

Lab-14.3

Name: k.jyoshnoprav

2303A53016

Batch:46

Task 1: AI-Assisted Personal Portfolio Website

Scenario

A student wants to showcase academic projects, technical skills, and contact information through a personal portfolio website. AI tools are used to speed up development while maintaining quality and customization.

Tasks

1. Use GitHub Copilot to generate a semantic HTML structure with sections:

Home

About

Projects

Contact

2. Ask Copilot to suggest responsive CSS using Flexbox or Grid.

3. Customize AI-generated CSS to add:

Hover effects on project cards

Smooth scrolling navigation

Expected Outcome

- Clean and well-structured portfolio website
- Responsive layout for desktop and mobile screens
- Interactive project cards **Code:**

```
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>My Portfolio</title>
7      <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.4.2/css/all.min.css">
8      <style>
9          body {
10              margin: 0;
11              font-family: 'Arial', sans-serif;
12              background: linear-gradient(to bottom, #1a0933, #3c1b6b, #2a1b4d);
13              color: white;
14              overflow-x: hidden;
15          }
16          header {
17              position: relative;
18              height: 100vh;
19              display: flex;
20              flex-direction: column;
21              justify-content: center;
22              align-items: center;
23              text-align: center;
24              background: url('https://via.placeholder.com/1920x1080?text=Space+Background') center/cover no-repeat fixed;
25              padding: 20px;
26          }
27          header h1 {
28              font-size: 3.5em;
29              margin: 0;
30              animation: neon 1.5s infinite alternate;
31          }
32          @keyframes neon {
33              from { text-shadow: 0 0 10px #fff, 0 0 20px #00ddeb, 0 0 30px #00ddeb; }
34              to { text-shadow: 0 0 20px #fff, 0 0 30px #8a2be2, 0 0 40px #8a2be2; }
35          }
36          nav {
37              display: flex;
38              justify-content: center;
39              gap: 15px;
40              margin: 20px 0;
41          }
42          nav a {
```

```
43         color: white;
44         text-decoration: none;
45         font-size: 1.1em;
46         padding: 8px 15px;
47         border: 2px solid #00ddeb;
48         border-radius: 25px;
49         transition: background 0.3s, transform 0.3s;
50     }
51     nav :hover {
52         background: #00ddeb;
53         color: #1a0933;
54         transform: scale(1.1);
55     }
56     section {
57         padding: 60px 20px;
58         max-width: 1200px;
59         margin: 0 auto;
60     }
61     #about {
62         text-align: center;
63     }
64     .profile-photo {
65         width: 180px;
66         height: 180px;
67         border-radius: 50%;
68         object-fit: cover;
69         border: 4px solid #00ddeb;
70         box-shadow: 0 0 20px #8a2be2, 0 0 40px #00ddeb;
71         margin: 20px auto;
