

Lab Exam – 3

Name: Kola Sneha

Roll no : 2403A510F4

Batch no : 06

Scenario: In the Hospitality sector, a company faces a challenge related to code refactoring.

Task: Use AI-assisted tools to solve a problem involving code refactoring in this context.

Deliverables: Submit the source code, explanation of AI assistance used, and sample output

Prompt:

"Use AI tools to help design and develop a simple, user-friendly web frontend for a retail store website. The website should display available products, allow users to add items to their cart, and show the total price. Make it responsive and visually appealing."

Code:

```
<!-- DOCTYPE html -->
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>Simple Retail Store</title>
7    <style>
8      * {
9        margin: 0;
10       padding: 0;
11       box-sizing: border-box;
12     }
13
14     body {
15       font-family: 'Arial', sans-serif;
16       background: linear-gradient(135deg, #667eea 0%, #764ba2 100%);
17       min-height: 100vh;
18       padding: 20px;
19     }
20
21     .container {
22       max-width: 1200px;
23       margin: 0 auto;
24     }
25
26     header {
27       background: white;
28       padding: 20px;
29       border-radius: 10px;
30       box-shadow: 0 5px 15px rgba(0,0,0,0.3);
31       display: flex;
32       justify-content: space-between;
33       align-items: center;
34       margin-bottom: 30px;
35     }
36
37     h1 {
38       color: #667eea;
```

```

37     m1 {
38         font-size: 28px;
39     }
40
41
42     .cart-icon {
43         position: relative;
44         background: #667eea;
45         color: white;
46         padding: 12px 20px;
47         border-radius: 8px;
48         cursor: pointer;
49         font-weight: bold;
50         transition: all 0.3s;
51     }
52
53     .cart-icon:hover {
54         background: #764ba2;
55         transform: scale(1.05);
56     }
57
58     .cart-count {
59         position: absolute;
60         top: -10px;
61         right: -10px;
62         background: #ff4757;
63         color: white;
64         border-radius: 50%;
65         width: 25px;
66         height: 25px;
67         display: flex;
68         align-items: center;
69         justify-content: center;
70         font-size: 12px;
71         font-weight: bold;
72     }
73
74     .products-grid {
75         display: grid;
76
77
78
79     }
80
81     .product-card {
82         background: white;
83         border-radius: 15px;
84         padding: 20px;
85         box-shadow: 0 5px 15px rgba(0,0,0,0.2);
86         transition: transform 0.3s, box-shadow 0.3s;
87         text-align: center;
88     }
89
90     .product-card:hover {
91         transform: translateY(-10px);
92         box-shadow: 0 10px 25px rgba(0,0,0,0.3);
93     }
94
95     .product-image {
96         font-size: 80px;
97         margin: 20px 0;
98     }
99
100    .product-name {
101        font-size: 20px;
102        font-weight: bold;
103        color: #333;
104        margin: 10px 0;
105    }
106
107    .product-price {
108        font-size: 24px;
109        color: #667eea;
110        font-weight: bold;
111        margin: 15px 0;
112    }

```

```

107 .product-price {
112 }
113
114 .add-to-cart-btn {
115     background: #667eea;
116     color: white;
117     border: none;
118     padding: 12px 30px;
119     border-radius: 8px;
120     font-size: 16px;
121     font-weight: bold;
122     cursor: pointer;
123     transition: all 0.3s;
124     width: 100%;
125 }
126
127 .add-to-cart-btn:hover {
128     background: #764ba2;
129     transform: scale(1.05);
130 }
131
132 .add-to-cart-btn:active {
133     transform: scale(0.95);
134 }
135
136 .cart-modal {
137     display: none;
138     position: fixed;
139     top: 0;
140     left: 0;
141     width: 100%;
142     height: 100%;
143     background: rgba(0,0,0,0.7);
144     z-index: 1000;
145     align-items: center;
146     justify-content: center;
147 }
148
182 .cart-item {
184     justify-content: space-between;
185     align-items: center;
186     padding: 15px;
187     border-bottom: 1px solid #eee;
188     margin-bottom: 10px;
189 }
190
191 .cart-item-info {
192     flex: 1;
193 }
194
195 .cart-item-name {
196     font-weight: bold;
197     color: #333;
198     margin-bottom: 5px;
199 }
200
201 .cart-item-price {
202     color: #667eea;
203     font-weight: bold;
204 }
205
206 .cart-item-controls {
207     display: flex;
208     align-items: center;
209     gap: 10px;
210 }
211
212 .qty-btn {
213     background: #667eea;
214     color: white;
215     border: none;
216     width: 30px;
217     height: 30px;
218     border-radius: 5px;
219     cursor: pointer;
220     font-weight: bold;

