

## Problem statement:

Create a website to have them have a shopping portal with all the information about the products like image, price, specifications, etc. Also, categorize the products according to their type. Let the user choose the product for purchase through image or other means. Product need to update in the checkout page and on checkout page, user may remove / delete the products chosen. Shown the cumulative products chosen as a list and total cost of checkout should be updated dynamically.

## HTML:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>products</title>
  <link rel="stylesheet" href="style.css">
</head>
<body class="">

  <div class="container">
    <header>
      <div class="title">PRODUCTS</div>
      <header>
        <nav>
          <ul class="sf-menu">
            <li>
              <a href="index.html" class="current">Home</a>
            </li>
            <li>
              <a href="#">Fashion</a>
              <ul>
                <li>
                  <a href="#men">men's wear</a>
                </li>
                <li>
                  <a href="#women">women's wear</a>
                </li>
                <li>
                  <a href="#kids">kids wear</a>
                </li>
              </ul>
            </li>
          </ul>
        </nav>
      </header>
    </div>
  </body>
</html>
```

```

        </li>
        <li>
            <a href="#watches">watches and
accessories</a>
        </li>
    </ul>
</li>
<li>
    <a href="#">Electronics</a>
    <ul>
        <li>
            <a href="#laptop">laptop and desktops</a>
        </li>
        <li>
            <a href="#mobile">mobiles</a>
        </li>
        <li>
            <a href="#laptopa">laptop accessories</a>
        </li>
        <li>
            <a href="#mobilea">mobile accessories</a>
        </li>
    </ul>
</li>
<li>
    <a href="#">Home and Furniture</a>
    <ul>
        <li>
            <a href="#homef">home furniture</a>
        </li>
        <li>
            <a href="#kitchen">kitchen Furniture</a>
        </li>
        <li>
            <a href="#light">lighting and
electronics</a>
        </li>
        <li>
            <a href="#decor">home decor</a>
        </li>
    </ul>
</li>
<li>
    <a href="#" class="icon-cart">check out</a>
</li>
<li>
    <a href="about.html">about us</a>
</li>

```

```

        </ul>
    </nav>
</header>
<div class="icon-cart">
    <svg aria-hidden="true" xmlns="http://www.w3.org/2000/svg"
fill="none" viewBox="0 0 18 20">
        <path stroke="currentColor" stroke-linecap="round" stroke-
linejoin="round" stroke-width="2" d="M6 15a2 2 0 1 0 0 4 2 2 0 0 0 0-4Zm0
0h8m-8 0-1-4m9 4a2 2 0 1 0 0 4 2 2 0 0 0 0-4Zm-9-4h10l2-7H3m2 7L3 4m0 0-.792-
3H1"/>
    </svg>
    <span>0</span>
</div>
</header>
<div class="listProduct">

    </div>
</div>
<div class="cartTab">
    <h1>Shopping Cart</h1>
    <div class="listCart">

    </div>
    <div class="total">
        <div class="totallog">total:</div>
        <div class="totprice"></div>
    </div>
    <div class="btn">
        <button class="close">CLOSE</button>
        <button class="checkOut">Check Out</button>
    </div>
</div>

<script src="app.js"></script>

</body>
</html>

```

## STYLES:

```

body{
    margin: 0;
    font-family: Poppins;
    background: linear-gradient(to right,green,blue);
}
.container{
    width: 900px;

```

```
margin: auto;
max-width: 90vw;
text-align: center;
padding-top: 10px;
transition: transform .5s;
}
svg{
  width: 30px;
}
header{
  display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 20px 0;
}
.icon-cart{
  position: relative;
}
.icon-cart span{
  position: absolute;
  background-color: red;
  width: 30px;
  height: 30px;
  display: flex;
  justify-content: center;
  align-items: center;
  border-radius: 50%;
  color: #fff;
  top: 50%;
  right: -20px;
}
.title{
  font-size: 25px;
}
.listProduct .item img{
  width: 90%;
  filter: drop-shadow(0 50px 20px #0009);
}
.listProduct{
  display: grid;
  grid-template-columns: repeat(4, 1fr);
  gap: 20px;
}
.listProduct .item{
  background-color: #EEEEEE6;
  padding: 20px;
  border-radius: 20px;
}
```

```

}
.listProduct .item h2{
  font-weight: 500;
  font-size: large;
}
.listProduct .item .price{
  letter-spacing: 7px;
  font-size: small;
}
.listProduct .item button{
  background-color: #353432;
  color: #eee;
  border: none;
  padding: 5px 10px;
  margin-top: 10px;
  border-radius: 20px;
}