72     }
73     #projects {
74         display: grid;
75         grid-template-columns: repeat(auto-fit, minmax(320px, 1fr));
76         gap: 25px;
77     }
78     .skills {
79         display: grid;
80         grid-template-columns: repeat(auto-fit, minmax(300px, 1fr));
81         gap: 20px;
82         background: rgba(255, 255, 255, 0.1);
83         border-radius: 10px;
84         padding: 20px;
85         text-align: center;
86         transition: transform 0.3s;
87     }
88     .skills:hover{
89         transform: translate(-10px);
90     }
91     .project-card {
92         perspective: 1000px;
93         height: 300px;
94     }
95     .project-inner {
96         position: relative;
97         width: 100%;
98         height: 100%;
99         transition: transform 0.6s;
100        transform-style: preserve-3d;
101    }
102    .project-card:hover .project-inner {
103        transform: rotateY(180deg);
104    }
105    .project-front, .project-back {
106        position: absolute;
107        width: 100%;
108        height: 100%;
109        backface-visibility: hidden;
110        border-radius: 10px;
111        background: rgba(255, 255, 255, 0.1);
112        padding: 20px;
113        text-align: center;
114    }
115    .project-back {
116        transform: rotateY(180deg);
117        display: flex;
118        flex-direction: column;
119        justify-content: center;
120    }
121    .project-front img {
122        width: 100%;
123        height: 200px;
124        object-fit: covers;
125        border-radius: 10px;
126    }
127    #contact {
128        text-align: center;
```

```

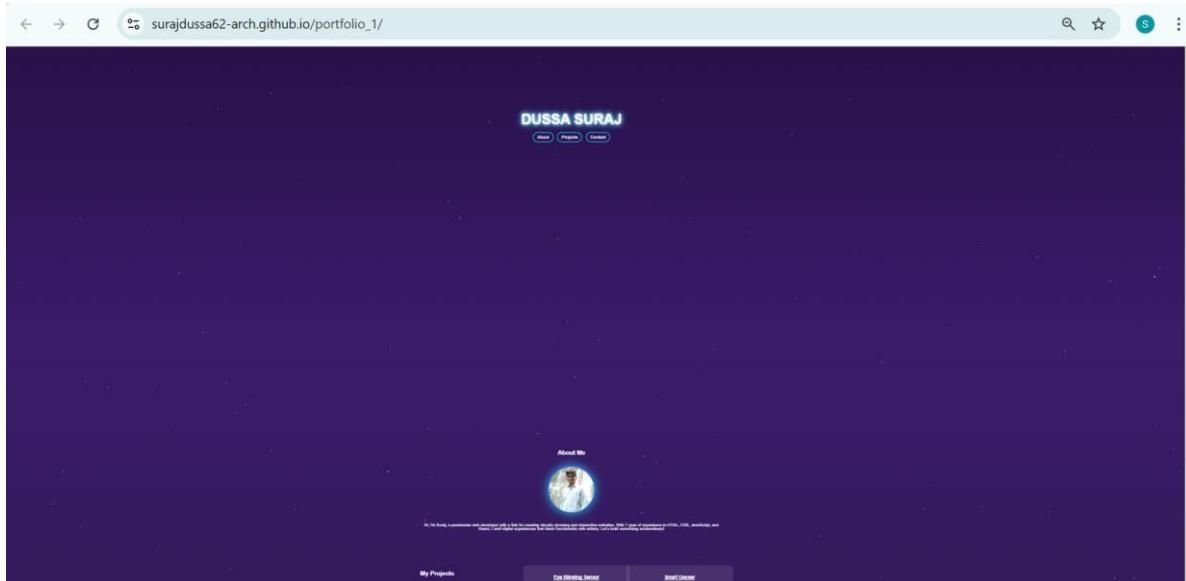
149
150 }
151
152 form {
153   display: flex;
154   flex-direction: column;
155   max-width: 500px;
156   margin: 0 auto;
157   gap: 10px;
158 }
159 input, textarea {
160   padding: 10px;
161   border: none;
162   width: 100%;
163   font-size: 16px;
164   background-color: #fff;
165   background: radial-gradient(rgba(255, 255, 255, 0.9));
166 }
167 button {
168   padding: 10px;
169   background-color: #8a8ae2;
170   border: none;
171   color: white;
172   cursor: pointer;
173   border-radius: 6px;
174   transition: background 0.3s;
175 }
176 button:hover {
177   background: #8080d0;
178 }
179 footer {
180   text-align: center;
181   padding: 10px;
182   background-color: #fff;
183   background: radial-gradient(0, 0, 0, 0);
184 }
185 .social-links {
186   margin-top: 15px;
187   display: flex;
188   justify-content: center;
189   gap: 20px;
190 }
191 .social-links a {
192   color: white;
193   font-size: 1.2em;
194   text-decoration: none;
195   padding: 10px;
196   border-radius: 50px;
197   transition: background 0.3s, transform 0.3s;
198 }
199 .social-links a:hover {
200   background: #8080d0;
201   transform: scale(1.2);
202 }
203 .stars {
204   position: fixed;
205   top: 0;
206   left: 0;
207   width: 1000px;
208   height: 1000px;
209   pointer-events: none;
210   z-index: -1;
211 }
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243