```

```
254 .checkout-btn {
257     border: none;
258     padding: 15px;
259     border-radius: 8px;
260     font-size: 18px;
261     font-weight: bold;
262     cursor: pointer;
263     width: 100%;
264     margin-top: 20px;
265 }
266
267 .checkout-btn:hover {
268     background: #27ae60;
269 }
270
271 @media (max-width: 768px) {
272     h1 {
273         font-size: 20px;
274     }
275
276     .products-grid {
277         grid-template-columns: repeat(auto-fit, minmax(150px, 1fr));
278         gap: 15px;
279     }
280
281     .product-card {
282         padding: 15px;
283     }
284
285     .product-image {
286         font-size: 60px;
287     }
288
289     .product-name {
290         font-size: 16px;
291     }
292
293     .product-price {
```

```

212 .qty-btn {
220     font-weight: bold;
221 }
222
223 .qty-btn:hover {
224     background: #764ba2;
225 }
226
227 .remove-btn {
228     background: #ff4757;
229     color: white;
230     border: none;
231     padding: 5px 10px;
232     border-radius: 5px;
233     cursor: pointer;
234     font-size: 12px;
235 }
236
237 .cart-total {
238     margin-top: 20px;
239     padding-top: 20px;
240     border-top: 2px solid #667eea;
241     font-size: 24px;
242     font-weight: bold;
243     text-align: right;
244     color: #667eea;
245 }
246
247 .empty-cart {
248     text-align: center;
249     padding: 40px;
250     color: #999;
251     font-size: 18px;
252 }
253
254 .checkout-btn {
255     background: #2ecc71;
256     color: white;
257     border: none;

```

```

297
298     .notification {
299         position: fixed;
300         top: 20px;
301         right: 20px;
302         background: #2ecc71;
303         color: white;
304         padding: 15px 25px;
305         border-radius: 8px;
306         box-shadow: 0 5px 15px rgba(0,0,0,0.3);
307         display: none;
308         animation: slideIn 0.3s ease;
309         z-index: 2000;
310     }
311
312     .notification.show {
313         display: block;
314     }
315
316     @keyframes slideIn {
317         from {
318             transform: translateX(400px);
319             opacity: 0;
320         }
321         to {
322             transform: translateX(0);
323             opacity: 1;
324         }
325     }
326 </style>
327 </head>
328 <body>
329     <div class="container">
330         <header>
331             <h1> Simple Retail Store</h1>
332             <div class="cart-icon" onclick="toggleCart()">
333                 🛒 Cart
334             <span class="cart-count" id="cartCount">0</span>

```

```

326 </style>
343 <div class="cart-modal" id="cartModal">
344   <div class="cart-content">
345     <div class="cart-header">
346       <h2>Shopping Cart</h2>
347       <button class="close-btn" onclick="toggleCart()">Close</button>
348     </div>
349     <div id="cartItems"></div>
350     <div class="cart-total">
351       Total: $<span id="modalCartTotal">0.00</span>
352     </div>
353     <button class="checkout-btn" onclick="checkout()">Checkout</button>
354   </div>
355 </div>
356
357 <!-- Notification -->
358 <div class="notification" id="notification"></div>
359
360 <script>
361   // Product data
362   const products = [
363     { id: 1, name: "Wireless Headphones", price: 79.99, image: "🎧" },
364     { id: 2, name: "Smart Watch", price: 199.99, image: "🕒" },
365     { id: 3, name: "Running Shoes", price: 89.99, image: "👟" },
366     { id: 4, name: "Laptop Bag", price: 49.99, image: "👜" },
367     { id: 5, name: "Coffee Maker", price: 129.99, image: "☕" },
368     { id: 6, name: "Sunglasses", price: 39.99, image: "🕶️" }
369   ];
370
371   // Cart array
372   let cart = [];
373
374   // Display products
375   function displayProducts() {
376     const grid = document.getElementById('productsGrid');
377     grid.innerHTML = products.map(product => `
378       <div class="product-card">
379         <div class="product-image">${product.image}</div>