/* cart */
.cartTab{
  width: 400px;
  background-color: #353432;
  color: #eee;
  position: fixed;
  top: 0;
  right: -400px;
  bottom: 0;
  display: grid;
  grid-template-rows: 70px 1fr 70px;
  transition: .5s;
}
body.showCart .cartTab{
  right: 0;
}
body.showCart .container{
  transform: translateX(-250px);
}
.cartTab h1{
  padding: 20px;
  margin: 0;
  font-weight: 300;
}
.total{
  display: grid;
  font-size: 30px;
  grid-template-columns: repeat(1, 1fr);
}

```

```
.cartTab .btn{
  height: 60px;
  display: grid;
  grid-template-columns: repeat(2, 1fr);
}
.cartTab button{
  background-color: #E8BC0E;
  border: none;
  font-family: Poppins;
  font-weight: 500;
  cursor: pointer;
}
.cartTab .close{
  background-color: #eee;
}
.listCart .item img{
  width: 100%;
}
.listCart .item{
  display: grid;
  grid-template-columns: 70px 150px 50px 1fr;
  gap: 10px;
  text-align: center;
  align-items: center;
}
.listCart .quantity span{
  display: inline-block;
  width: 25px;
  height: 25px;
  background-color: #eee;
  border-radius: 50%;
  color: #555;
  cursor: pointer;
}
.listCart .quantity span:nth-child(2){
  background-color: transparent;
  color: #eee;
  cursor: auto;
}
.listCart .item:nth-child(even){
  background-color: #eee1;
}
.listCart{
  overflow: auto;
}
.listCart::-webkit-scrollbar{
  width: 0;
}
```

```

@media only screen and (max-width: 992px) {
    .listProduct{
        grid-template-columns: repeat(3, 1fr);
    }
}
@media only screen and (max-width: 768px) {
    .listProduct{
        grid-template-columns: repeat(2, 1fr);
    }
}
nav ul { list-style:none; }

body { font:12px/18px Arial, Helvetica, sans-serif; *font-size:small; }

.sf-menu li a{
    -webkit-transition:-webkit-transform 0.1s ease-in;
    -o-transition:-o-transform 0.1s ease-in;
}

.sf-menu li a:hover{
    -moz-transform:scale(1.3); /*scale up image 1.8x*/
    -webkit-transform:scale(1.3);
    -o-transform:scale(1.3);
}

header {padding:21px 0 25px; width:960px; margin:0 auto; position:relative; z-index:200;background: linear-gradient(to right,rgb(83, 128, 83),rgb(26, 26, 159));}

nav {position:absolute; top:10px; right:0 ;}

.sf-menu ul {
    position:         absolute;
    top:              -999em;
    width:            150px;
    padding:5px 0;
    background:#4c6161;
    margin-left:-10px
}
.sf-menu ul li {
    width:            100%;

```

```

}

.sf-menu li {
    float: left;
    position: relative;
    background:none !important;
}

.sf-menu a {
    display: block;
    position: relative;
}

.sf-menu li:hover ul,
.sf-menu li.sfHover ul {
    left: 0;
    top: 50px;
    z-index: 99;
}

.sf-menu {
    float: left;
}

.sf-menu a {
    text-decoration:none;
    color:rgb(253, 253, 255);
    padding-top:25px;
    padding-bottom:10px
}

.sf-menu a:hover {
    color:white;
}

.sf-menu li {
    float:left;
    font-size:15px;
    padding-right:25px;
}

```

## JSON:

[



```
{
  "id": 1,
  "name": "blue-t-shirt",
  "price": 200,
  "image": "images/blue-t-shirt.jpg"
},
{
  "id": 2,
  "name": "cobalt-blue-t-shirt",
  "price": 250,
  "image": "images/cobalt-blue-t-shirt.jpg"
},
{
  "id": 3,
  "name": "green-t-shirt",
  "price": 290,
  "image": "images/green-t-shirt.jpg"
},
{
  "id": 4,
  "name": "grey-t-shirt",
  "price": 200,
  "image": "images/grey-t-shirt.jpg"
},
{
  "id": 5,
  "name": "purple-t-shirt",
  "price": 300,
  "image": "images/purple-t-shirt.jpg"
},
{
  "id": 6,
  "name": "red-t-shirt",
  "price": 200,
  "image": "images/red-t-shirt.jpg"
},
{
  "id": 7,
  "name": "white-tshirt-template",
  "price": 200,
  "image": "images/white-tshirt-template.jpg"
},
{
  "id": 8,
  "name": "black-t-shirt",
  "price": 200,
```

```

        "image":
"https://www.bing.com/th?id=0IP.aC0n5pzZ80kFX1D37EJJAAHaH8&w=150&h=161&c=8&rs=
1&qlt=90&o=6&pid=3.1&rm=2"
    }
]

```

## JAVASCRIPT:

```

let listProductHTML = document.querySelector('.listProduct');
let listCartHTML = document.querySelector('.listCart');
let iconCart = document.querySelector('.icon-cart');
let iconCartSpan = document.querySelector('.icon-cart span');
let body = document.querySelector('body');
let closeCart = document.querySelector('.close');
let products = [];
let cart = [];
let cost=0;
let costv=0;
let tq=0;

iconCart.addEventListener('click', () => {
    body.classList.toggle('showCart');
})
closeCart.addEventListener('click', () => {
    body.classList.toggle('showCart');
})

const addDataToHTML = () => {
    // remove datas default from HTML

    // add new datas
    if(products.length > 0) // if has data
    {
        products.forEach(product => {
            let newProduct = document.createElement('div');
            newProduct.dataset.id = product.id;
            newProduct.classList.add('item');
            newProduct.innerHTML =
`
<h2>${product.name}</h2>
<div class="price">${product.price}</div>
<button class="addCart">Add To Cart</button>`;
            listProductHTML.appendChild(newProduct);

        });
    }
}

```

```

listProductHTML.addEventListener('click', (event) => {
  let positionClick = event.target;
  if(positionClick.classList.contains('addCart')){
    let id_product = positionClick.parentElement.dataset.id;
    addToCart(id_product);
  }
})
const addToCart = (product_id) => {
  let positionThisProductInCart = cart.findIndex((value) => value.product_id
== product_id);
  if(cart.length <= 0){
    cart = [{
      product_id: product_id,
      quantity: 1
    }];
  }else if(positionThisProductInCart < 0){
    cart.push({
      product_id: product_id,
      quantity: 1
    });
  }else{
    cart[positionThisProductInCart].quantity =
cart[positionThisProductInCart].quantity + 1;
  }
  addCartToHTML();
  addCartToMemory();
}
const addCartToMemory = () => {
  localStorage.setItem('cart', JSON.stringify(cart));
}
const addCartToHTML = () => {
  listCartHTML.innerHTML = '';
  let totalQuantity = 0;
  if(cart.length > 0){
    cart.forEach(item => {
      totalQuantity = totalQuantity + item.quantity;
      let newItem = document.createElement('div');
      newItem.classList.add('item');
      newItem.dataset.id = item.product_id;

      let positionProduct = products.findIndex((value) => value.id ==
item.product_id);
      let info = products[positionProduct];
      listCartHTML.appendChild(newItem);
      cost=info.price
      newItem.innerHTML = `
        <div class="image">
          

```

```

        </div>
        <div class="name">
            ${info.name}
        </div>
        <div class="totalPrice">${info.price * item.quantity}</div>
        <div class="quantity">
            <span class="minus"><</span>
            <span>${item.quantity}</span>
            <span class="plus">>>/span>
        </div>
    `;
}

    })
}
iconCartSpan.innerText = totalQuantity;
if (tq<totalQuantity){
    costv=costv+cost;
}
else{
    costv=costv-cost;
}
//costv=costv+cost;
document.querySelector('.totprice').innerHTML=`<p>${costv}</p>`
tq=totalQuantity;
}

listCartHTML.addEventListener('click', (event) => {
    let positionClick = event.target;
    if(positionClick.classList.contains('minus') ||
positionClick.classList.contains('plus')){
        let product_id = positionClick.parentElement.parentElement.dataset.id;
        let type = 'minus';
        if(positionClick.classList.contains('plus')){
            type = 'plus';
        }
        changeQuantityCart(product_id, type);
    }
})

const changeQuantityCart = (product_id, type) => {
    let positionItemInCart = cart.findIndex((value) => value.product_id ==
product_id);
    if(positionItemInCart >= 0){
        let info = cart[positionItemInCart];
        switch (type) {
            case 'plus':
                cart[positionItemInCart].quantity =
cart[positionItemInCart].quantity + 1;
                break;

```

```

        default:
            let changeQuantity = cart[positionItemInCart].quantity - 1;
            if (changeQuantity > 0) {
                cart[positionItemInCart].quantity = changeQuantity;
            } else {
                cart.splice(positionItemInCart, 1);
            }
            break;
        }
    }
    addCartToHTML();
    addCartToMemory();
}

const initApp = () => {
    // get data product
    fetch('products.json')
        .then(response => response.json())
        .then(data => {
            products = data;
            addDataToHTML();

            // get data cart from memory
            if(localStorage.getItem('cart')){
                cart = JSON.parse(localStorage.getItem('cart'));
                addCartToHTML();
            }
        })
}
initApp();

```

**BY HEMANTH REDDY YANAMALA**