```

```

244     <section id="contact">
245         <h2>Connect with Me</h2>
246         <form>
247             <input type="text" placeholder="Your Name">
248             <input type="email" placeholder="Your Email">
249             <textarea placeholder="Your Message"></textarea>
250             <button type="submit">Send Message</button>
251         </form>
252     </section>
253 
254     <footer>
255         <p>©Copy, 2025 Dussa Suraj. All rights reserved. Created with Passion </p>
256         <div class="social-links">
257             <a href="mailto:surajdussa62@gmail.com" title="Email"><i class="fab fa-email"></i></a>
258             <a href="https://www.linkedin.com/in/suraj-dussa-515aa349/" title="LinkedIn"><i class="fab fa-linkedin"></i></a>
259             <a href="https://github.com/DUSSA084771ake/repositories" title="GitHub"><i class="fab fa-github"></i></a>
260         </div>
261     </footer>
262 
263     <script>
264         // Add twinkling stars for cosmic effect
265         const starContainer = document.querySelector('.stars');
266         for (let i = 0; i < 120; i++) {
267             const star = document.createElement('div');
268             star.classList.add('star');
269             star.style.width = `${Math.random() * 4}px`;
270             star.style.height = star.style.width;
271             star.style.top = `${Math.random() * 100}vh`;
272             star.style.left = `${Math.random() * 100}%;`;
273             star.style.animationDelay = `${Math.random() * 4}s`;
274             starContainer.appendChild(star);
275         }
276     // Smooth scrolling for our links
277     document.querySelectorAll('a').forEach(anchor => {
278         anchor.addEventListener('click', function(e) {
279             e.preventDefault();
280             document.querySelector(this.getAttribute('href')).scrollIntoView({
281                 behavior: 'smooth'
282             });
283         });
284     });
285     </script>
286 </body>
287 </html>
288 
```