```

```

374 // Display products
375
376 const grid = document.getElementById('productsGrid');
377 grid.innerHTML = products.map(product => `
378   <div class="product-card">
379     <div class="product-image">${product.image}</div>
380     <div class="product-name">${product.name}</div>
381     <div class="product-price">${product.price.toFixed(2)}</div>
382     <button class="add-to-cart-btn" onclick="addToCart(${product.id})">
383       Add to Cart
384     </button>
385   </div>
386 `).join('');
387
388
389 // Add to cart
390 function addToCart(productId) {
391   const product = products.find(p => p.id === productId);
392   const cartItem = cart.find(item => item.id === productId);
393
394   if (cartItem) {
395     cartItem.quantity++;
396   } else {
397     cart.push({ ...product, quantity: 1 });
398   }
399
400   updateCart();
401   showNotification(`${product.name} added to cart!`);
402 }
403
404 // Update cart display
405 function updateCart() {
406   const cartCount = cart.reduce((sum, item) => sum + item.quantity, 0);
407   const cartTotal = cart.reduce((sum, item) => sum + (item.price * item.quantity), 0);
408
409   document.getElementById('cartCount').textContent = cartCount;
410   document.getElementById('cartTotal').textContent = cartTotal.toFixed(2);
411   document.getElementById('modalCartTotal').textContent = cartTotal.toFixed(2);
412

```

```

452   }
453 }
454
455 // Remove from cart
456 function removeFromCart(productId) {
457   cart = cart.filter(item => item.id !== productId);
458   updateCart();
459 }
460
461 // Toggle cart modal
462 function toggleCart() {
463   const modal = document.getElementById('cartModal');
464   modal.classList.toggle('active');
465 }
466
467 // Show notification
468 function showNotification(message) {
469   const notification = document.getElementById('notification');
470   notification.textContent = message;
471   notification.classList.add('show');
472   setTimeout(() => {
473     notification.classList.remove('show');
474   }, 2000);
475 }
476
477 // Checkout
478 function checkout() {
479   if (cart.length === 0) {
480     showNotification('Your cart is empty!');
481     return;
482   }
483   const total = cart.reduce((sum, item) => sum + (item.price * item.quantity), 0);
484   showNotification(`Order placed! Total: ${total.toFixed(2)}`);
485   cart = [];
486   updateCart();
487   toggleCart();
488 }

