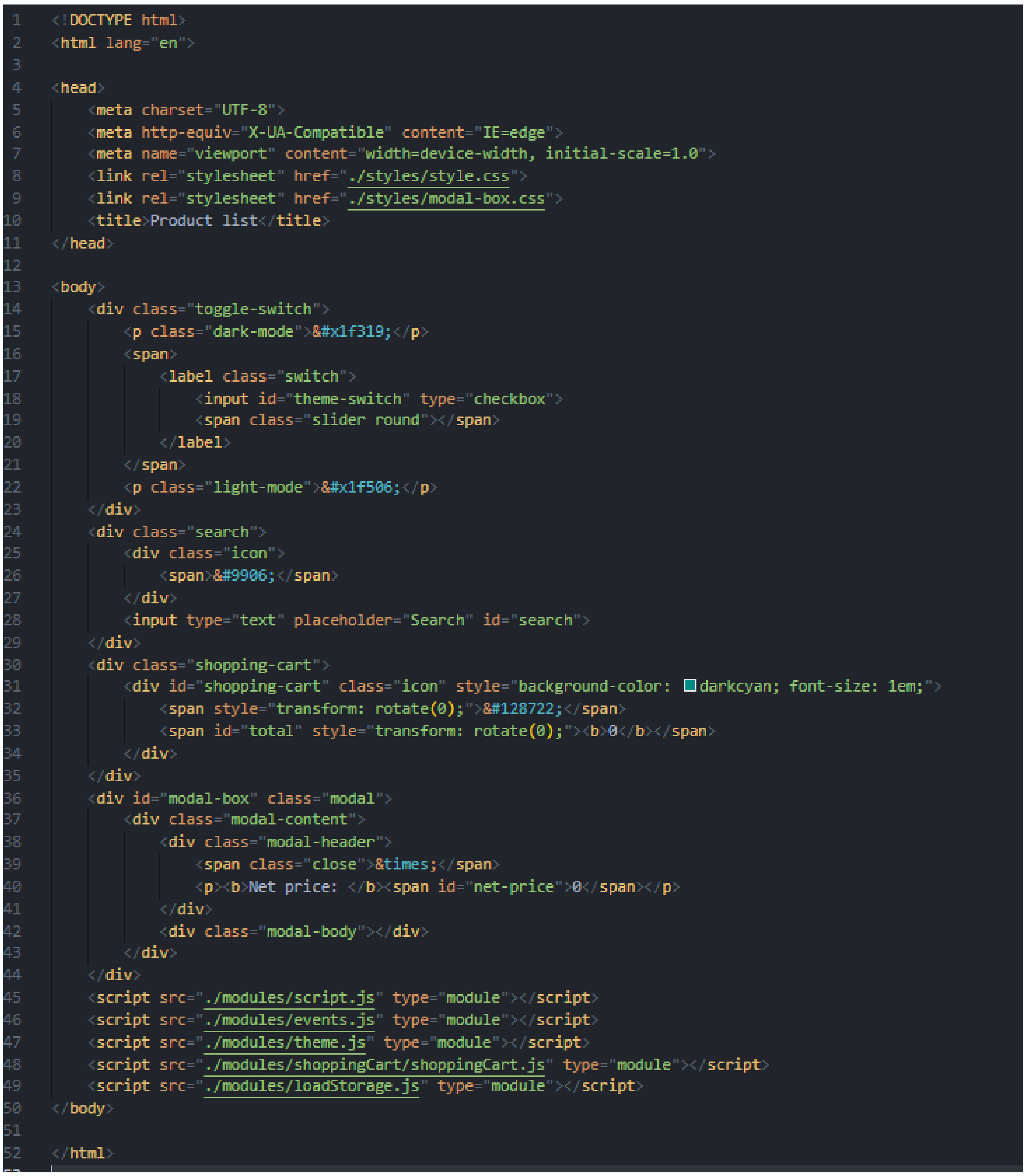
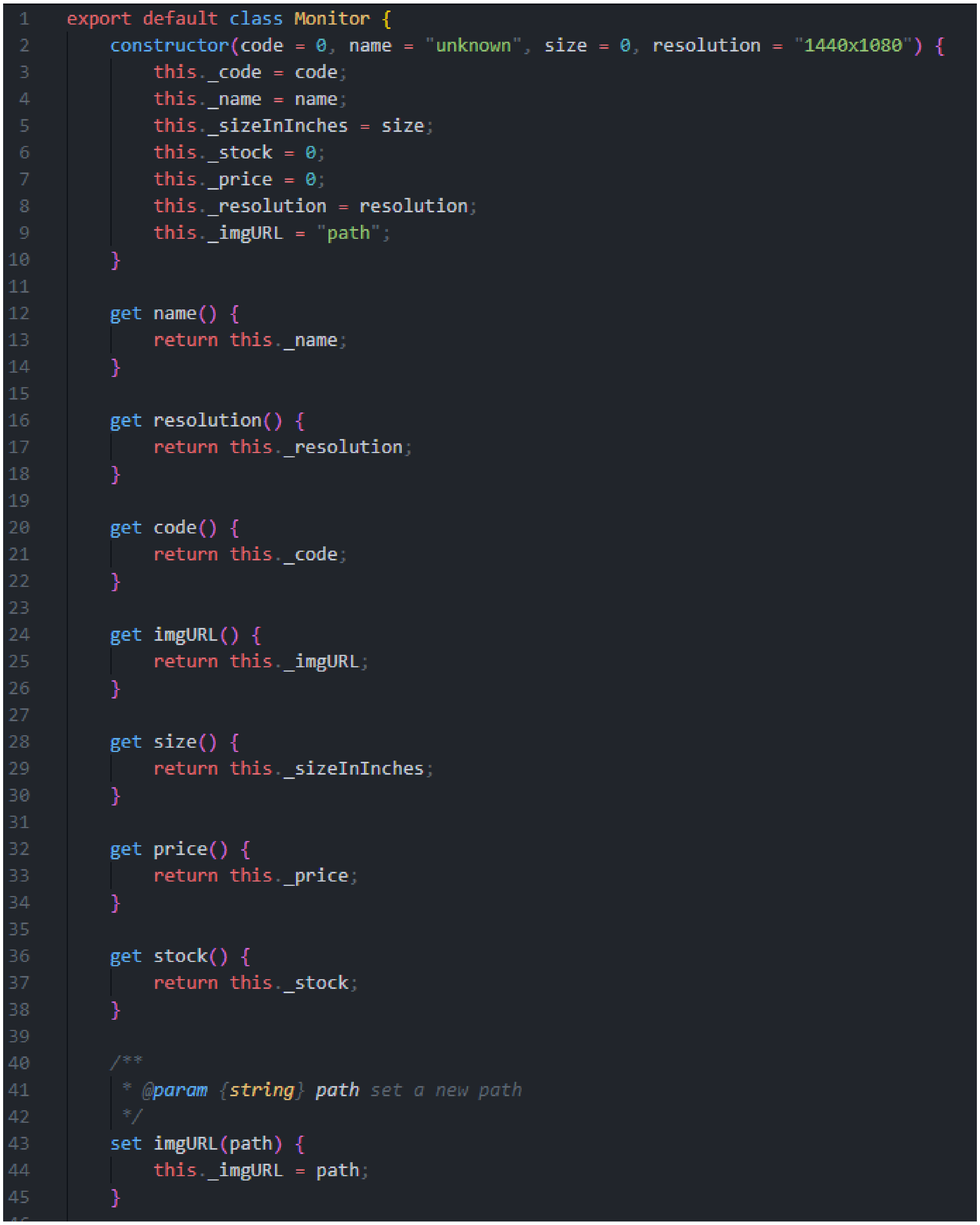
index:

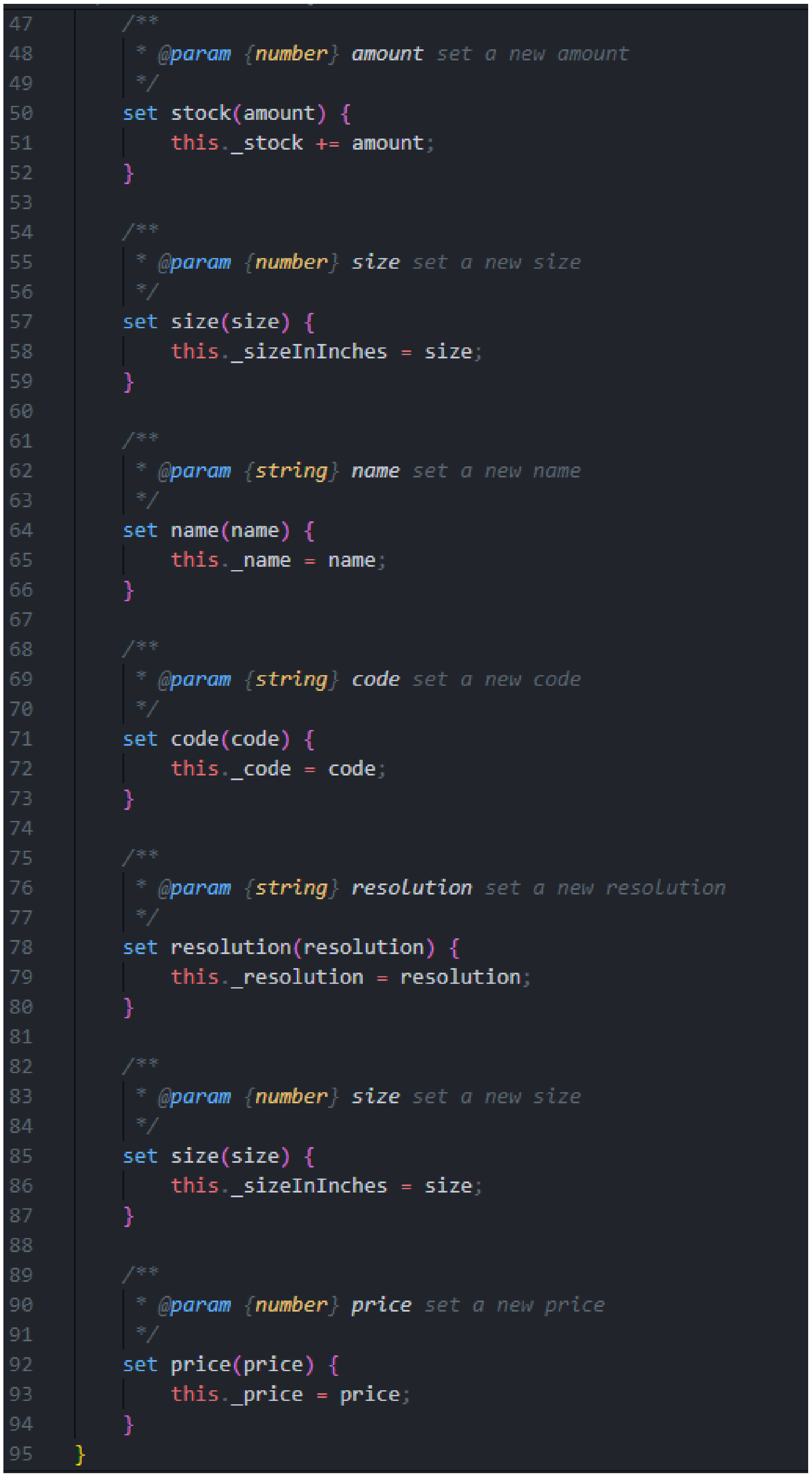


```html
<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link rel="stylesheet" href="./styles/style.css">
    <link rel="stylesheet" href="./styles/modal-box.css">
    <title>Product list</title>
</head>

<body>
    <div class="toggle-switch">
        <p class="dark-mode">&#x1f319;</p>
        <span>
            <label class="switch">
                <input id="theme-switch" type="checkbox">
                <span class="slider round"></span>
            </label>
        </span>
        <p class="light-mode">&#x1f506;</p>
    </div>
    <div class="search">
        <div class="icon">
            <span>&#9906;</span>
        </div>
        <input type="text" placeholder="Search" id="search">
    </div>
    <div class="shopping-cart">
        <div id="shopping-cart" class="icon" style="background-color: darkcyan; font-size: 1em;">
            <span style="transform: rotate(0);">&#128722;</span>
            <span id="total" style="transform: rotate(0);"><b>0</b></span>
        </div>
    </div>
    <div id="modal-box" class="modal">
        <div class="modal-content">
            <div class="modal-header">
                <span class="close">&times;</span>
                <p><b>Net price: </b><span id="net-price">0</span></p>
            </div>
            <div class="modal-body"></div>
        </div>
    </div>
    <script src="./modules/script.js" type="module"></script>
    <script src="./modules/events.js" type="module"></script>
    <script src="./modules/theme.js" type="module"></script>
    <script src="./modules/shoppingCart/shoppingCart.js" type="module"></script>
    <script src="./modules/loadStorage.js" type="module"></script>
</body>

</html>
```

monitor.js:

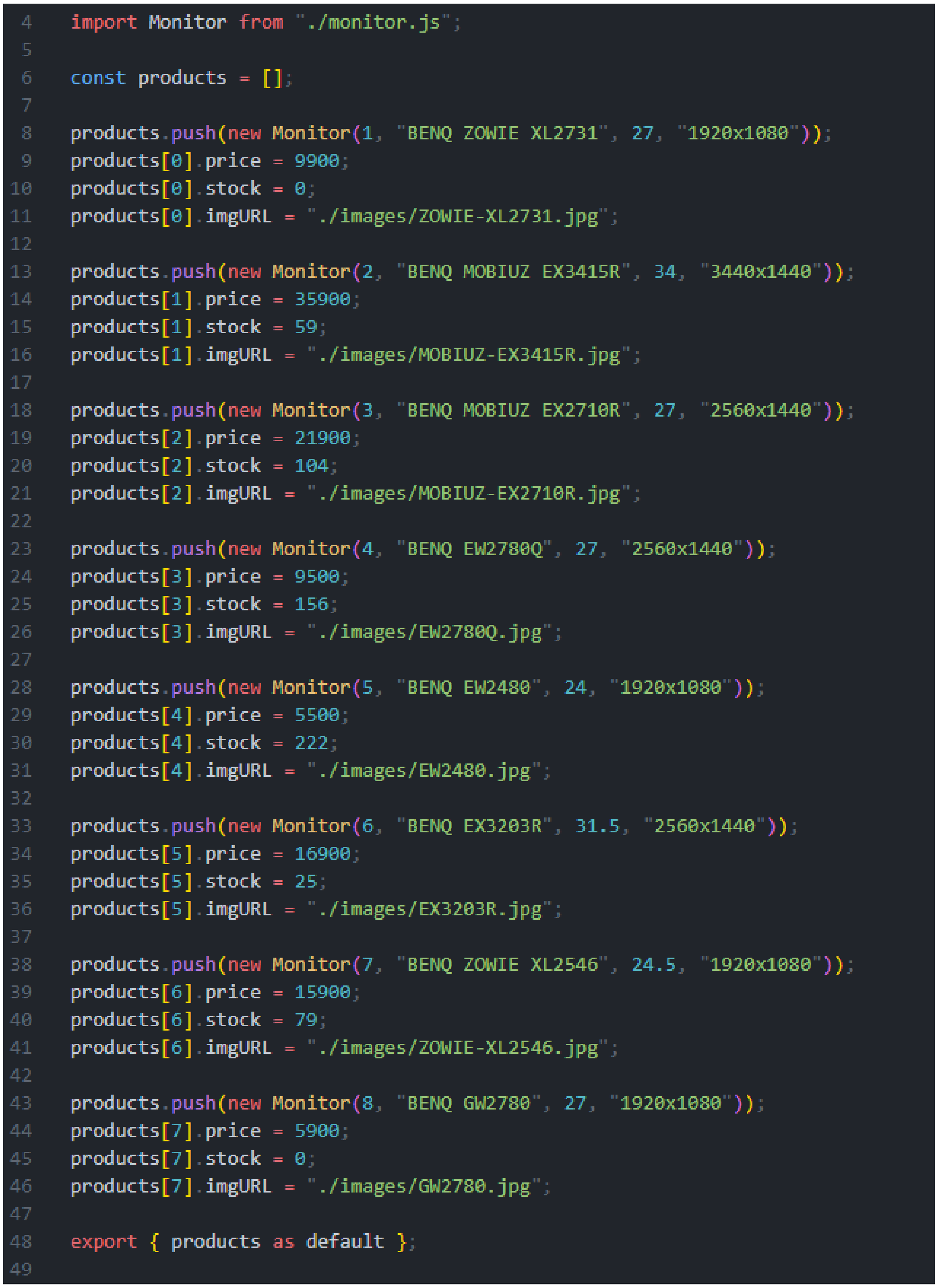


```javascript
export default class Monitor {
    constructor(code = 0, name = "unknown", size = 0, resolution = "1440x1080") {
        this._code = code;
        this._name = name;
        this._sizeInInches = size;
        this._stock = 0;
        this._price = 0;
        this._resolution = resolution;
        this._imgURL = "path";
    }

    get name() {
        return this._name;
    }

    get resolution() {
        return this._resolution;
    }

    get code() {
        return this._code;
    }

    get imgURL() {
        return this._imgURL;
    }

    get size() {
        return this._sizeInInches;
    }

    get price() {
        return this._price;
    }

    get stock() {
        return this._stock;
    }

    /**
     * @param {string} path set a new path
     */
    set imgURL(path) {
        this._imgURL = path;
    }
```



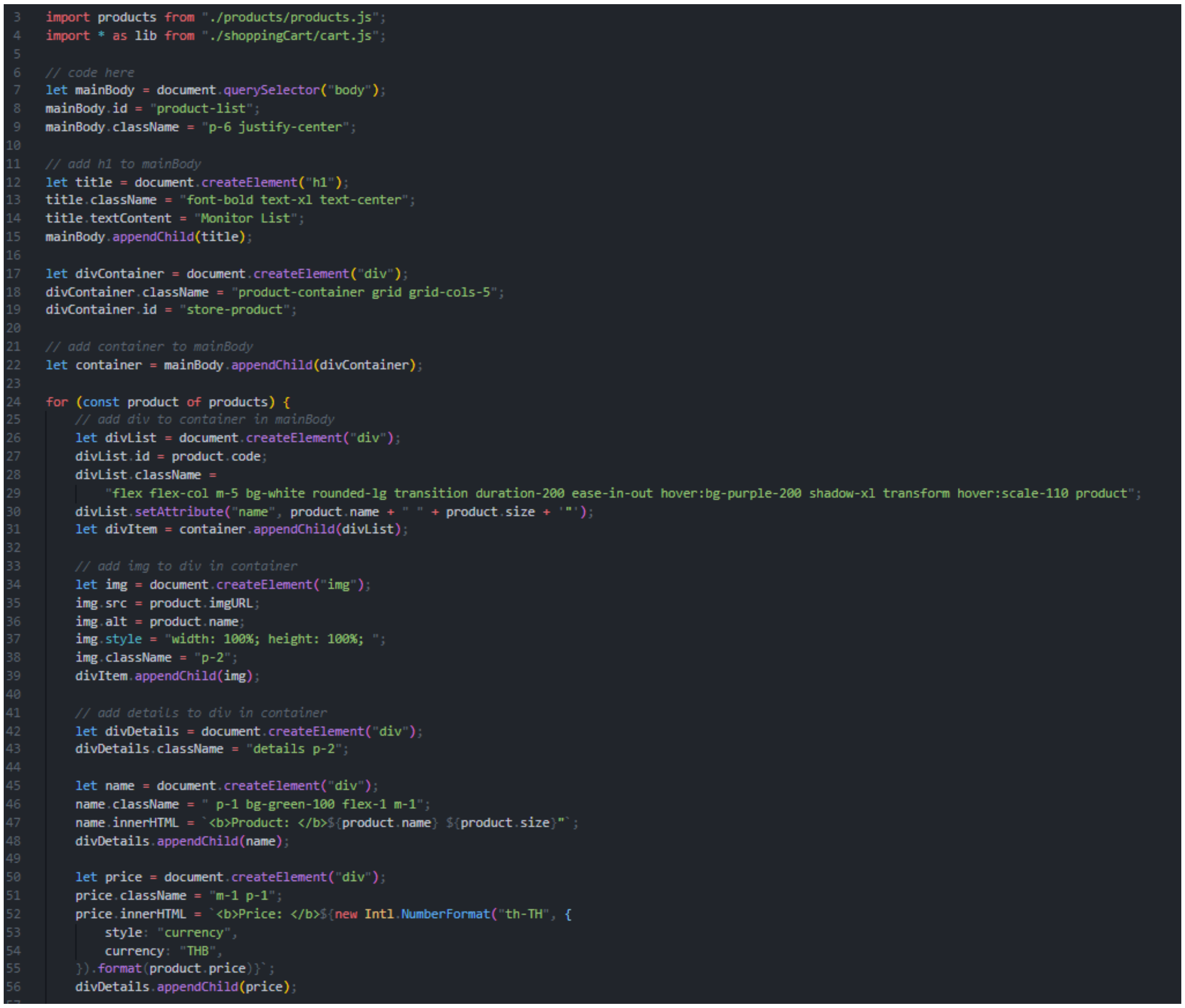
```
    /**
     * @param {number} amount set a new amount
     */
    set stock(amount) {
        this._stock += amount;
    }

    /**
     * @param {number} size set a new size
     */
    set size(size) {
        this._sizeInInches = size;
    }

    /**
     * @param {string} name set a new name
     */
    set name(name) {
        this._name = name;
    }

    /**
     * @param {string} code set a new code
     */
    set code(code) {
        this._code = code;
    }

    /**
     * @param {string} resolution set a new resolution
     */
    set resolution(resolution) {
        this._resolution = resolution;
    }

    /**
     * @param {number} size set a new size
     */
    set size(size) {
        this._sizeInInches = size;
    }

    /**
     * @param {number} price set a new price
     */
    set price(price) {
        this._price = price;
    }
}
```

products.js:

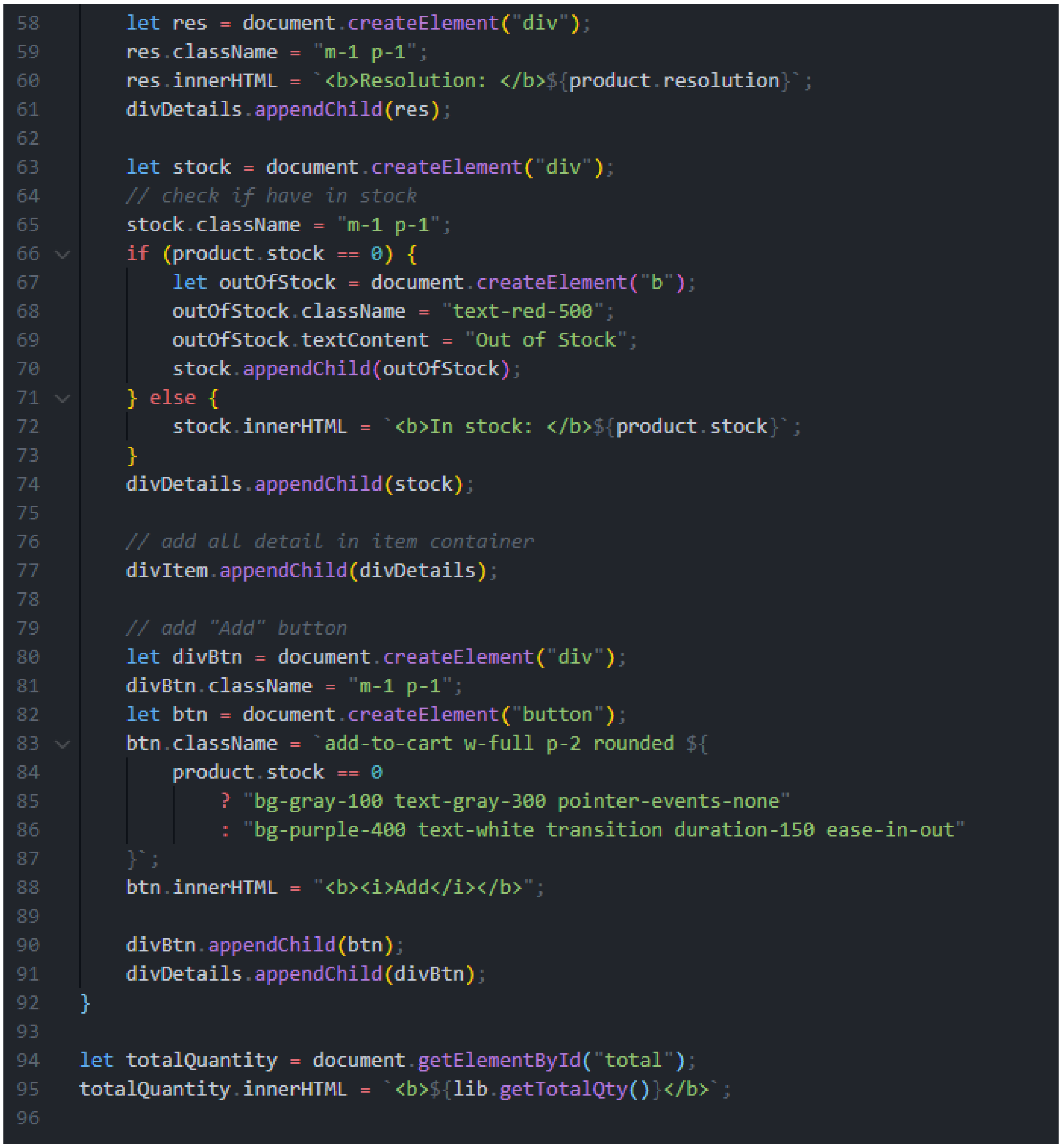


```
import Monitor from "./monitor.js";

const products = [];

products.push(new Monitor(1, "BENQ ZOWIE XL2731", 27, "1920x1080"));
products[0].price = 9900;
products[0].stock = 0;
products[0].imgURL = "./images/ZOWIE-XL2731.jpg";

products.push(new Monitor(2, "BENQ MOBIUZ EX3415R", 34, "3440x1440"));
products[1].price = 35900;
products[1].stock = 59;
products[1].imgURL = "./images/MOBIUZ-EX3415R.jpg";

products.push(new Monitor(3, "BENQ MOBIUZ EX2710R", 27, "2560x1440"));
products[2].price = 21900;
products[2].stock = 104;
products[2].imgURL = "./images/MOBIUZ-EX2710R.jpg";

products.push(new Monitor(4, "BENQ EW2780Q", 27, "2560x1440"));
products[3].price = 9500;
products[3].stock = 156;
products[3].imgURL = "./images/EW2780Q.jpg";

products.push(new Monitor(5, "BENQ EW2480", 24, "1920x1080"));
products[4].price = 5500;
products[4].stock = 222;
products[4].imgURL = "./images/EW2480.jpg";

products.push(new Monitor(6, "BENQ EX3203R", 31.5, "2560x1440"));
products[5].price = 16900;
products[5].stock = 25;
products[5].imgURL = "./images/EX3203R.jpg";

products.push(new Monitor(7, "BENQ ZOWIE XL2546", 24.5, "1920x1080"));
products[6].price = 15900;
products[6].stock = 79;
products[6].imgURL = "./images/ZOWIE-XL2546.jpg";

products.push(new Monitor(8, "BENQ GW2780", 27, "1920x1080"));
products[7].price = 5900;
products[7].stock = 0;
products[7].imgURL = "./images/GW2780.jpg";

export { products as default };

```

script.js:



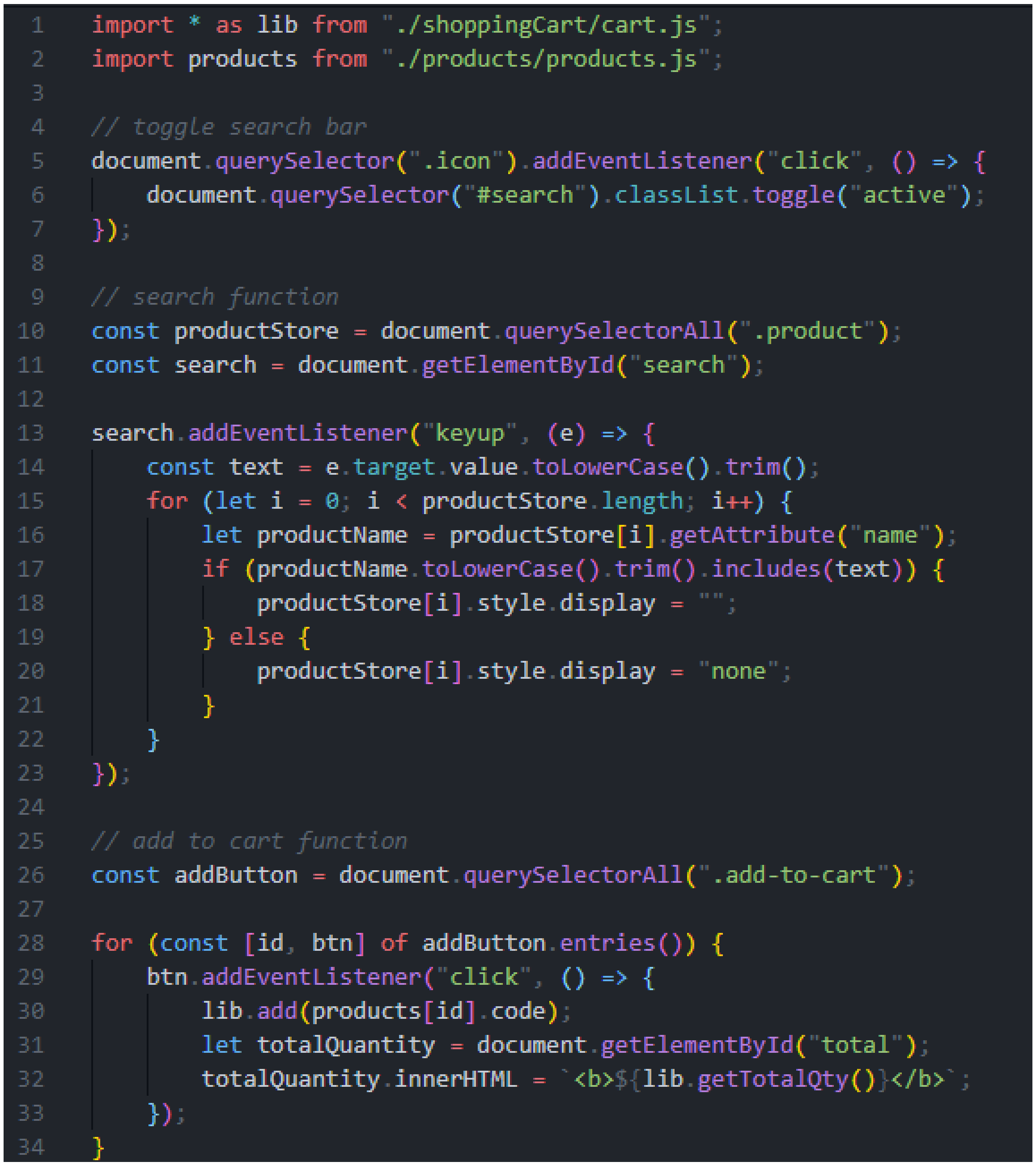
```javascript
import products from "./products/products.js";
import * as lib from "./shoppingCart/cart.js";

// code here
let mainBody = document.querySelector("body");
mainBody.id = "product-list";
mainBody.className = "p-6 justify-center";

// add h1 to mainBody
let title = document.createElement("h1");
title.className = "font-bold text-xl text-center";
title.textContent = "Monitor List";
mainBody.appendChild(title);

let divContainer = document.createElement("div");
divContainer.className = "product-container grid grid-cols-5";
divContainer.id = "store-product";

// add container to mainBody
let container = mainBody.appendChild(divContainer);

for (const product of products) {
    // add div to container in mainBody
    let divList = document.createElement("div");
    divList.id = product.code;
    divList.className =
        "flex flex-col m-5 bg-white rounded-lg transition duration-200 ease-in-out hover:bg-purple-200 shadow-xl transform hover:scale-110 product";
    divList.setAttribute("name", product.name + " " + product.size + '"');
    let divItem = container.appendChild(divList);

    // add img to div in container
    let img = document.createElement("img");
    img.src = product.imgURL;
    img.alt = product.name;
    img.style = "width: 100%; height: 100%; ";
    img.className = "p-2";
    divItem.appendChild(img);

    // add details to div in container
    let divDetails = document.createElement("div");
    divDetails.className = "details p-2";

    let name = document.createElement("div");
    name.className = " p-1 bg-green-100 flex-1 m-1";
    name.innerHTML = `<b>Product: </b>${product.name} ${product.size}"`;
    divDetails.appendChild(name);

    let price = document.createElement("div");
    price.className = "m-1 p-1";
    price.innerHTML = `<b>Price: </b>${new Intl.NumberFormat("th-TH", {
        style: "currency",
        currency: "THB",
    }).format(product.price)}`;
    divDetails.appendChild(price);
```



```
    let res = document.createElement("div");
    res.className = "m-1 p-1";
    res.innerHTML = `<b>Resolution: </b>${product.resolution}`;
    divDetails.appendChild(res);

    let stock = document.createElement("div");
    // check if have in stock
    stock.className = "m-1 p-1";
    if (product.stock == 0) {
        let outOfStock = document.createElement("b");
        outOfStock.className = "text-red-500";
        outOfStock.textContent = "Out of Stock";
        stock.appendChild(outOfStock);
    } else {
        stock.innerHTML = `<b>In stock: </b>${product.stock}`;
    }
    divDetails.appendChild(stock);

    // add all detail in item container
    divItem.appendChild(divDetails);

    // add "Add" button
    let divBtn = document.createElement("div");
    divBtn.className = "m-1 p-1";
    let btn = document.createElement("button");
    btn.className = `add-to-cart w-full p-2 rounded ${
        product.stock == 0
            ? "bg-gray-100 text-gray-300 pointer-events-none"
            : "bg-purple-400 text-white transition duration-150 ease-in-out"
    }`;
    btn.innerHTML = "<b><i>Add</i></b>";

    divBtn.appendChild(btn);
    divDetails.appendChild(divBtn);
}

let totalQuantity = document.getElementById("total");
totalQuantity.innerHTML = `<b>${lib.getTotalQty()}</b>`;
```

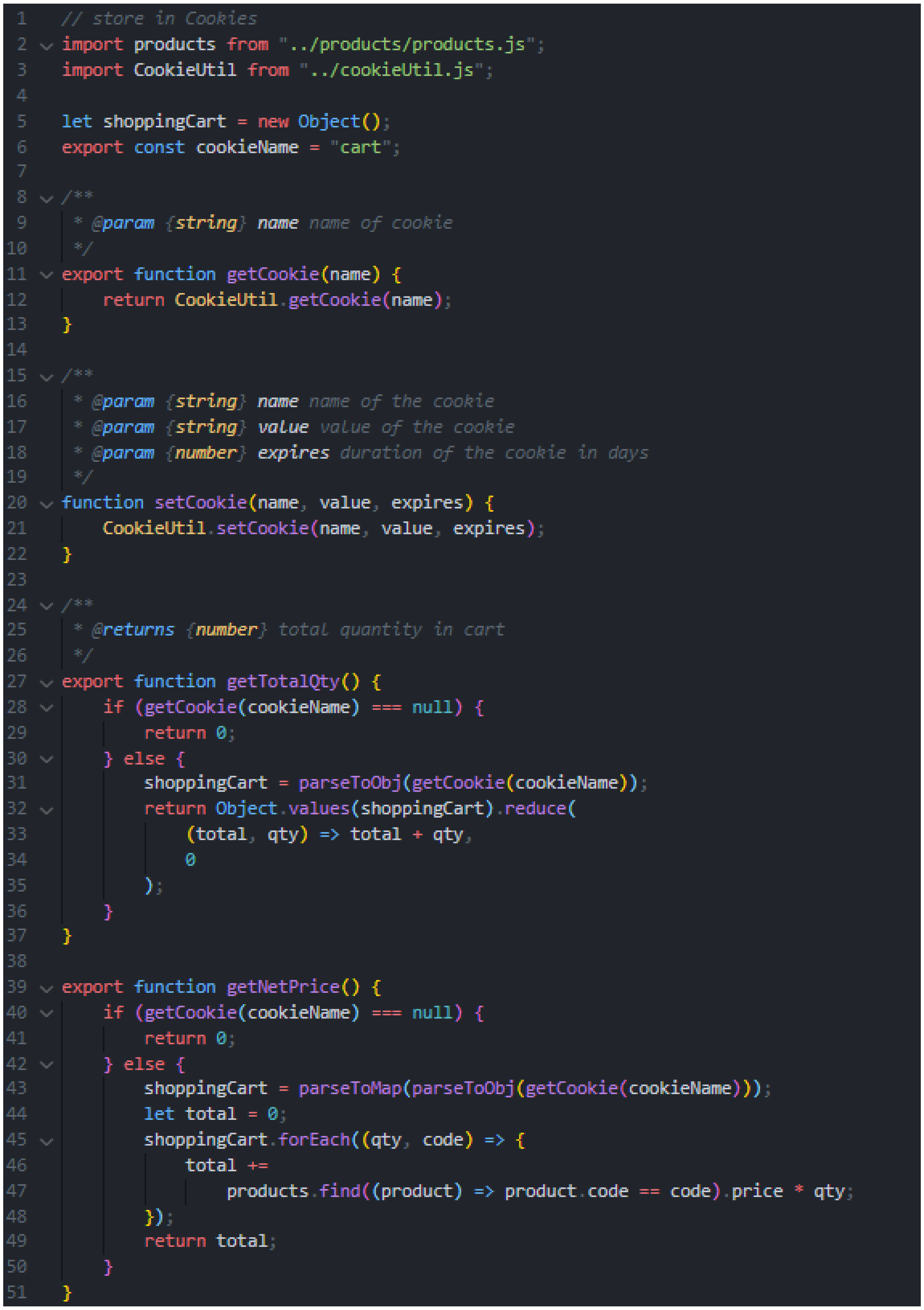
events.js:

change toggle search bar to arrow function

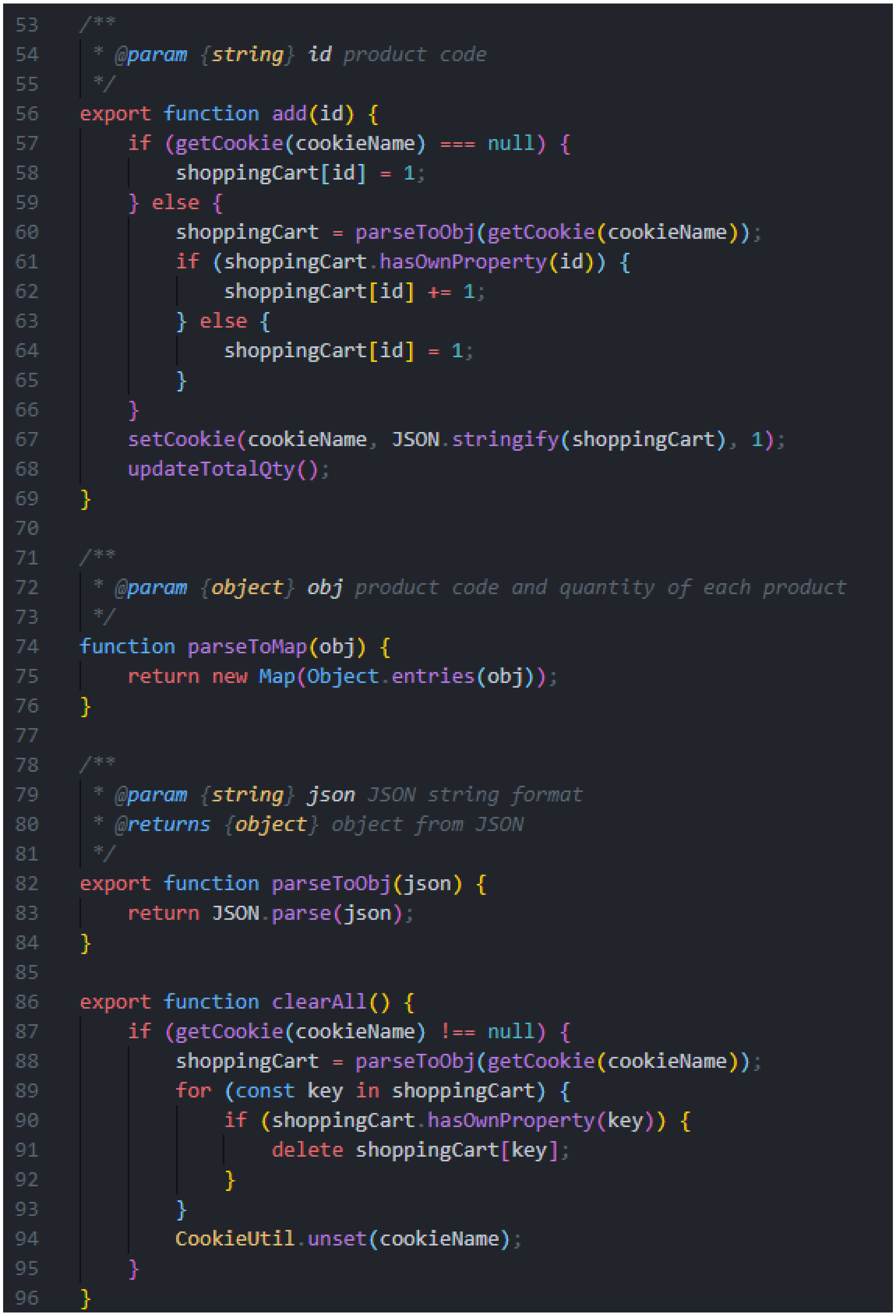


```javascript
import * as lib from "./shoppingCart/cart.js";
import products from "./products/products.js";

// toggle search bar
document.querySelector(".icon").addEventListener("click", () => {
    document.querySelector("#search").classList.toggle("active");
});

// search function
const productStore = document.querySelectorAll(".product");
const search = document.getElementById("search");

search.addEventListener("keyup", (e) => {
    const text = e.target.value.toLowerCase().trim();
    for (let i = 0; i < productStore.length; i++) {
        let productName = productStore[i].getAttribute("name");
        if (productName.toLowerCase().trim().includes(text)) {
            productStore[i].style.display = "";
        } else {
            productStore[i].style.display = "none";
        }
    }
});

// add to cart function
const addButton = document.querySelectorAll(".add-to-cart");

for (const [id, btn] of addButton.entries()) {
    btn.addEventListener("click", () => {
        lib.add(products[id].code);
        let totalQuantity = document.getElementById("total");
        totalQuantity.innerHTML = `<b>${lib.getTotalQty()}</b>`;
    });
}
```

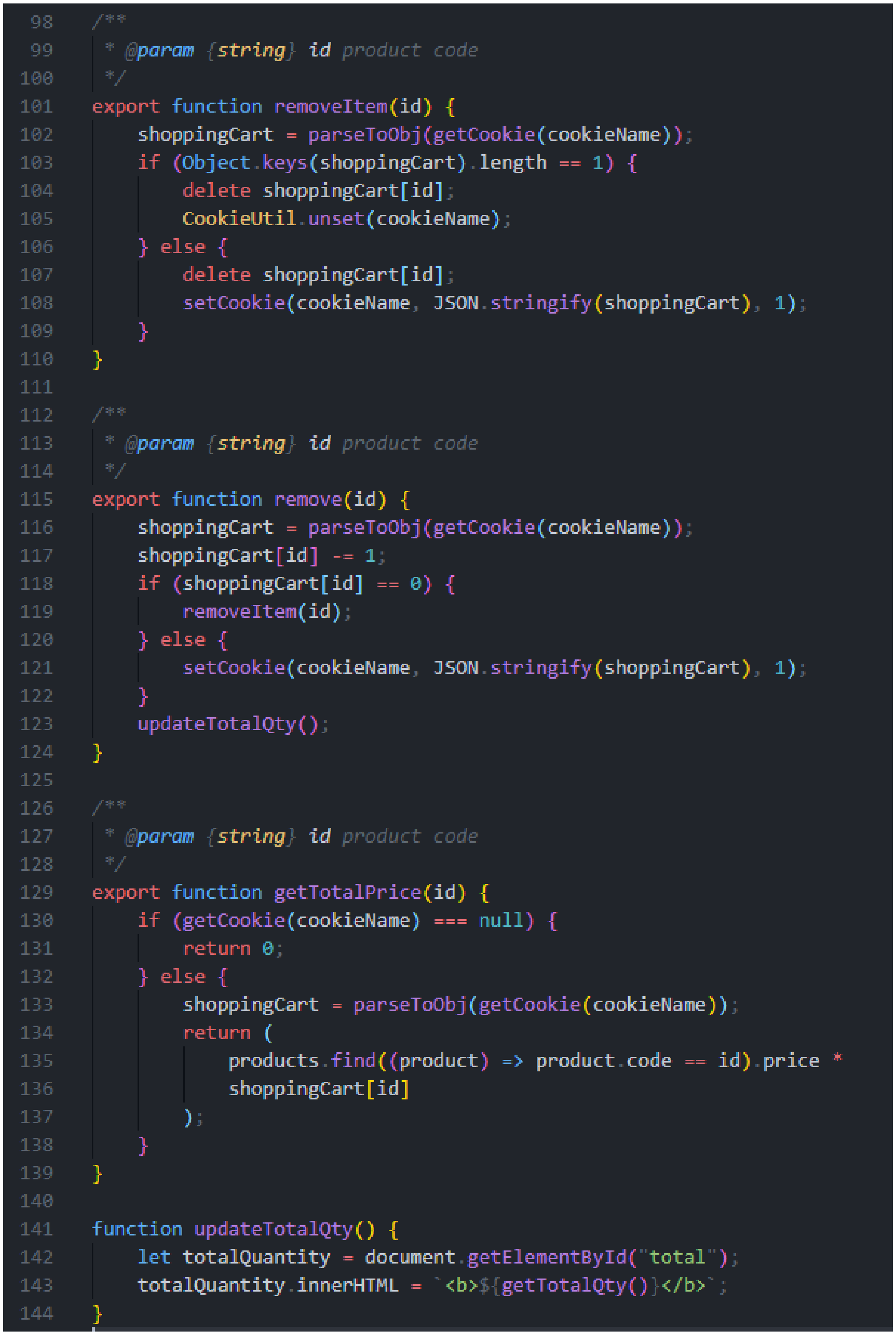
cart.js:



```javascript
// store in Cookies
import products from "../products/products.js";
import CookieUtil from "../cookieUtil.js";

let shoppingCart = new Object();
export const cookieName = "cart";

/**
 * @param {string} name name of cookie
 */
export function getCookie(name) {
    return CookieUtil.getCookie(name);
}

/**
 * @param {string} name name of the cookie
 * @param {string} value value of the cookie
 * @param {number} expires duration of the cookie in days
 */
function setCookie(name, value, expires) {
    CookieUtil.setCookie(name, value, expires);
}

/**
 * @returns {number} total quantity in cart
 */
export function getTotalQty() {
    if (getCookie(cookieName) === null) {
        return 0;
    } else {
        shoppingCart = parseToObj(getCookie(cookieName));
        return Object.values(shoppingCart).reduce(
            (total, qty) => total + qty,
            0
        );
    }
}

export function getNetPrice() {
    if (getCookie(cookieName) === null) {
        return 0;
    } else {
        shoppingCart = parseToMap(parseToObj(getCookie(cookieName)));
        let total = 0;
        shoppingCart.forEach((qty, code) => {
            total +=
                products.find((product) => product.code == code).price * qty;
        });
        return total;
    }
}
```