Output:



Explanation:

HTML Structure

- create a portfolio website with:
 - **Header** (Name + Navigation) ○ **About**
 - **Section** ○ **Projects Section** ○ **Skills**
 - **Section**
 - **Contact Form**
 - **Footer with Social Links**

CSS Design

- Used **gradient background** for a cosmic look.
 - Added **neon animation** for your name.
 - Created **hover effects** for buttons.
 - Used **flip card animation** for projects.
 - Added **glowing circular profile photo**.
 - Designed responsive grid layout.
-

JavaScript Features

- Created **twinkling stars background**.
- Added **smooth scrolling** when clicking navigation links.

Output:

When you open the file in a browser, you see:

- Full-screen header with glowing **DUSSA SURAJ**
- Navigation buttons (About, Projects, Contact)
- About section with circular photo
- Project cards that flip on hover
- Skills section
- Contact form
- Animated twinkling stars background
- Social media icons in footer

Task 2: AI-Generated Restaurant Landing Page

Scenario

A local restaurant requires a simple yet attractive landing page to display its offerings. The developer uses AI assistance to rapidly generate UI components.

Tasks

1. Use Copilot to create a navigation bar with links:

Home

Menu

Gallery

Contact

2. Generate a menu section styled with CSS cards.

3. Implement a JavaScript-based image slider for the gallery using Copilot suggestions.

4. Customize animations or transitions for better user experience.

Expected Outcome

- Fully functional restaurant landing page
- Responsive navigation bar
- Working image slider with smooth transitions

Code:

```
1  !DOCTYPE html>
2  
3  
4      
5      
6      Elegant Restaurant
7
8      <style>
9          *
10         {
11             margin:0;
12             padding:0;
13             box-sizing:border-box;
14             font-family: 'Segoe UI', sans-serif;
15         }
16
17     html{
18         scroll-behavior:smooth;
19     }
20
21     body{
22         background: #f1f1f1;
23         color: black;
24     }
25
26     /* NAVBAR */
27     .navbar{
28         background: #f1f1f1;
29         padding:20px;
30         position:fixed;
31         width:100%;
32         top:0;
33         z-index:1000;
34     }
35
36     .nav-menu{
37         display:flex;
38         justify-content:center;
39         gap:40px;
40         list-style:none;
41     }
42
43     .nav-menu a{
44         color: black;
45         text-decoration:none;
46         font-size:18px;
47         transition:0.3s;
48     }
49
50     .nav-menu a:hover{
51         color: #d4af37;
52     }
53
54     /* HERO */
55     #home{
56         height:100vh;
57         background: url("https://images.unsplash.com/photo-1517248135467-4c7edcad34c4") center/cover no-repeat;
58         display:flex;
59         flex-direction:column;
60         justify-content:center;
61         align-items:center;
62         text-align:center;
63         padding-top:80px;
64     }
65
66     #home h1{
67         font-size:48px;
68         font-weight:bold;
69         color: black;
70         margin-bottom:10px;
71     }
72
73     #home p{
74         font-size:18px;
75         color: black;
76         margin-bottom:20px;
77     }
78
79     #home .button{
80         background-color: #d4af37;
81         color: white;
82         border: none;
83         padding: 10px 20px;
84         font-size: 16px;
85         cursor: pointer;
86         border-radius: 5px;
87         transition: background-color 0.3s ease;
88     }
89
90     #home .button:hover{
91         background-color: #e0c060;
92     }
```

