

WEB DEVELOPMENT

LAB-7

NAME- Pushpendra Singh Dangi

ROLL No - 22it3033

BRANCH- INFORMATION TECHNOLOGY

```
1. <html lang="en">
2. <head>
3. <meta charset="UTF-8">
4. <meta name="viewport" content="width=device-width,
   initial-scale=1.0">
5. <title>Shopping List</title>
6.
7.
8. </head>
9. <style>
10.     body {
11.         font-family: Arial, sans-serif;
12.     }
13.
14.
15.     .navbar {
16.         display: flex;
17.         justify-content: space-between;
18.         align-items: center;
```

```
19.     background-color: #333;
20.     color: #fff;
21.     padding: 10px 20px;
22. }
23.
24.
25. .logo {
26.     font-size: 20px;
27.     font-weight: bold;
28. }
29.
30.
31. .menu-toggle {
32.     display: none;
33.     cursor: pointer;
34. }
35.
36.
37. .hamburger-menu {
38.     display: none;
39.     cursor: pointer;
40. }
41.
42.
43. .hamburger-menu div {
44.     width: 30px;
```

```
45.     height: 3px;
46.     background-color: #fff;
47.     margin: 5px;
48. }
49.
50.
51. .line {
52.     width: 25px;
53.     height: 3px;
54.     background-color: #fff;
55.     margin: 5px;
56. }
57.
58.
59. .menu {
60.     display: flex;
61.     list-style: none;
62. }
63.
64.
65. .menu li {
66.     margin-right: 20px;
67. }
68.
69.
70. .menu a {
```

```
71.     color: #fff;
72.     text-decoration: none;
73. }
74.
75.
76. .container {
77.     max-width: 800px;
78.     margin: 0 auto;
79.     padding: 20px;
80. }
81.
82.
83. .search-container {
84.     margin-bottom: 20px;
85. }
86.
87.
88. .items {
89.     display: flex;
90.     flex-wrap: wrap;
91.     justify-content: space-around;
92. }
93.
94.
95. .item {
96.     margin: 10px;
```

```
97.     padding: 10px;
98.     border: 1px solid #ccc;
99.     border-radius: 5px;
100.    width: 200px;
101.    transition: transform 0.3s ease-in-out;
102. }
103.
104.
105. .item:hover {
106.     transform: scale(1.05);
107. }
108.
109.
110. .item-img {
111.     width: 100%;
112.     max-height: 150px;
113.     object-fit: cover;
114.     border-radius: 5px;
115. }
116.
117.
118. .item-desc {
119.     font-size: 14px;
120.     margin-top: 10px;
121.     text-align: center;
122. }
```

```
123.
124.
125. .add-to-cart-btn {
126.     margin-top: 10px;
127.     padding: 5px 10px;
128.     background-color: #333;
129.     color: #fff;
130.     border: none;
131.     border-radius: 5px;
132.     cursor: pointer;
133.     transition: background-color 0.3s;
134. }
135.
136.
137. .add-to-cart-btn:hover {
138.     background-color: #555;
139. }
140.
141.
142. .cart {
143.     display: none;
144.     width: 200px;
145.     border: 1px solid #ccc;
146.     border-radius: 5px;
147.     padding: 10px;
148.     position: fixed;
```

```
149.     top: 50px;
150.     right: 20px;
151.     background-color: #fff;
152. }
153.
154.
155. .cart h2 {
156.     margin-bottom: 10px;
157. }
158.
159.
160. .cart ul {
161.     padding: 0;
162. }
163.
164.
165. .cart-item {
166.     list-style: none;
167.     margin-bottom: 5px;
168. }
169. </style>
170. <body>
171. <div class="navbar">
172.     <div class="logo">Shopping List</div>
173.     <div class="menu-toggle" onclick="toggleMenu()">
174.         <div class="line"></div>
```

```
175.         <div class="line"></div>
176.         <div class="line"></div>
177.     </div>
178.     <div class="hamburger-menu" onclick="toggleMenu()">
179.         <div></div>
180.         <div></div>
181.         <div></div>
182.     </div>
183.     <ul class="menu">
184.         <li><a href="#">Home</a></li>
185.         <li><a href="#">Products</a></li>
186.         <li><a href="#">Contact</a></li>
187.     </ul>
188. </div>
189. <div class="container">
190.     <h1>Adit's List</h1>
191.     <div class="search-container">
192.         <input type="text" id="search" placeholder="Search
193.         items">
194.         <button onclick="filterItems()">Filter</button>
195.     </div>
196.     <div class="items">
197.         <div class="item">
198.             
199.             <p class="item-desc">camera 1</p>
```



```
199.         <button class="add-to-cart-btn"
    onclick="addToCart(this)">Add to Cart</button>
200.     </div>
201.     <div class="item">
202.         
203.         <p class="item-desc">camera 2</p>
204.         <button class="add-to-cart-btn"
    onclick="addToCart(this)">Add to Cart</button>
205.     </div>
206.     <div class="item">
207.         
208.         <p class="item-desc">camera 3</p>
209.         <button class="add-to-cart-btn"
    onclick="addToCart(this)">Add to Cart</button>
210.     </div>
211.     <div class="item">
212.         
213.         <p class="item-desc">cam 4</p>
214.         <button class="add-to-cart-btn"
    onclick="addToCart(this)">Add to Cart</button>
215.     </div>
216.     <div class="item">
217.         
218.         <p class="item-desc">camera 5</p>
219.         <button class="add-to-cart-btn"
    onclick="addToCart(this)">Add to Cart</button>
220.     </div>
221.     <div class="item">
```

```
222.         
223.         <p class="item-desc">camera 6</p>
224.         <button class="add-to-cart-btn"
    onclick="addToCart(this)">Add to Cart</button>
225.     </div>
226.
227. </div>
228. </div>
229. <div class="cart">
230.     <h2>Cart</h2>
231.     <ul id="cart-items">
232.     </ul>
233. </div>
234. <script > function toggleMenu() {
235.     const menu = document.querySelector('.menu');
236.     menu.style.display = menu.style.display === 'none' ? 'flex' :
    'none';
237. }
238.
239.
240. function filterItems() {
241.     const searchValue =
    document.getElementById('search').value.toLowerCase();
242.     const items = document.querySelectorAll('.item');
243.
244.     items.forEach(item => {
```

```
245.         const description =
            item.querySelector('.item-desc').textContent.toLowerCase();
246.         if (description.includes(searchValue)) {
247.             item.style.display = 'block';
248.         } else {
249.             item.style.display = 'none';
250.         }
251.     });
252. }
253.
254.
255. function addToCart(button) {
256.     const item = button.parentElement;
257.     const itemDesc =
        item.querySelector('.item-desc').textContent;
258.
259.     const cartItems = document.getElementById('cart-items');
260.     const newItem = document.createElement('li');
261.     newItem.textContent = itemDesc;
262.     newItem.classList.add('cart-item');
263.
264.     cartItems.appendChild(newItem);
265.
266.     showCart();
267. }
268.
```

```
269.
270. function showCart() {
271.     const cart = document.querySelector('.cart');
272.     cart.style.display = 'block';
273. }</script>
274. <script>
275.     // Function to add item to cart and store it in localStorage
276. function addToCart(button) {
277.     const item = button.parentElement;
278.     const itemDesc =
        item.querySelector('.item-desc').textContent;
279.
280.     // Retrieve existing cart items from localStorage
281.     let cartItems = JSON.parse(localStorage.getItem('cartItems'))
        || [];
282.
283.     // Add new item to cart
284.     cartItems.push(itemDesc);
285.
286.     // Store updated cart items in localStorage
287.     localStorage.setItem('cartItems', JSON.stringify(cartItems));
288.
289.     showCart();
290. }
291.
292. // Function to display cart items from localStorage
```