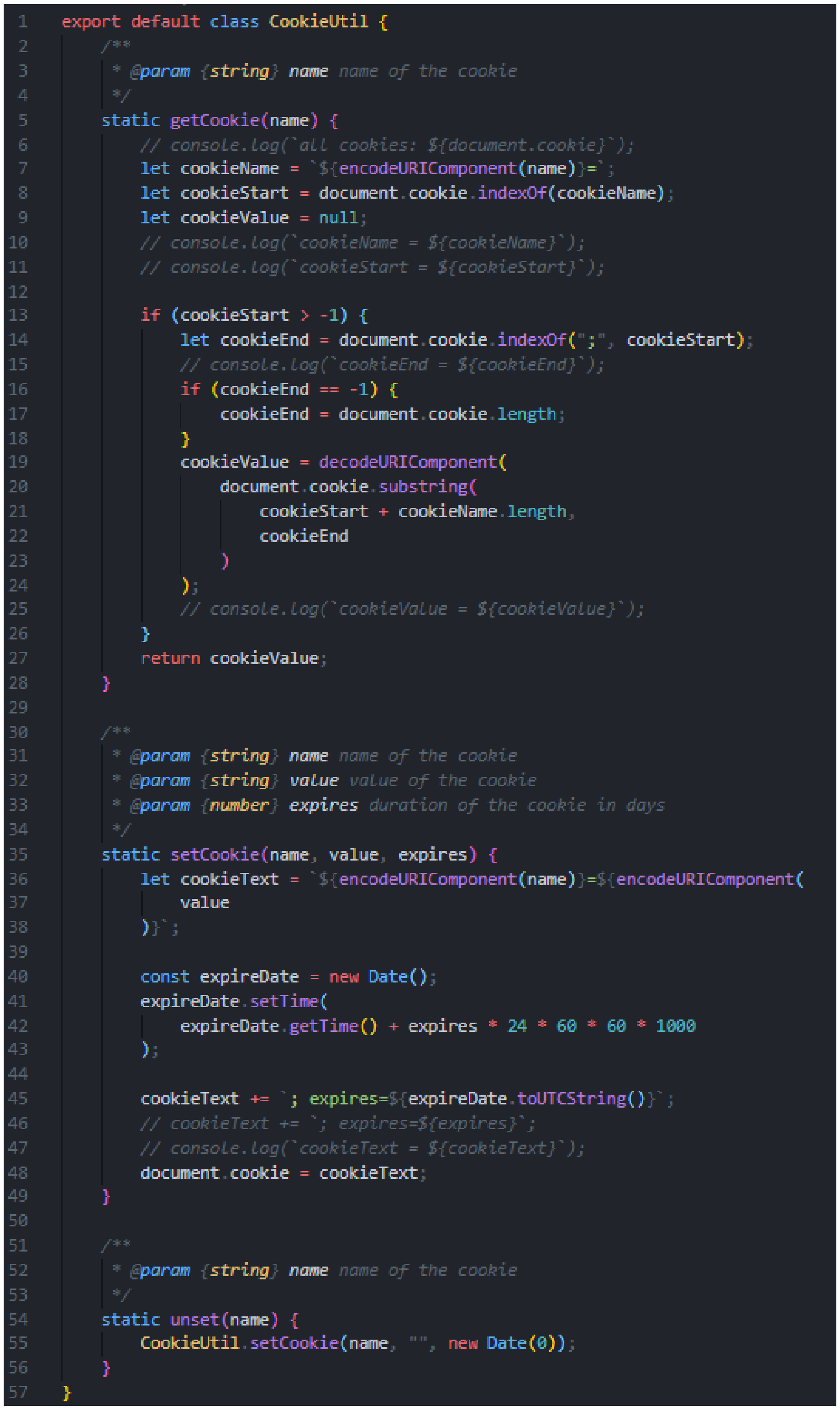


```javascript
/**
 * @param {string} id product code
 */
export function add(id) {
    if (getCookie(cookieName) === null) {
        shoppingCart[id] = 1;
    } else {
        shoppingCart = parseToObj(getCookie(cookieName));
        if (shoppingCart.hasOwnProperty(id)) {
            shoppingCart[id] += 1;
        } else {
            shoppingCart[id] = 1;
        }
    }
    setCookie(cookieName, JSON.stringify(shoppingCart), 1);
    updateTotalQty();
}

/**
 * @param {object} obj product code and quantity of each product
 */
function parseToMap(obj) {
    return new Map(Object.entries(obj));
}

/**
 * @param {string} json JSON string format
 * @returns {object} object from JSON
 */
export function parseToObj(json) {
    return JSON.parse(json);
}

export function clearAll() {
    if (getCookie(cookieName) !== null) {
        shoppingCart = parseToObj(getCookie(cookieName));
        for (const key in shoppingCart) {
            if (shoppingCart.hasOwnProperty(key)) {
                delete shoppingCart[key];
            }
        }
        CookieUtil.unset(cookieName);
    }
}
```



```
/**
 * @param {string} id product code
 */
export function removeItem(id) {
    shoppingCart = parseToObj(getCookie(cookieName));
    if (Object.keys(shoppingCart).length == 1) {
        delete shoppingCart[id];
        CookieUtil.unset(cookieName);
    } else {
        delete shoppingCart[id];
        setCookie(cookieName, JSON.stringify(shoppingCart), 1);
    }
}

/**
 * @param {string} id product code
 */
export function remove(id) {
    shoppingCart = parseToObj(getCookie(cookieName));
    shoppingCart[id] -= 1;
    if (shoppingCart[id] == 0) {
        removeItem(id);
    } else {
        setCookie(cookieName, JSON.stringify(shoppingCart), 1);
    }
    updateTotalQty();
}

/**
 * @param {string} id product code
 */
export function getTotalPrice(id) {
    if (getCookie(cookieName) === null) {
        return 0;
    } else {
        shoppingCart = parseToObj(getCookie(cookieName));
        return (
            products.find((product) => product.code == id).price *
            shoppingCart[id]
        );
    }
}

function updateTotalQty() {
    let totalQuantity = document.getElementById("total");
    totalQuantity.innerHTML = `<b>${getTotalQty()}</b>`;
}
```

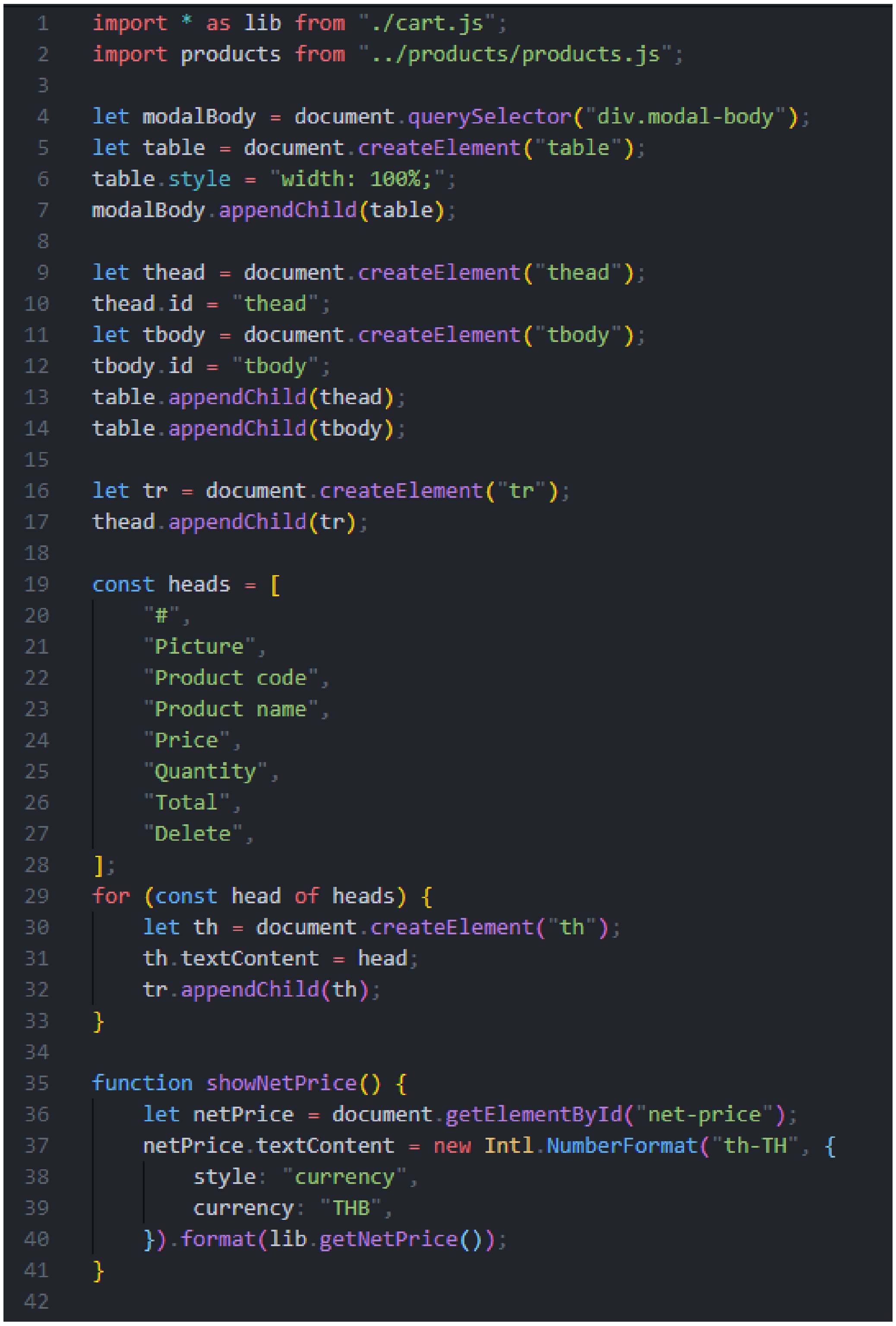
cookieUtil.js:



```
export default class CookieUtil {
    /**
     * @param {string} name name of the cookie
     */
    static getCookie(name) {
        // console.log(`all cookies: ${document.cookie}`);
        let cookieName = `${encodeURIComponent(name)}=`;
        let cookieStart = document.cookie.indexOf(cookieName);
        let cookieValue = null;
        // console.log(`cookieName = ${cookieName}`);
        // console.log(`cookieStart = ${cookieStart}`);

        if (cookieStart > -1) {
            let cookieEnd = document.cookie.indexOf(";", cookieStart);
            // console.log(`cookieEnd = ${cookieEnd}`);
            if (cookieEnd == -1) {
                cookieEnd = document.cookie.length;
            }
            cookieValue = decodeURIComponent(
                document.cookie.substring(
                    cookieStart + cookieName.length,
                    cookieEnd
                )
            );
            // console.log(`cookieValue = ${cookieValue}`);
        }
        return cookieValue;
    }

    /**
     * @param {string} name name of the cookie
     * @param {string} value value of the cookie
     * @param {number} expires duration of the cookie in days
     */
    static setCookie(name, value, expires) {
        let cookieText = `${encodeURIComponent(name)}=${encodeURIComponent(
            value
        )}`;

        const expireDate = new Date();
        expireDate.setTime(
            expireDate.getTime() + expires * 24 * 60 * 60 * 1000
        );

        cookieText += `; expires=${expireDate.toUTCString()}`;
        // cookieText += `; expires=${expires}`;
        // console.log(`cookieText = ${cookieText}`);
        document.cookie = cookieText;
    }

    /**
     * @param {string} name name of the cookie
     */
    static unset(name) {
        CookieUtil.setCookie(name, "", new Date(0));
    }
}
```

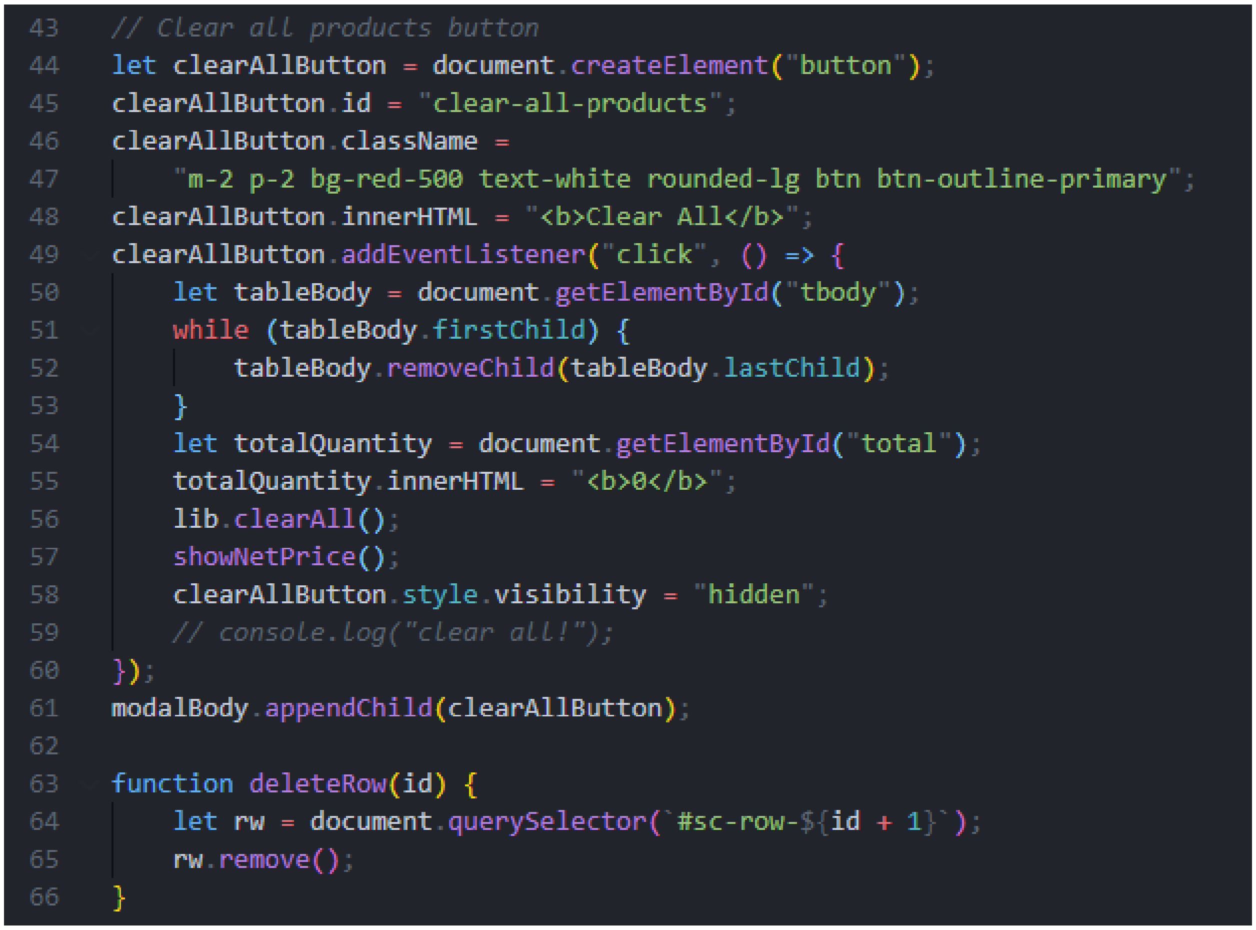
shoppingCart.js:

add function showNetPrice & revise code

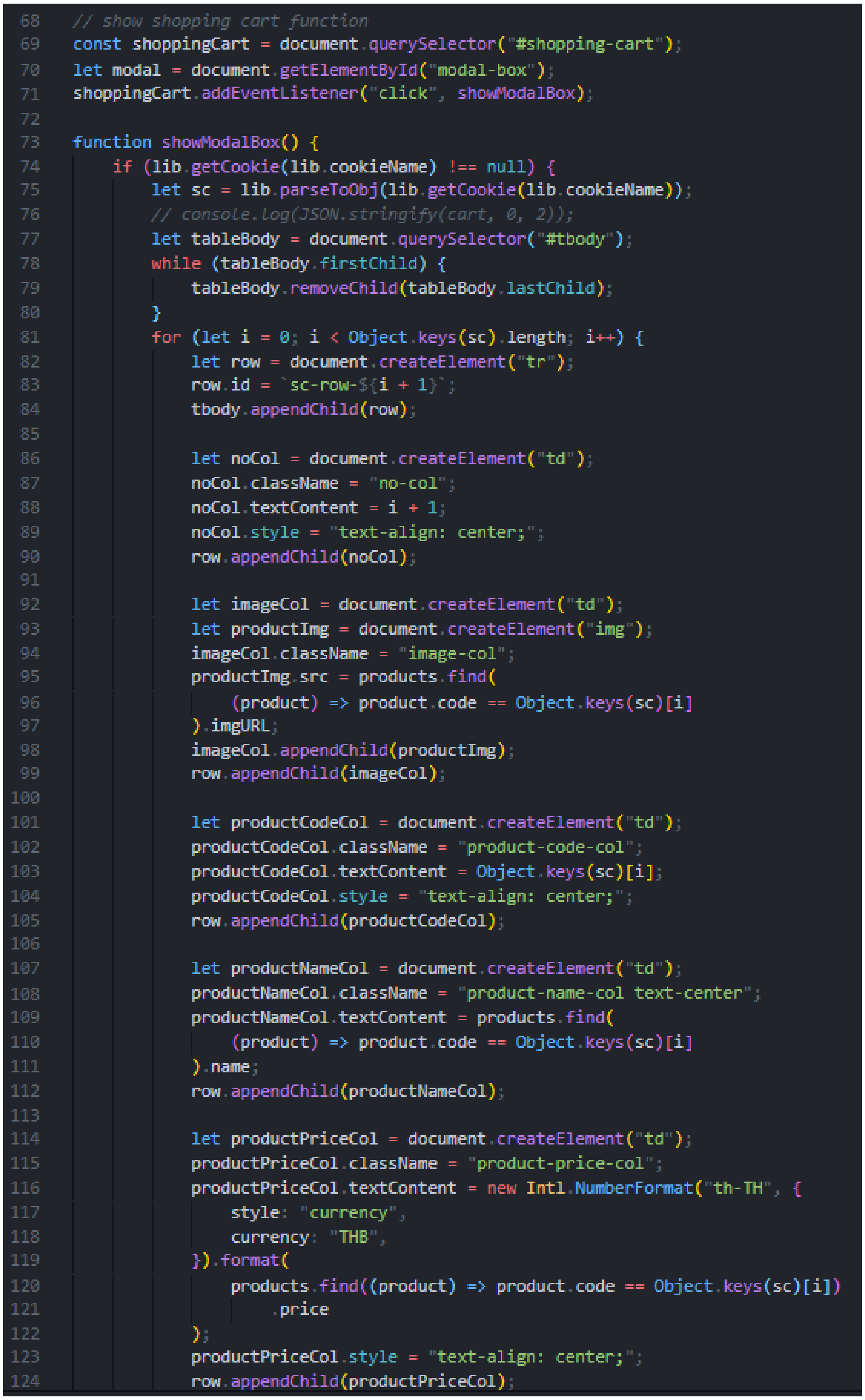


```
import * as lib from "./cart.js";
import products from "../products/products.js";

let modalBody = document.querySelector("div.modal-body");
let table = document.createElement("table");
table.style = "width: 100%;";
modalBody.appendChild(table);

let thead = document.createElement("thead");
thead.id = "thead";
let tbody = document.createElement("tbody");
tbody.id = "tbody";
table.appendChild(thead);
table.appendChild(tbody);

let tr = document.createElement("tr");
thead.appendChild(tr);

const heads = [
    "#",
    "Picture",
    "Product code",
    "Product name",
    "Price",
    "Quantity",
    "Total",
    "Delete",
];
for (const head of heads) {
    let th = document.createElement("th");
    th.textContent = head;
    tr.appendChild(th);
}

function showNetPrice() {
    let netPrice = document.getElementById("net-price");
    netPrice.textContent = new Intl.NumberFormat("th-TH", {
        style: "currency",
        currency: "THB",
    }).format(lib.getNetPrice());
}

```

revise clear all and deleteRow

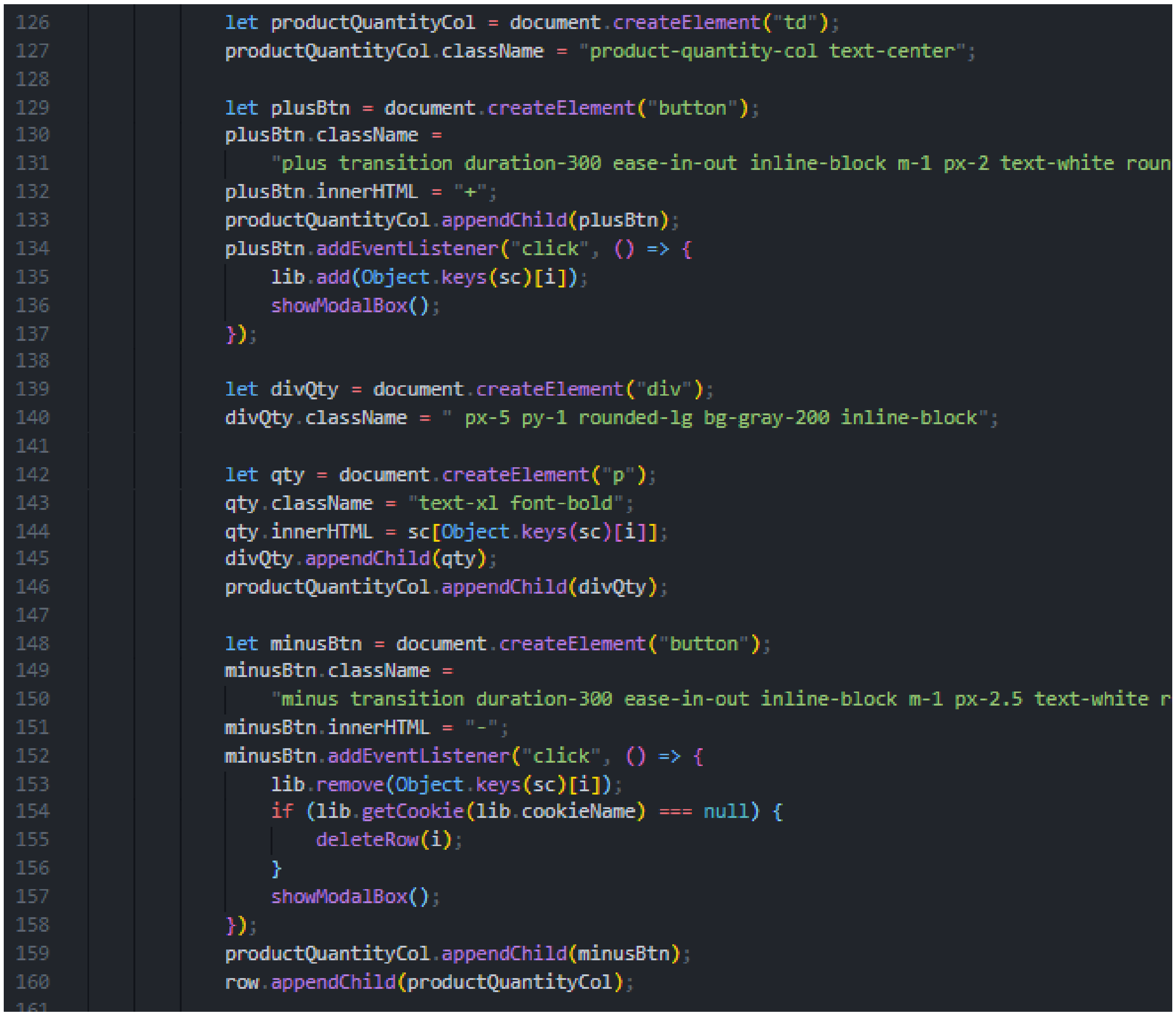


```javascript
// Clear all products button
let clearAllButton = document.createElement("button");
clearAllButton.id = "clear-all-products";
clearAllButton.className =
    "m-2 p-2 bg-red-500 text-white rounded-lg btn btn-outline-primary";
clearAllButton.innerHTML = "<b>Clear All</b>";
clearAllButton.addEventListener("click", () => {
    let tableBody = document.getElementById("tbody");
    while (tableBody.firstChild) {
        tableBody.removeChild(tableBody.lastChild);
    }
    let totalQuantity = document.getElementById("total");
    totalQuantity.innerHTML = "<b>0</b>";
    lib.clearAll();
    showNetPrice();
    clearAllButton.style.visibility = "hidden";
    // console.log("clear all!");
});
modalBody.appendChild(clearAllButton);

function deleteRow(id) {
    let rw = document.querySelector(`#sc-row-${id + 1}`);
    rw.remove();
}
```



```javascript
// show shopping cart function
const shoppingCart = document.querySelector("#shopping-cart");
let modal = document.getElementById("modal-box");
shoppingCart.addEventListener("click", showModalBox);

function showModalBox() {
    if (lib.getCookie(lib.cookieName) !== null) {
        let sc = lib.parseToObj(lib.getCookie(lib.cookieName));
        // console.log(JSON.stringify(cart, 0, 2));
        let tableBody = document.querySelector("#tbody");
        while (tableBody.firstChild) {
            tableBody.removeChild(tableBody.lastChild);
        }
        for (let i = 0; i < Object.keys(sc).length; i++) {
            let row = document.createElement("tr");
            row.id = `sc-row-${i + 1}`;
            tbody.appendChild(row);

            let noCol = document.createElement("td");
            noCol.className = "no-col";
            noCol.textContent = i + 1;
            noCol.style = "text-align: center;";
            row.appendChild(noCol);

            let imageCol = document.createElement("td");
            let productImg = document.createElement("img");
            imageCol.className = "image-col";
            productImg.src = products.find(
                (product) => product.code == Object.keys(sc)[i]
            ).imgURL;
            imageCol.appendChild(productImg);
            row.appendChild(imageCol);

            let productCodeCol = document.createElement("td");
            productCodeCol.className = "product-code-col";
            productCodeCol.textContent = Object.keys(sc)[i];
            productCodeCol.style = "text-align: center;";
            row.appendChild(productCodeCol);

            let productNameCol = document.createElement("td");
            productNameCol.className = "product-name-col text-center";
            productNameCol.textContent = products.find(
                (product) => product.code == Object.keys(sc)[i]
            ).name;
            row.appendChild(productNameCol);

            let productPriceCol = document.createElement("td");
            productPriceCol.className = "product-price-col";
            productPriceCol.textContent = new Intl.NumberFormat("th-TH", {
                style: "currency",
                currency: "THB",
            }).format(
                products.find((product) => product.code == Object.keys(sc)[i])
                    .price
            );
            productPriceCol.style = "text-align: center;";
            row.appendChild(productPriceCol);
```