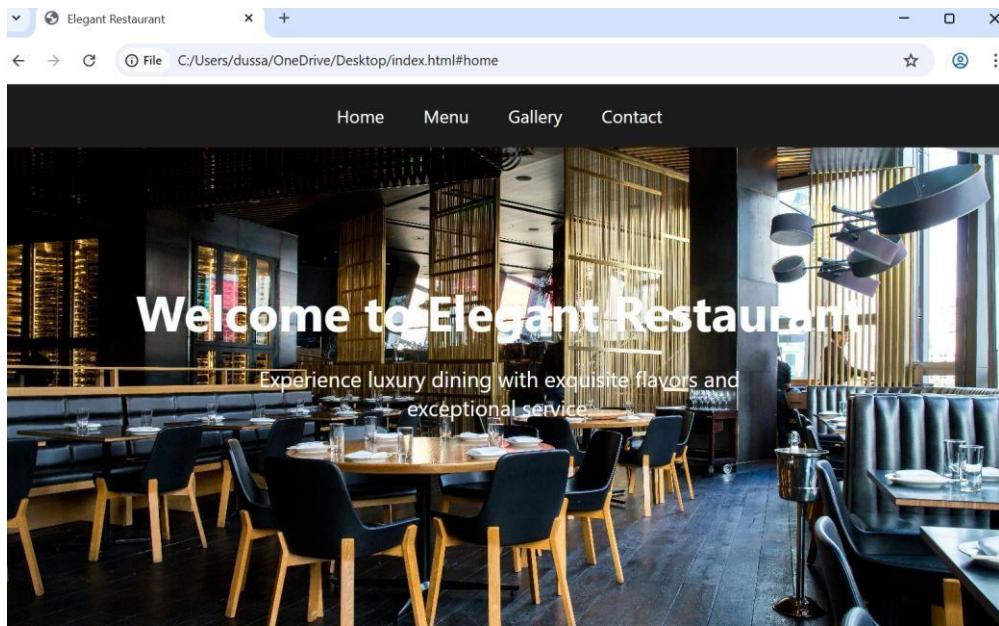
```
2 <html lang="en">
3   <head>
4     <style>
54   #home{
61     text-align:center;
62     padding-top:80px;
63   }
64   #home h1{
65     font-size:50px;
66     margin-bottom:20px;
67   }
68   #home p{
69     font-size:22px;
70     max-width:600px;
71   }
72   /* SECTIONS */
74   section{
75     padding:100px 20px;
76     text-align:center;
77   }
78   h2{
79     font-size:35px;
80     margin-bottom:30px;
81     color:#d4af37;
82   }
83   /* MENU */
84   .menu-items div{
85     margin:12px 0;
86     font-size:20px;
87   }
88   /* GALLERY */
89   .gallery-grid{
90     display:grid;
91     grid-template-columns:repeat(auto-fit,minmax(250px,1fr));
92     gap:20px;
93     margin-top:30px;
94   }
95   .gallery-grid img{
96     width:100%;
97     height:250px;
98     object-fit:cover;
99     border-radius:10px;
100    transition:0.4s;
101  }
102  .gallery-grid img:hover{
103    transform:scale(1.05);
104  }
105  /* CONTACT */
106  .contact-box{
107    max-width:600px;
108    margin:auto;
109    background:#1c1c1c;
110    padding:30px;
111    border-radius:10px;
112  }
113  .contact-box p{
114    margin-bottom:10px;
115    color:#d4af37;
116  }
117  .contact-box p a{
118    color:#d4af37;
119  }
120  .contact-box p a:hover{
121    color:#d4af37;
122  }
```

```

122     margin:10px 0;
123     font-size:18px;
124   }
125 
126 /* FOOTER */
127 footer{
128   background:#f1f1f1;
129   text-align:center;
130   padding:25px;
131   margin-top:50px;
132 }
133 </style>
134 </head>
135 
136 <body>
137 
138   <nav class="navbar">
139     <ul class="nav-menu">
140       <li><a href="#home">Home</a></li>
141       <li><a href="#menu">Menu</a></li>
142       <li><a href="#gallery">Gallery</a></li>
143       <li><a href="#contact">Contact</a></li>
144     </ul>
145   </nav>
146 
147   <!-- HOME -->
148   <section id="home">
149     <h1>Welcome to Elegant Restaurant</h1>
150     <p>Experience luxury dining with exquisite flavors and exceptional service.</p>
151   </section>
152 
153   <!-- MENU -->
154   <section id="menu">
155     <h2>Our Special Menu</h2>
156     <div class="menu-items">
157       <div>● Grilled Steak - $25</div>
158       <div>● Creamy Alfredo Pasta - $18</div>
159       <div>● Wood Fired Pizza - $16</div>
160       <div>● Sushi Platter - $22</div>
161       <div>● Chocolate Lava Cake - $9</div>
162     </div>
163   </section>
164 
165   <!-- GALLERY -->
166   <section id="gallery">
167     <h2>Our Gallery</h2>
168     <div class="gallery-grid">
169       
170       
171       
172       
173     </div>
174   </section>
175 
176   <!-- CONTACT -->
177   <section id="contact">
178     <h2>Contact Us</h2>
179     <div class="contact-box">
180       <p>123 Fine Dining Street, Culinary City</p>
181       <p>+1 234 567 890</p>
182       <p>✉ info@elegantrestaurant.com</p>
183       <p>🕒 Mon - Sun: 10 AM - 11 PM</p>
184     </div>
185   </section>
186 
187   <footer>
188     <p>© 2026 Elegant Restaurant. All Rights Reserved.</p>
189   </footer>
190 
191 </body>
192 </html>
193