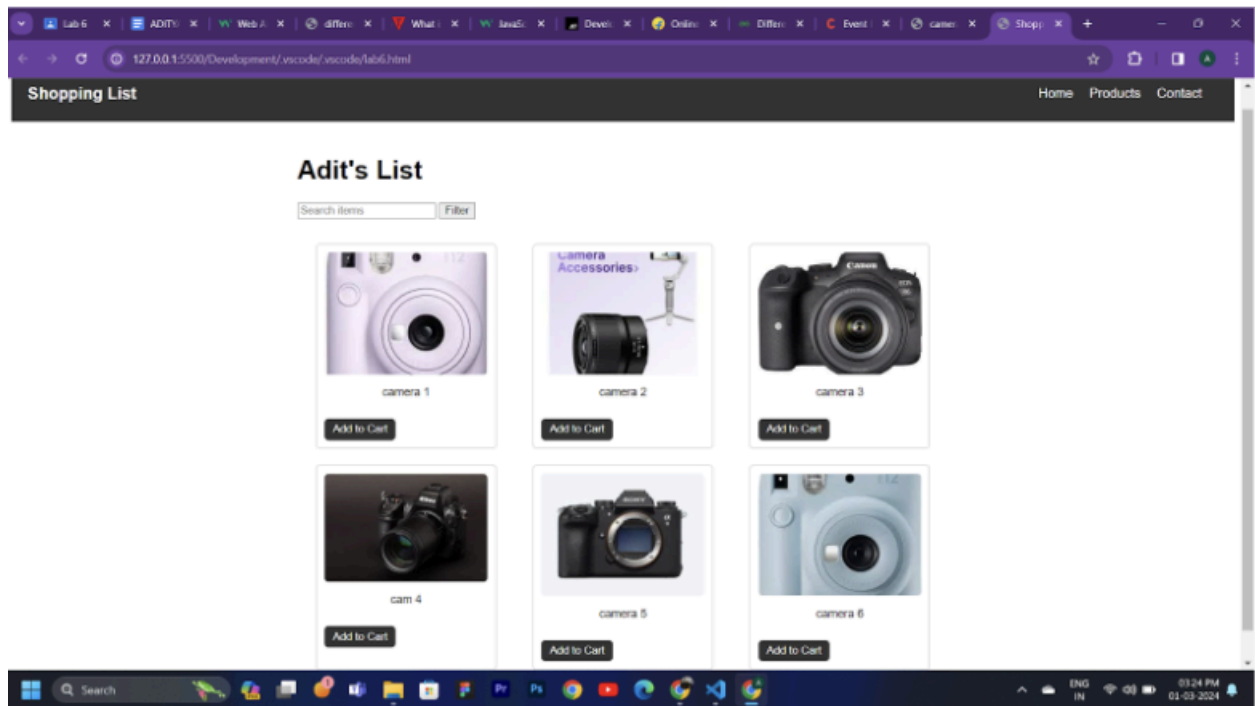
```
293. function showCart() {
294.     const cart = document.querySelector('.cart');
295.     const cartItems = document.getElementById('cart-items');
296.
297.     // Retrieve cart items from localStorage
298.     const storedCartItems =
        JSON.parse(localStorage.getItem('cartItems')) || [];
299.
300.     // Clear existing cart items
301.     cartItems.innerHTML = '';
302.
303.     // Populate cart with stored items
304.     storedCartItems.forEach(itemDesc => {
305.         const newItem = document.createElement('li');
306.         newItem.textContent = itemDesc;
307.         newItem.classList.add('cart-item');
308.         cartItems.appendChild(newItem);
309.     });
310.
311.     cart.style.display = 'block';
312. }
313.
314. // Call showCart function when the page loads to populate the
    cart
315. window.addEventListener('load', showCart);
316.
```

```

317. </script>
318. </body>
319. </html>