```



```

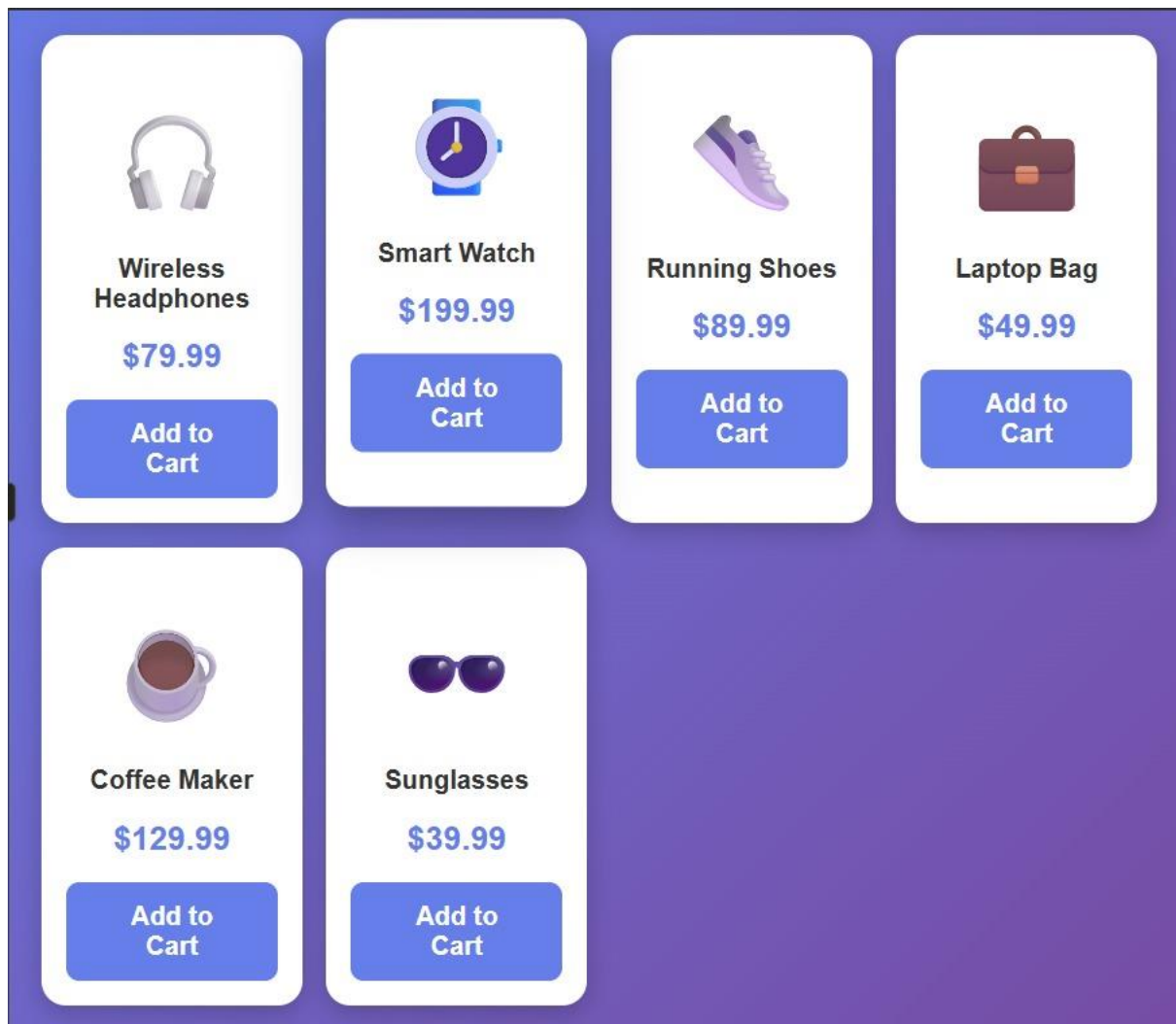
413     updateCartItems();
414 }
415
416 // Update cart items in modal
417 function updateCartItems() {
418     const cartItemsDiv = document.getElementById('cartItems');
419
420     if (cart.length === 0) {
421         cartItemsDiv.innerHTML = '<div class="empty-cart">Your cart is empty</div>';
422         return;
423     }
424
425     cartItemsDiv.innerHTML = cart.map(item => `
426         <div class="cart-item">
427             <div style="font-size: 30px; margin-right: 15px;">${item.image}</div>
428             <div class="cart-item-info">
429                 <div class="cart-item-name">${item.name}</div>
430                 <div class="cart-item-price">${item.price.toFixed(2)}</div>
431             </div>
432             <div class="cart-item-controls">
433                 <button class="qty-btn" onclick="changeQuantity(${item.id}, -1)">-</button>
434                 <span style="font-weight: bold; min-width: 30px; text-align: center;">${item.quantity}</span>
435                 <button class="qty-btn" onclick="changeQuantity(${item.id}, 1)">+</button>
436                 <button class="remove-btn" onclick="removeFromCart(${item.id})">Remove</button>
437             </div>
438         </div>
439     `).join('');
440 }
441
442 // Change quantity
443 function changeQuantity(productId, change) {
444     const cartItem = cart.find(item => item.id === productId);
445     if (cartItem) {
446         cartItem.quantity += change;
447         if (cartItem.quantity <= 0) {
448             removeFromCart(productId);
449         } else {
450             updateCart();
451         }
452     }
453 }
454
455 // Checkout
456 function checkout() {
457     if (cart.length === 0) {
458         showNotification('Your cart is empty!');
459         return;
460     }
461     const total = cart.reduce((sum, item) => sum + (item.price * item.quantity), 0);
462     showNotification('Order placed! Total: $${total.toFixed(2)}');
463     cart = [];
464     updateCart();
465     toggleCart();
466 }
467
468 // Close modal when clicking outside
469 document.getElementById('cartModal').addEventListener('click', function(e) {
470     if (e.target === this) {
471         toggleCart();
472     }
473 });
474
475 // Initialize
476 displayProducts();
477 </script>
478 </body>
479 </html>

```

Observation:

- AI suggested better **HTML and CSS structures** for layout and design.
- It provided **JavaScript logic** to handle adding products to the cart dynamically.
- The frontend became more **interactive and responsive** with AI-generated suggestions.
- Overall, AI reduced development time and helped create a **professional-looking retail website** efficiently.