```

Output:



Explanation:

- Navbar → Fixed dark menu bar with smooth scrolling links (Home, Menu, Gallery, Contact).
- Theme → Dark luxury design with gold accent color for a premium look.
- Hero Section → Full-screen background image with welcome text.
- Menu Section → Displays special food items with prices.
- Gallery Section → Food and restaurant images in grid layout with hover zoom effect.
- Contact Section → Shows address, phone number, email, and opening hours inside a styled box.
- Footer → Simple copyright section at bottom.

Task 3: AI-Powered Event Registration Form

Scenario

SR University is hosting a technical fest and requires an online registration form with real-time validation to ensure accurate user input.

Tasks

1. Ask Copilot to generate an HTML form with the following fields:

Name

Email

Phone Number

Department

Event Selection

2. Use AI assistance to style the form using CSS for readability and accessibility.

3. Implement JavaScript validation using Copilot:

Email format validation

Phone number length validation

Required field checks

Expected Outcome

- User-friendly and visually appealing form
- Real-time validation messages • Prevention of invalid submissions **Code:**

```
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>Event Registration Form</title>
7   </head>
8   <body>
9     font-family: Arial, sans-serif;
10    max-width: 500px;
11    margin: auto;
12    padding: 20px;
13    background-color: #f5f5f5;
14  </body>
15
16  .form-container {
17    background: #fff;
18    padding: 30px;
19    border-radius: 8px;
20    box-shadow: 0 2px 10px rgba(0,0,0,.1);
21  }
22
23  h1 {
24    color: #333;
25    text-align: center;
26  }
27
28  .form-group {
29    margin-bottom: 20px;
30  }
31
32  label {
33    display: block;
34    margin-bottom: 5px;
35    font-weight: bold;
36    color: #333;
37  }
38
39  input, select {
40    width: 100%;
41    padding: 10px;
42    border: 1px solid #ddd;
43    border-radius: 4px;
44    font-size: 14px;
45    box-sizing: border-box;
46  }
47
48  input:focus, select:focus {
49    outline: none;
50    border-color: #4CAF50;
51  }
52
53  .error {
54    border-color: #D32F2F !important;
55  }
56
57  .error-message {
58    color: #D32F2F;
59    font-size: 12px;
60    margin-top: 5px;
61    display: none;
62  }
63
64  .success-message {
65    color: #4CAF50;
66    ...
67  }
```

```

66     font-size: 16px;
67     margin-top: 5px;
68     display: none;
69   }
70 
71   button {
72     width: 100%;
73     padding: 12px;
74     background-color: #4CAF50;
75     color: white;
76     border: none;
77     border-radius: 4px;
78     font-size: 16px;
79     font-weight: bold;
80     cursor: pointer;
81     transition: background-color 0.3s;
82   }
83 
84   button:hover {
85     background-color: #45a049;
86   }
87 
88   button:disabled {
89     background-color: #cccccc;
90     cursor: not-allowed;
91   }
92 
93 
```

