items you want to buy.

unit (s)

## Your Buying List

(Pending: 1 items, Done: 0 items)

1. book, 3 unit, 🛒

โค้ดส่วนที่ใช้แสดงผล จากไฟล์ script.js

```

index.html JS script.js JS items.js
63130500106 > final_materials > JS script.js > buttonSubmit.addEventListener('click') callback
1  import { product } from "../items.js";
2
3  //1.2.1 Add event from input
4  const inputText = document.getElementById('item');
5  const inputUnit = document.getElementById('qty');
6  const buttonSubmit = document.getElementById('btn-submit');
7
8  buttonSubmit.addEventListener('click',()=>{
9      let productName = inputText.value;
10     let productUnit = inputUnit.value;
11     if(productName.length !== 0){
12         product.add(productName,productUnit);
13         product.save();
14         location.reload();
15         inputText.value="";
16     }
17 });
18
19 //1.2.1 show Count
20 const countPendingItems = document.getElementById('countPendingItems');
21 countPendingItems.textContent = `${product.checkCountPending()}`
22 const countDoneItems = document.getElementById('countDoneItems');
23 countDoneItems.textContent = `${product.checkCountDone()}`
24
25

```

```

index.html JS script.js X JS items.js
63130500106 > final_materials > JS script.js > showProduct

25
26 //1.2.1 add show product-list
27 const orderList = document.getElementById('buying-list');
28
29 let productList = product.list();
30 productList.forEach((item)=>{
31     showProduct(item);
32 })
33 function showProduct(item){
34     let{itemName,itemUnit,itemStatus} = item;
35     const listItem = document.createElement('li');
36
37     const imgStatus = document.createElement('img');
38     if(itemStatus == 'Pending'){
39         imgStatus.src = './images/pending.svg';
40     }else if(itemStatus == 'Done'){
41         imgStatus.src = './images/done.svg';
42         listItem.style = "text-decoration: line-through; color:grey;";
43     }
44     //1.2.3 Add event icon
45     imgStatus.addEventListener('click',()=>{
46         product.buy(itemName);
47         product.save();
48         location.reload();
49     });
50
51     listItem.textContent = `${itemName}, ${itemUnit} unit,`
52     listItem.appendChild(imgStatus);
53     orderList.appendChild(listItem);
54 }
55

```

Function ที่ใช้ จากไฟล์ item.js

```

index.html X JS script.js JS items.js X
63130500106 > final_materials > JS items.js > product > removeAll

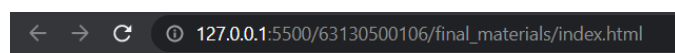
1 //1.1
2 export let product = {
3     items:[],
4     //1.2.1
5     add: function(name,unit){
6         let foundIndex = -1;
7         foundIndex = product.items.findIndex(
8             (item) => item.itemName === name
9         );
10        if (foundIndex === -1) {
11            product.items.push({
12                itemName: name,
13                itemUnit: Number(unit),
14                itemStatus: 'Pending'
15            });
16        } else {
17            product.items[foundIndex].itemUnit = unit;
18        }
19    }
20 }

```

```
23 //1.2.1
24 load: function(){
25     product.items = localStorage.getItem('ItemsList');
26     if (product.items == undefined || product.items === 0) {
27         product.items = [];
28     } else {
29         product.items = JSON.parse(product.items);
30     }
31 },
32 //1.2.1
33 list: function () {
34     this.load();
35     let listProduct = [];
36     product.items.forEach((item) => {
37         listProduct.push({ ...item });
38     });
39     return listProduct;
40 },
41 //1.2.1
42 checkCountPending : function(){
43     this.load();
44     let itemsPending = product.items.filter((item)=>{return item.itemStatus == 'Pending'});
45     return itemsPending.length;
46 },
47 //1.2.1
48 checkCountDone : function(){
49     this.load();
50     let itemsPending = product.items.filter((item)=>{return item.itemStatus == 'Done'});
51     return itemsPending.length;
52 },
53
```

## 1.2.2

ก่อนกด clear items

**Buying Items**

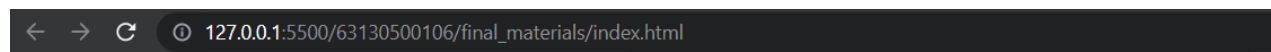
Add items you want to buy.

  unit (s)  
**Your Buying List**

(Pending: 3 items, Done:0 items)

1. book, 3 unit, 📖
2. drink, 1 unit, 🥤
3. milk, 2 unit, 🥛

หลังกด clear items

**Buying Items**

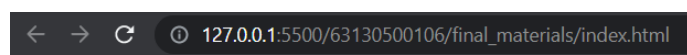
Add items you want to buy.

  unit (s)  
**Your Buying List**

(Pending: 3 items, Done:0 items)