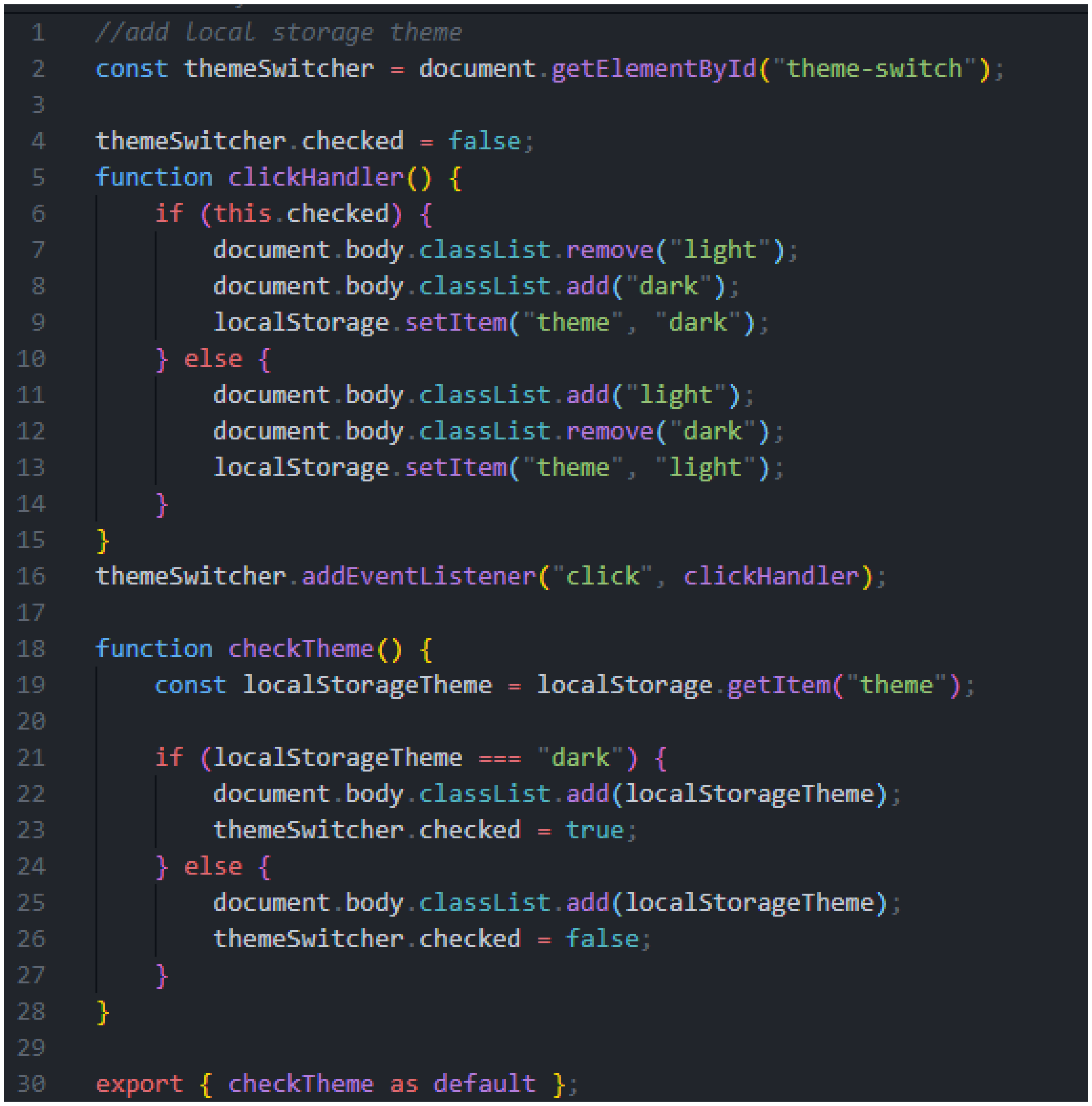
Add plus button and minus button



```
let productQuantityCol = document.createElement("td");
productQuantityCol.className = "product-quantity-col text-center";

let plusBtn = document.createElement("button");
plusBtn.className =
    "plus transition duration-300 ease-in-out inline-block m-1 px-2 text-white roun
plusBtn.innerHTML = "+";
productQuantityCol.appendChild(plusBtn);
plusBtn.addEventListener("click", () => {
    lib.add(Object.keys(sc)[i]);
    showModalBox();
});

let divQty = document.createElement("div");
divQty.className = " px-5 py-1 rounded-lg bg-gray-200 inline-block";

let qty = document.createElement("p");
qty.className = "text-xl font-bold";
qty.innerHTML = sc[Object.keys(sc)[i]];
divQty.appendChild(qty);
productQuantityCol.appendChild(divQty);

let minusBtn = document.createElement("button");
minusBtn.className =
    "minus transition duration-300 ease-in-out inline-block m-1 px-2.5 text-white r
minusBtn.innerHTML = "-";
minusBtn.addEventListener("click", () => {
    lib.remove(Object.keys(sc)[i]);
    if (lib.getCookie(lib.cookieName) === null) {
        deleteRow(i);
    }
    showModalBox();
});
productQuantityCol.appendChild(minusBtn);
row.appendChild(productQuantityCol);
```

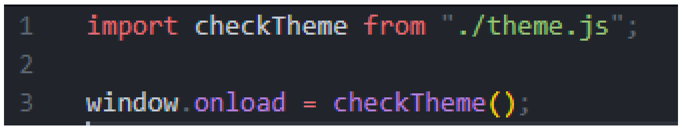
theme.js:

revise code



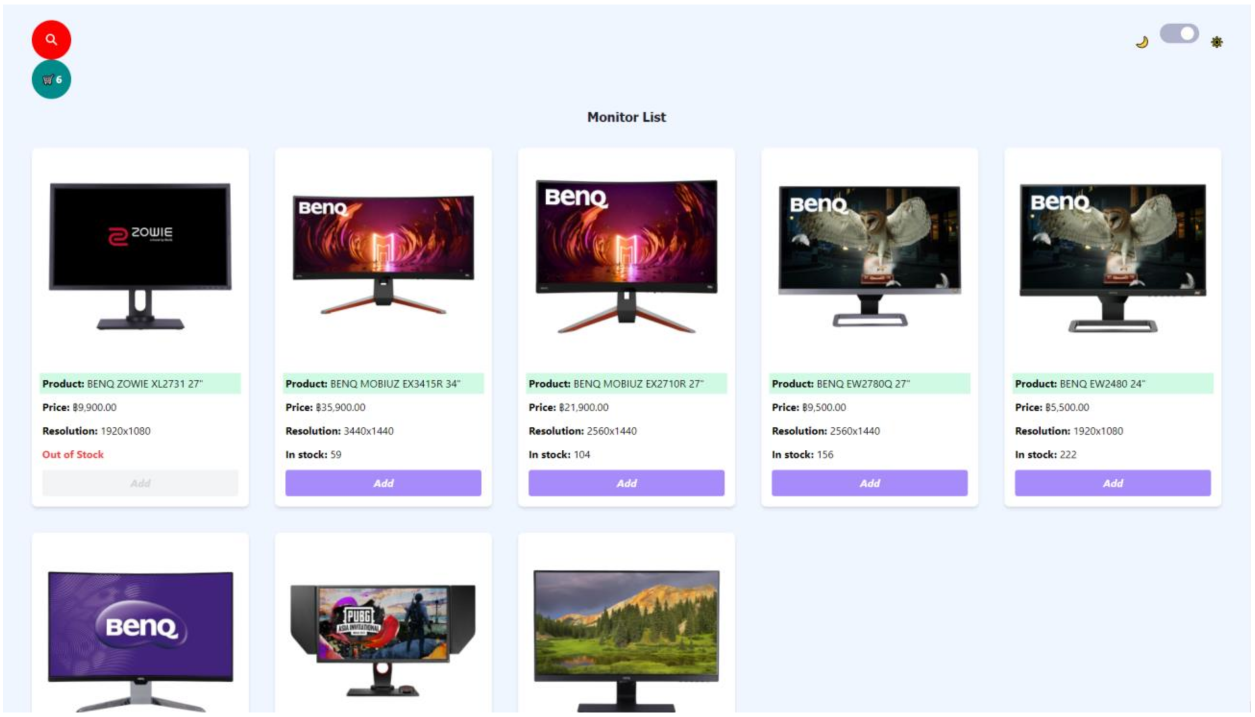
```javascript
//add local storage theme
const themeSwitcher = document.getElementById("theme-switch");

themeSwitcher.checked = false;
function clickHandler() {
    if (this.checked) {
        document.body.classList.remove("light");
        document.body.classList.add("dark");
        localStorage.setItem("theme", "dark");
    } else {
        document.body.classList.add("light");
        document.body.classList.remove("dark");
        localStorage.setItem("theme", "light");
    }
}
themeSwitcher.addEventListener("click", clickHandler);

function checkTheme() {
    const localStorageTheme = localStorage.getItem("theme");

    if (localStorageTheme === "dark") {
        document.body.classList.add(localStorageTheme);
        themeSwitcher.checked = true;
    } else {
        document.body.classList.add(localStorageTheme);
        themeSwitcher.checked = false;
    }
}

export { checkTheme as default };
```

loadStorage.js:



```javascript
import checkTheme from "./theme.js";

window.onload = checkTheme();
```

Output:



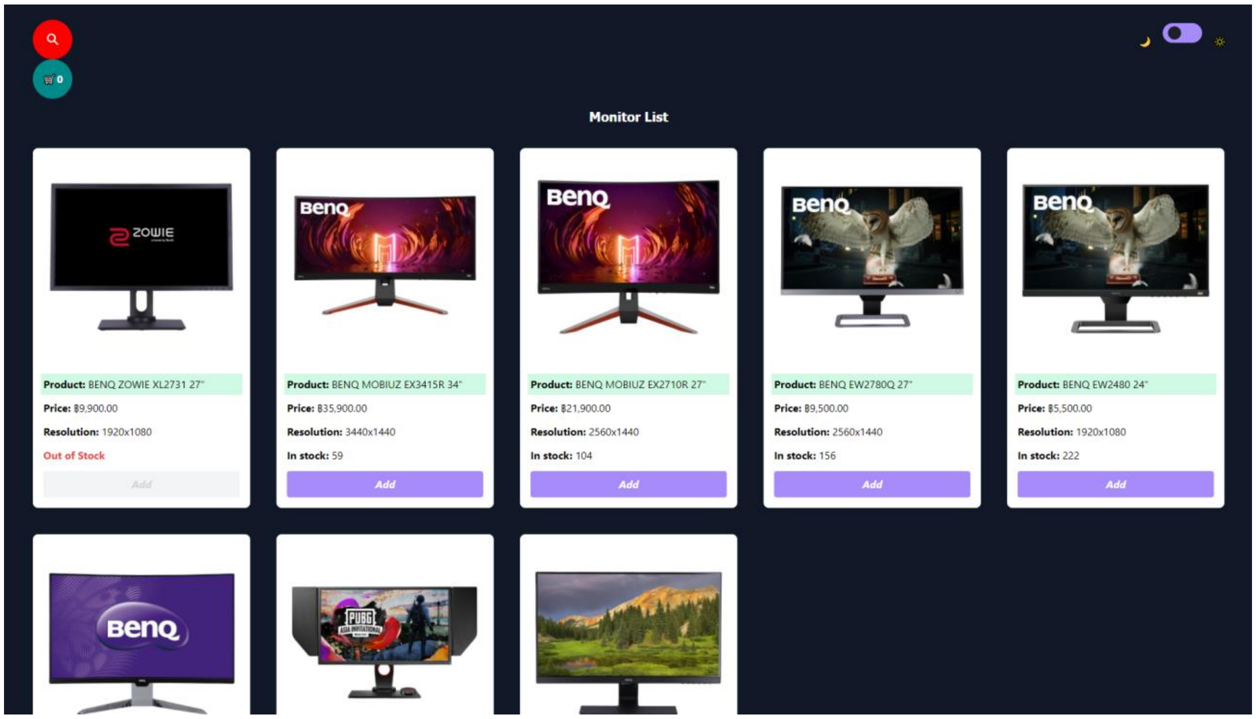
Product: BENQ ZOWIE XL2731 27"
Price: ฿9,900.00
Resolution: 1920x1080
Out of Stock