</style>

```

94 </head>
95 <body>
96   <div class="form-container">
97     <h1>Event Registration</h1>
98     <form id="registrationForm">
99       <div class="form-group">
100         <label for="name">Name </label>
101         <input type="text" id="name" name="name" required>
102         <div class="error-message" id="nameError">Name must be at least 2 characters.</div>
103         <div class="success-message" id="nameSuccess"></div>
104       </div>
105 
106       <div class="form-group">
107         <label for="email">Email </label>
108         <input type="email" id="email" name="email" required>
109         <div class="error-message" id="emailError">Enter a valid email address.</div>
110         <div class="success-message" id="emailSuccess"></div>
111       </div>
112 
113       <div class="form-group">
114         <label for="phone">Phone </label>
115         <input type="tel" id="phone" name="phone" required>
116         <div class="error-message" id="phoneError">Enter a valid phone number (10 digits).</div>
117         <div class="success-message" id="phoneSuccess"></div>
118       </div>
119 
120       <div class="form-group">
121         <label for="department">Department </label>
122         <select id="department" name="department" required>
123           <option value="">Select Department</option>
124           <option value="hr">Human Resources</option>
125           <option value="it">IT</option>
126           <option value="sales">Sales</option>
127       </div>
128 
129       <div class="form-group">
130         <label for="event">Event Selection </label>
131         <select id="event" name="event" required>
132           <option value="">Select Event</option>
133           <option value="conference">Annual Conference</option>
134           <option value="workshop">Technical Workshop</option>
135           <option value="webinar">Webinar</option>
136           <option value="networking">Networking Dinner</option>
137         </select>
138         <div class="error-message" id="eventError">Please select an event.</div>
139         <div class="success-message" id="eventSuccess"></div>
140       </div>
141 
142       <button type="submit" id="submitBtn">Register</button>
143     </form>
144   </div>
145 
146   <script>
147     const form = document.getElementById('registrationForm');
148     const nameInput = document.getElementById('name');
149     const emailInput = document.getElementById('email');
150     const phoneInput = document.getElementById('phone');
151     const departmentSelect = document.getElementById('department');
152     const eventSelect = document.getElementById('event');
153     const submitBtn = document.getElementById('submitBtn');

154     const validators = {
155       name: (value) => value.trim().length > 2,
156       email: (value) => /^[^\s]+@[^\s]+\.[^\s]+$/i.test(value),
157       phone: (value) => /^\d{10}\d?/.test(value.replace(/\D/g, '')),
158       department: (value) => value === '',
159       event: (value) => value === ''
160     };

161 
162     function validateField(fieldName, value) {
163       const isValid = validators[fieldName](value);
164       const fieldElement = document.getElementById(fieldName);
165       const errorElement = document.getElementById(fieldName + 'Error');
166       const successElement = document.getElementById(fieldName + 'Success');

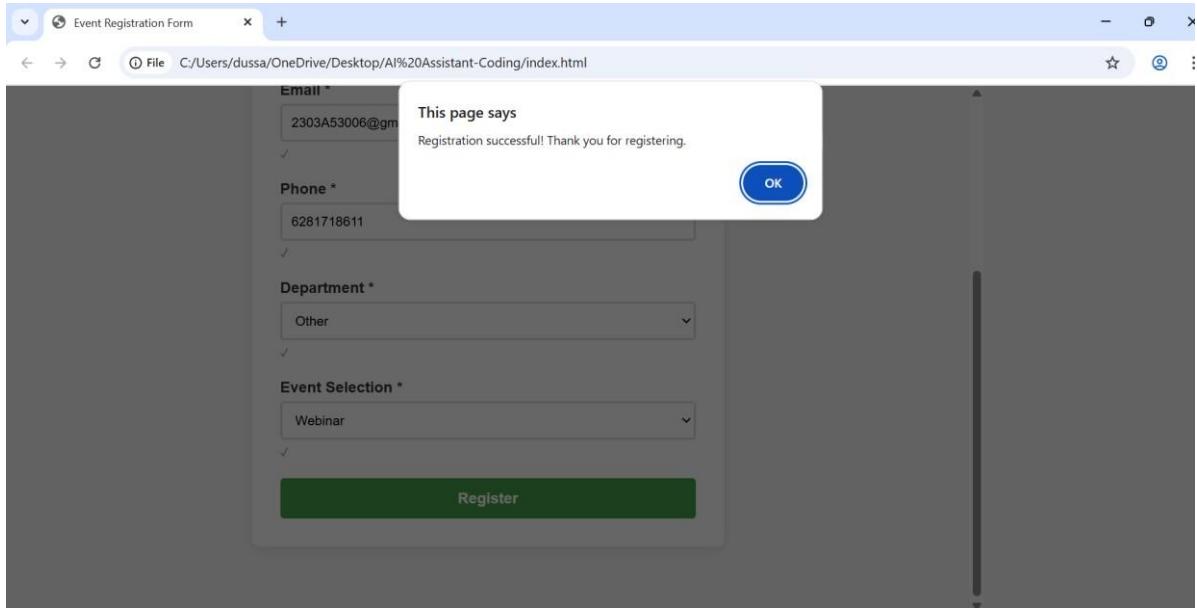
167       if (isValid) {
168         fieldElement.classList.remove('error');
169         errorElement.style.display = 'none';
170         successElement.style.display = 'block';
171       } else {
172         fieldElement.classList.add('error');
173         errorElement.style.display = 'block';
174         successElement.style.display = 'none';
175       }
176 
177       return isValid;
178     }

179   </script>
180 
181   <script>
182     function checkFormValidity() {
183       const nameInput = document.getElementById('name');
184       const emailInput = document.getElementById('email');
185       const phoneInput = document.getElementById('phone');
186       const departmentSelect = document.getElementById('department');
187       const eventSelect = document.getElementById('event');
188 
189       const isValid = validateField('name', nameInput.value) &&
190         validateField('email', emailInput.value) &&
191         validateField('phone', phoneInput.value) &&
192         validateField('department', departmentSelect.value) &&
193         validateField('event', eventSelect.value);

194       submitBtn.disabled = !isValid;
195     }
196 
197     nameInput.addEventListener('input', (e) => validateField('name', e.target.value) && checkFormValidity());
198     emailInput.addEventListener('input', (e) => validateField('email', e.target.value) && checkFormValidity());
199     phoneInput.addEventListener('input', (e) => validateField('phone', e.target.value) && checkFormValidity());
200     departmentSelect.addEventListener('change', (e) => validateField('department', e.target.value) && checkFormValidity());
201     eventSelect.addEventListener('change', (e) => validateField('event', e.target.value) && checkFormValidity());
202 
203     form.addEventListener('submit', (e) => {
204       e.preventDefault();
205       alert('Registration successful! Thank you for registering.');
206       form.reset();
207       submitBtn.disabled = true;
208       document.querySelectorAll('.success-message').forEach(el => el.style.display = 'none');
209     });
210 
211     checkFormValidity();
212   </script>
213 
214 </body>