1. book, 3 unit, 📖
2. drink, 1 unit, 🥤
3. milk, 2 unit, 🥛

หลังกด ok ลีนค้หายหมด

**Buying Items**

Add items you want to buy.

  unit (s)  
**Your Buying List**

(Pending: 0 items, Done:0 items)

127.0.0.1:5500 says

Empry List

โค้ดส่วนที่ใช้แสดงผล จากไฟล์ script.js

```
index.html X JS script.js X JS items.js
63130500106 > final_materials > JS script.js > ...
55
56 //1.2.2 add Clear
57 const buttonClear = document.getElementById('btn-clear');
58 buttonClear.addEventListener('click', ()=>{
59     product.removeAll();
60     location.reload();
61 })
62
```

Function ที่ใช้ จากไฟล์ item.js

```
index.html X JS script.js JS items.js X
63130500106 > final_materials > JS items.js > ...
54 //1.2.2
55 removeAll: function(){
56     if (confirm('Emptry List')) {
57         product.items = [];
58         localStorage.removeItem('ItemsList');
59     }
60 },
61
```



## 1.2.3

ก่อนกด icon หลังสินค้า

127.0.0.1:5500/63130500106/final\_materials/index.html

## Buying Items

Add items you want to buy.

e.g. milk

1

unit (s)

submit

### Your Buying List

(Pending: 3 items, Done:0 items)

1. book, 3 unit,

2. drink, 1 unit,

3. milk, 2 unit,

clear items

หลังกด icon หลังสินค้า

127.0.0.1:5500/63130500106/final\_materials/index.html

## Buying Items

Add items you want to buy.

e.g. milk

1

unit (s)

submit

### Your Buying List

(Pending: 2 items, Done:1 items)

1. book, 3 unit,

2. drink, 1 unit,

3. milk, 2 unit,

clear items

127.0.0.1:5500/63130500106/final\_materials/index.html

## Buying Items

Add items you want to buy.

e.g. milk

1

unit (s)

submit

### Your Buying List

(Pending: 0 items, Done:3 items)

1. book, 3 unit,

2. drink, 1 unit,

3. milk, 2 unit,

clear items

โค้ดส่วนที่ใช้แสดงผล จากไฟล์ script.js

```

index.html JS script.js X JS items.js
63130500106 > final_materials > JS script.js > showProduct

25
26 //1.2.1 add show product-list
27 const orderList = document.getElementById('buying-list');
28
29 let productList = product.list();
30 productList.forEach((item)=>{
31     showProduct(item);
32 })
33 function showProduct(item){
34     let{itemName,itemUnit,itemStatus} = item;
35     const listItem = document.createElement('li');
36
37     const imgStatus = document.createElement('img');
38     if(itemStatus == 'Pending'){
39         imgStatus.src = './images/pending.svg';
40     }else if(itemStatus == 'Done'){
41         imgStatus.src = './images/done.svg';
42         listItem.style = "text-decoration: line-through; color:grey;";
43     }
44     //1.2.3 Add event icon
45     imgStatus.addEventListener('click',()=>{
46         product.buy(itemName);
47         product.save();
48         location.reload();
49     });
50
51     listItem.textContent = `${itemName}, ${itemUnit} unit,`
52     listItem.appendChild(imgStatus);
53     orderList.appendChild(listItem);
54 }
55

```

Function ที่ใช้ จากไฟล์ item.js

```

index.html JS script.js JS items.js X
63130500106 > final_materials > JS items.js > ...

61
62 //1.2.3
63 buy: function(itemId){
64     this.load()
65     let indexToFind = product.items.findIndex((item)=>{
66         return item.itemName.includes(itemId);
67     });
68     product.items[indexToFind].itemStatus = 'Done';
69 },
70
71
72 }

```

## 1.2.4

The screenshot displays a web browser window with the URL `127.0.0.1:5500/63130500106/final_materials/index.html`. The application, titled "Buying Items", allows users to add items to a list. The input fields show "e.g. milk" and "1" unit, with a "submit" button. Below, the "Your Buying List" section shows a list of items: "1. book, 3 unit", "2. drink, 1 unit", "3. milk, 2 unit", and "4. coffee, 10 unit". A "clear items" button is at the bottom.

The browser's developer tools are open, showing the "Application" tab. The "ItemsList" key in local storage contains the following JSON array:

```
[{"itemName": "book", "itemUnit": 3, "itemStatus": "Done"}, {"itemName": "drink", "itemUnit": 1, "itemStatus": "Done"}, {"itemName": "milk", "itemUnit": 2, "itemStatus": "Done"}, {"itemName": "coffee", "itemUnit": 10, "itemStatus": "Pending"}]
```

The "Sources" tab shows the file `items.js` with the following code:

```
18 }
19 //1.2.4
20 },save: function () {
21     localStorage.setItem('ItemsList', JSON.stringify(product.items));
22 },
```