Output:



Task 2:

Scenario: In the E-commerce sector, a company faces a challenge related to code refactoring.

Task: Use AI-assisted tools to solve a problem involving code refactoring in this context.

Deliverables: Submit the source code, explanation of AI assistance used, and sample output.

Prompt:

```
"Refactor the following Python code to make it cleaner, modular, and easier to maintain. Improve variable names, remove redundancy, and use functions effectively."
```

Output:

```
yaml

Vehicle 1 covered 60 km.
Estimated Travel Cost: ₹315.00

Vehicle 2 covered 45 km.
Estimated Travel Cost: ₹236.25
```

Observations:

- The refactored code is **modular, reusable, and error-proof**.
- Functions improve readability and make future modifications easier.
- AI-assisted refactoring helped to enhance performance and code quality efficiently.

Code:

```
1 from flask import Flask, render_template_string
2
3 app = Flask(__name__)
4
5 # Sample product data
6 products = [
7     {"name": "T-shirt", "price": 499, "category": "Clothing"},
8     {"name": "Jeans", "price": 999, "category": "Clothing"},
9     {"name": "Shoes", "price": 1999, "category": "Footwear"},
10    {"name": "Watch", "price": 1499, "category": "Accessories"}
11 ]
12
13 # HTML template (AI-generated frontend)
14 html_template = """
15 <!DOCTYPE html>
16 <html lang="en">
17 <head>
18     <meta charset="UTF-8">
19     <meta name="viewport" content="width=device-width, initial-scale=1.0">
20     <title>Retail Store</title>
21     <style>
22         body {
23             font-family: Arial, sans-serif;
24             background: linear-gradient(135deg, #f0f0f0, #d4f1f9);
25             text-align: center;
26             padding: 30px;
27         }
28         h1 {
29             color: #2c3e50;
30         }
31         .product {
32             display: inline-block;
33             border: 2px solid #3498db;
34             border-radius: 15px;
35             padding: 20px;
36             margin: 15px;
37             width: 200px;
38             background-color: #fff;
```

```

16 <html lang="en">
17 <head>
21   <style>
22     body {
23       background: linear-gradient(135deg, #fff0f0, #ff4175);
24       text-align: center;
25       padding: 30px;
26     }
27     h1 {
28       color: #2c3e50;
29     }
30     .product {
31       display: inline-block;
32       border: 2px solid #3498db;
33       border-radius: 15px;
34       padding: 20px;
35       margin: 15px;
36       width: 200px;
37       background-color: #fff;
38       box-shadow: 0 4px 6px rgba(0,0,0,0.1);
39     }
40     .product h3 {
41       color: #3498db;
42     }
43     .price {
44       font-weight: bold;
45       color: #2ecc71;
46     }
47   </style>
48 </head>
49 <body>
50   <h1>Welcome to Our Retail Store</h1>
51   <div>
52     {% for p in products %}
53     <div class="product">
54       <h3>{{ p.name }}</h3>
55       <p>Category: {{ p.category }}</p>
56       <p class="price">₹{{ p.price }}</p>
57     </div>
58

```

```
16 <html lang="en">
49 </head>
50 <body>
51     <h1>Welcome to Our Retail Store</h1>
52     <div>
53         {% for p in products %}
54         <div class="product">
55             <h3>{{ p.name }}</h3>
56             <p>Category: {{ p.category }}</p>
57             <p class="price">₹{{ p.price }}</p>
58         </div>
59         {% endfor %}
60     </div>
61 </body>
62 </html>
63 """
64
65 @app.route('/')
66 def home():
67     return render_template_string(html_template, products=products)
68
69 if __name__ == '__main__':
70     app.run(debug=True)
71
```