```

OUTPUT:-



2.<html lang="en">

```

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

</head>
<style>
    body {

```

```
    font-family: Arial, sans-serif;
}

.navbar {
    display: flex;
    justify-content: space-between;
    align-items: center;
    background-color: #333;
    color: #fff;
    padding: 10px 20px;
}

.logo {
    font-size: 20px;
    font-weight: bold;
}

.menu-toggle {
    display: none;
    cursor: pointer;
}

.hamburger-menu {
    display: none;
    cursor: pointer;
}

.hamburger-menu div {
    width: 30px;
    height: 3px;
    background-color: #fff;
    margin: 5px;
}
```

```
.line {  
  width: 25px;  
  height: 3px;  
  background-color: #fff;  
  margin: 5px;  
}
```

```
.menu {  
  display: flex;  
  list-style: none;  
}
```

```
.menu li {  
  margin-right: 20px;  
}
```

```
.menu a {  
  color: #fff;  
  text-decoration: none;  
}
```

```
.container {  
  max-width: 800px;  
  margin: 0 auto;  
  padding: 20px;  
}
```

```
.search-container {  
  margin-bottom: 20px;  
}
```

```
.items {  
  display: flex;  
  flex-wrap: wrap;
```



```
        justify-content: space-around;
    }

    .item {
        margin: 10px;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 5px;
        width: 200px;
        transition: transform 0.3s ease-in-out;
    }

    .item:hover {
        transform: scale(1.05);
    }

    .item-img {
        width: 100%;
        max-height: 150px;
        object-fit: cover;
        border-radius: 5px;
    }

    .item-desc {
        font-size: 14px;
        margin-top: 10px;
        text-align: center;
    }

    .add-to-cart-btn {
        margin-top: 10px;
        padding: 5px 10px;
        background-color: #333;
        color: #fff;
        border: none;
    }
}
```

```
border-radius: 5px;
cursor: pointer;
transition: background-color 0.3s;
}

.add-to-cart-btn:hover {
  background-color: #555;
}

.cart {
  display: none;
  width: 200px;
  border: 1px solid #ccc;
  border-radius: 5px;
  padding: 10px;
  position: fixed;
  top: 50px;
  right: 20px;
  background-color: #fff;
}

.cart h2 {
  margin-bottom: 10px;
}

.cart ul {
  padding: 0;
}

.cart-item {
  list-style: none;
  margin-bottom: 5px;
}
</style>
<body>
```

```

<div class="navbar">
  <div class="logo">Shopping List</div>
  <div class="menu-toggle" onclick="toggleMenu()">
    <div class="line"></div>
    <div class="line"></div>
    <div class="line"></div>
  </div>
  <div class="hamburger-menu" onclick="toggleMenu()">
    <div></div>
    <div></div>
    <div></div>
  </div>
  <ul class="menu">
    <li><a href="#">Home</a></li>
    <li><a href="#">Products</a></li>
    <li><a href="#">Contact</a></li>
  </ul>
</div>
<div class="container">
  <h1>Adit's List</h1>
  <div class="search-container">
    <input type="text" id="search" placeholder="Search items">
    <button onclick="filterItems()">Filter</button>
  </div>
  <div class="items">
    <div class="item">
      
      <p class="item-desc">camera 1</p>
      <button class="add-to-cart-btn" onclick="addToCart(this)">Add
to Cart</button>
    </div>
    <div class="item">
      
      <p class="item-desc">camera 2</p>
      <button class="add-to-cart-btn" onclick="addToCart(this)">Add
to Cart</button>
    </div>
    <div class="item">
      
      <p class="item-desc">camera 3</p>