Product: BENQ MOBIUZ EX3415R 34"
Price: ฿35,900.00
Resolution: 3440x1440
In stock: 59

Product: BENQ MOBIUZ EX2710R 27"
Price: ฿21,900.00
Resolution: 2560x1440
In stock: 104

Product: BENQ EW2780Q 27"
Price: ฿9,500.00
Resolution: 2560x1440
In stock: 156

Product: BENQ EW2480 24"
Price: ฿5,500.00
Resolution: 1920x1080
In stock: 222



Product: BENQ ZOWIE XL2731 27"
Price: ฿9,900.00
Resolution: 1920x1080
Out of Stock

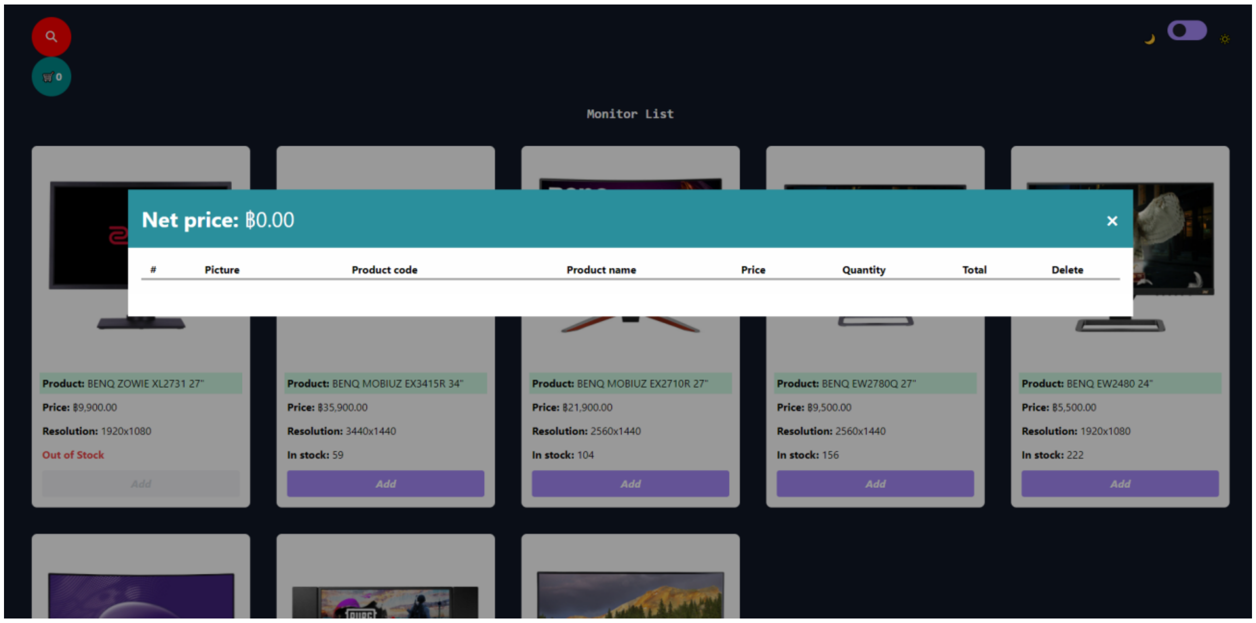
Product: BENQ MOBIUZ EX3415R 34"
Price: ฿35,900.00
Resolution: 3440x1440
In stock: 59

Product: BENQ MOBIUZ EX2710R 27"
Price: ฿21,900.00
Resolution: 2560x1440
In stock: 104

Product: BENQ EW2780Q 27"
Price: ฿9,500.00
Resolution: 2560x1440
In stock: 156

Product: BENQ EW2480 24"
Price: ฿5,500.00
Resolution: 1920x1080
In stock: 222

ถ้าไม่มีสินค้าจะไม่มีปุ่ม clear all



Net price: ฿0.00

#	Picture	Product code	Product name	Price	Quantity	Total	Delete

Product: BENQ ZOWIE XL2731 27"
Price: ฿9,900.00
Resolution: 1920x1080
Out of Stock

Product: BENQ MOBIUZ EX3415R 34"
Price: ฿35,900.00
Resolution: 3440x1440
In stock: 59

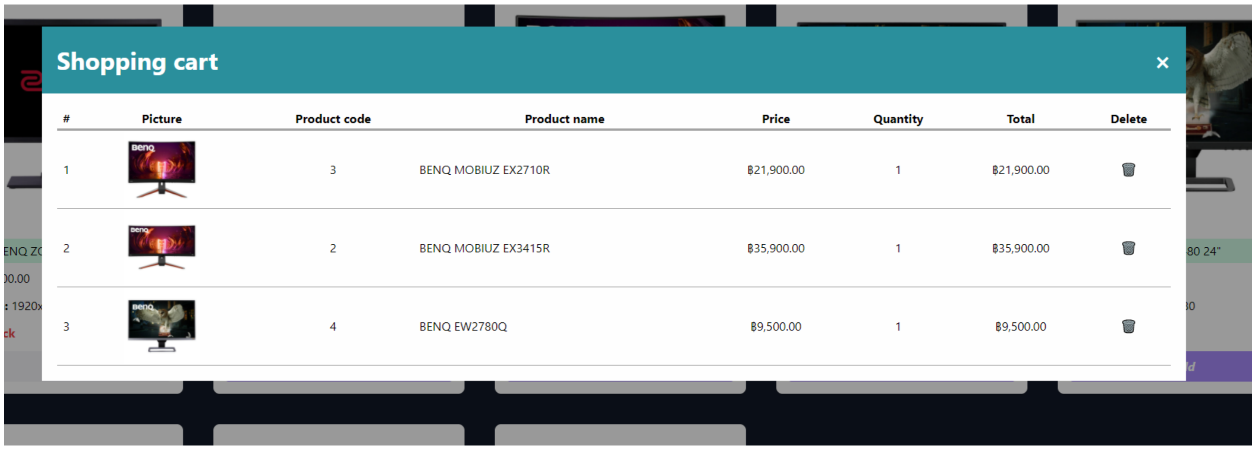
Product: BENQ MOBIUZ EX2710R 27"
Price: ฿21,900.00
Resolution: 2560x1440
In stock: 104

Product: BENQ EW2780Q 27"
Price: ฿9,500.00
Resolution: 2560x1440
In stock: 156

Product: BENQ EW2480 24"
Price: ฿5,500.00
Resolution: 1920x1080
In stock: 222

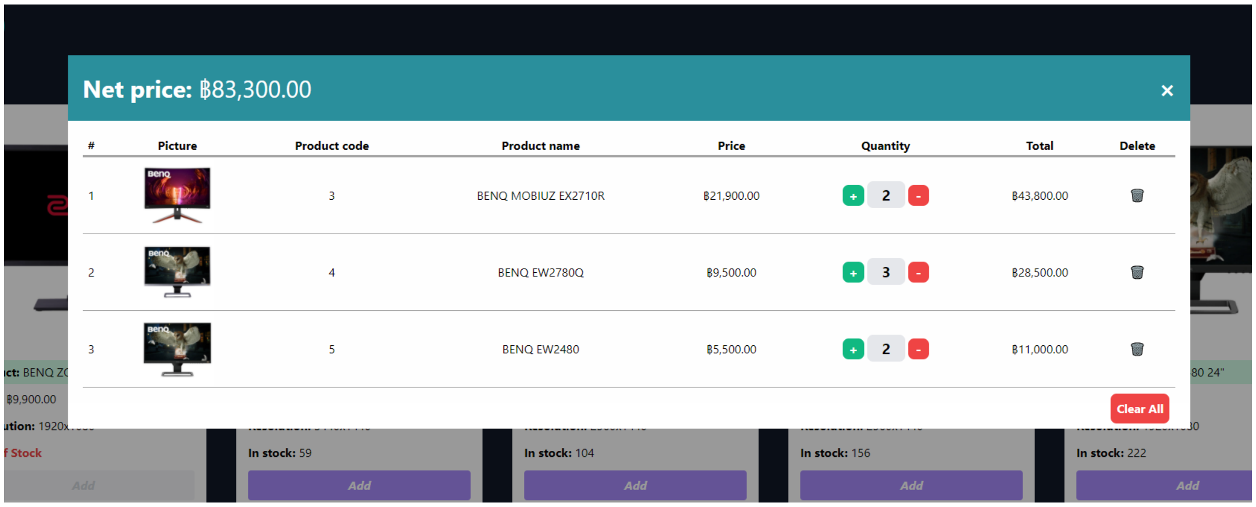
สามารถเพิ่มลดสินค้าได้โดยกดที่ปุ่มบวกและลบ

before



#	Picture	Product code	Product name	Price	Quantity	Total	Delete
1		3	BENQ MOBIUZ EX2710R	฿21,900.00	1	฿21,900.00	
2		2	BENQ MOBIUZ EX3415R	฿35,900.00	1	฿35,900.00	
3		4	BENQ EW2780Q	฿9,500.00	1	฿9,500.00	

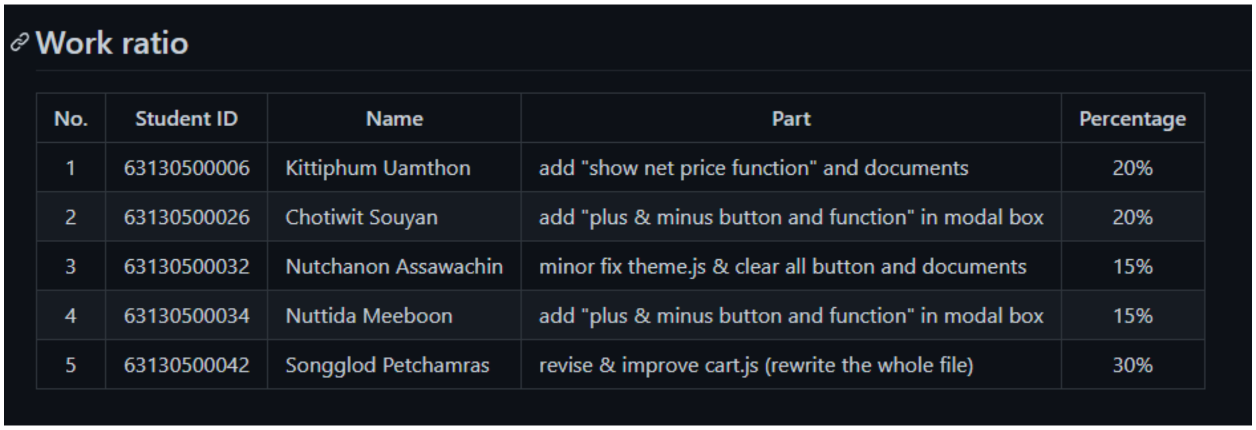
after



#	Picture	Product code	Product name	Price	Quantity	Total	Delete
1		3	BENQ MOBIUZ EX2710R	฿21,900.00	+ 2 -	฿43,800.00	
2		4	BENQ EW2780Q	฿9,500.00	+ 3 -	฿28,500.00	
3		5	BENQ EW2480	฿5,500.00	+ 2 -	฿11,000.00	

GIT LINK **(This Assignment):** ***Click here***

GIT LINK **(All Assignment):** ***Click here***



Work ratio

No.	Student ID	Name	Part	Percentage
1	63130500006	Kittiphum Uamthon	add "show net price function" and documents	20%
2	63130500026	Chotiwit Souyan	add "plus & minus button and function" in modal box	20%
3	63130500032	Nutchanon Assawachin	minor fix theme.js & clear all button and documents	15%
4	63130500034	Nuttida Meeboon	add "plus & minus button and function" in modal box	15%
5	63130500042	Songglod Petchamras	revise & improve cart.js (rewrite the whole file)	30%