```

```

        <button class="add-to-cart-btn" onclick="addToCart(this)">Add
to Cart</button>
    </div>
    <div class="item">
        
        <p class="item-desc">cam 4</p>
        <button class="add-to-cart-btn" onclick="addToCart(this)">Add
to Cart</button>
    </div>
    <div class="item">
        
        <p class="item-desc">camera 5</p>
        <button class="add-to-cart-btn" onclick="addToCart(this)">Add
to Cart</button>
    </div>
    <div class="item">
        
        <p class="item-desc">camera 6</p>
        <button class="add-to-cart-btn" onclick="addToCart(this)">Add
to Cart</button>
    </div>
</div>
<div class="cart">
    <h2>Cart</h2>
    <ul id="cart-items">
    </ul>
</div>
<script > function toggleMenu() {
    const menu = document.querySelector('.menu');
    menu.style.display = menu.style.display === 'none' ? 'flex' : 'none';
}

function filterItems() {
    const searchValue =
document.getElementById('search').value.toLowerCase();
    const items = document.querySelectorAll('.item');

```

```

        items.forEach(item => {
            const description =
item.querySelector('.item-desc').textContent.toLowerCase();
            if (description.includes(searchValue)) {
                item.style.display = 'block';
            } else {
                item.style.display = 'none';
            }
        });
    }

function addToCart(button) {
    const item = button.parentElement;
    const itemDesc = item.querySelector('.item-desc').textContent;

    const cartItems = document.getElementById('cart-items');
    const newItem = document.createElement('li');
    newItem.textContent = itemDesc;
    newItem.classList.add('cart-item');

    cartItems.appendChild(newItem);

    showCart();
}

function showCart() {
    const cart = document.querySelector('.cart');
    cart.style.display = 'block';
}
</script>
<script>
    // Function to add item to cart and store it in localStorage
function addToCart(button) {
    const item = button.parentElement;
    const itemDesc = item.querySelector('.item-desc').textContent;

    // Retrieve existing cart items from localStorage
    let cartItems = JSON.parse(localStorage.getItem('cartItems')) || [];

```

```

    // Add new item to cart
    cartItems.push(itemDesc);

    // Store updated cart items in localStorage
    localStorage.setItem('cartItems', JSON.stringify(cartItems));

    showCart();
}

// Function to display cart items from localStorage
function showCart() {
    const cart = document.querySelector('.cart');
    const cartItems = document.getElementById('cart-items');

    // Retrieve cart items from localStorage
    const storedCartItems = JSON.parse(localStorage.getItem('cartItems'))
    || [];

    // Clear existing cart items
    cartItems.innerHTML = '';

    // Populate cart with stored items
    storedCartItems.forEach(itemDesc => {
        const newItem = document.createElement('li');
        newItem.textContent = itemDesc;
        newItem.classList.add('cart-item');
        cartItems.appendChild(newItem);
    });

    cart.style.display = 'block';
}

// Call showCart function when the page loads to populate the cart
window.addEventListener('load', showCart);

</script>
<script>
    <script>
// Function to update and display access count
function updateAccessCount() {

```

```
// Retrieve current access count from localStorage
let accessCount = localStorage.getItem('accessCount');

// If access count is null (first time visit), set it to 0
if (accessCount === null) {
    accessCount = 0;
} else {
    // If not null, parse the count to integer
    accessCount = parseInt(accessCount);
}

// Increment the access count
accessCount++;

// Update the access count in localStorage
localStorage.setItem('accessCount', accessCount);

// Display the access count on the page
document.getElementById('accessCount').textContent = accessCount;
}

// Call the updateAccessCount function when the page loads
window.addEventListener('load', updateAccessCount);
</script>
</script>
</body>
</html>
```