```

Output:



Explanation:

- HTML Structure → Creates an Event Registration Form with Name, Email, Phone, Department, and Event fields.
- CSS Styling → Adds clean white card design, centered layout, rounded corners, shadow, and green theme.
- Required Fields → All inputs are mandatory (required).
- Error Messages → Hidden by default; shown if input is invalid.
- Success Message (✓) → Displays when a field is valid.
- JavaScript Validators →
 - Name → Minimum 2 characters
 - Email → Valid email format using regex
 - Phone → Exactly 10 digits
 - Department & Event → Must be selected
- Real-time Validation → Checks inputs while typing (input & change events).
- Submit Button → Disabled until all fields are valid.
- Form Submit → Shows success alert, resets form, disables button again.

Task 4: AI-Assisted E-Commerce Product Page

Scenario

A startup is developing a basic e-commerce product page to showcase products and manage simple cart functionality.

Tasks

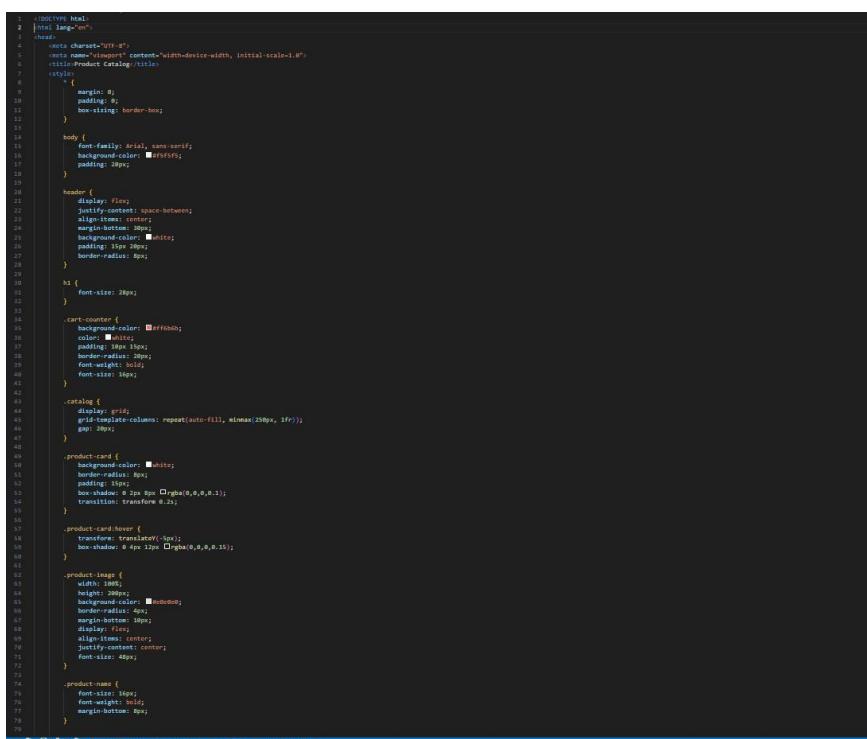
1. Use Copilot to generate a grid-based product catalog using HTML and CSS.
2. Implement an “Add to Cart” feature using JavaScript.
3. Modify AI-generated code to include:

Cart item counter at the top-right corner

Visual feedback when a product is added

Expected Outcome

- Functional product catalog
- Working cart counter • Interactive user experience



The screenshot shows a web browser displaying a product catalog. At the top right of the page, there is a small circular icon containing the number '1'. Below the catalog, there is a section labeled 'Cart' with a sub-section 'Cart (1)'. The main catalog area displays several products in a grid format. Each product card includes a small image, the product name, and a price of '\$19.99'. The overall design is clean and modern, using a light color palette and a sans-serif font.

```
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>Product Catalog</title>
7     <style>
8       body {
9         margin: 0;
10        padding: 0;
11        box-sizing: border-box;
12      }
13
14      body {
15        font-family: Arial, sans-serif;
16        background-color: #f0f0f0;
17        padding: 10px;
18      }
19
20      header {
21        display: flex;
22        justify-content: space-between;
23        align-items: center;
24        margin-bottom: 20px;
25        background-color: #fff;
26        padding: 15px 20px;
27        border-radius: 8px;
28      }
29
30      h1 {
31        font-size: 28px;
32      }
33
34      .cart-counter {
35        background-color: #fff;
36        color: #000;
37        padding: 10px 15px;
38        border-radius: 20px;
39        font-weight: bold;
40        font-size: 16px;
41      }
42
43      .catalog {
44        display: grid;
45        grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
46        gap: 10px;
47      }
48
49      .product-card {
50        background-color: #fff;
51        border-radius: 8px;
52        padding: 15px;
53        box-shadow: 0 2px 8px rgba(0,0,0,0.1);
54        transition: transform 0.2s;
55      }
56
57      .product-card-header {
58        transform: translate(-5px);
59        box-shadow: 4px 0px 10px rgba(0,0,0,0.15);
60      }
61
62      .product-image {
63        width: 100%;
64        height: 200px;
65        background-color: #f0f0f0;
66        border-radius: 4px;
67        margin-bottom: 10px;
68        display: flex;
69        align-items: center;
70        justify-content: center;
71        font-size: 40px;
72      }
73
74      .product-name {
75        font-size: 16px;
76        font-weight: bold;
77        margin-bottom: 8px;
78      }
79    </style>
80  </head>
81  <body>
82    <div>
83      <h1>Product Catalog</h1>
84      <div>
85        <span>Cart (1)</span>
86      </div>
87    </div>
88
89    <div>
90      <div>
91        <img alt="Product image" data-bbox="132 313 678 642"/>
92        <div>
93          <h3>Product Name</h3>
94          <p>$19.99</p>
95        </div>
96      </div>
97    </div>
98
99  </body>
100 </html>
```

```

    .product-price {
      font-size: 18px;
      color: #ff6666;
      font-weight: bold;
      margin-bottom: 10px;
    }

    .btn-add {
      width: 100px;
      padding: 10px;
      background-color: #CAF5B8;
      color: #fff;
      border: none;
      border-radius: 4px;
      cursor: pointer;
      font-weight: bold;
      transition: background-color 0.3s;
    }

    .btn-add:hover {
      background-color: #E45A8A;
    }

    .btn-add:active {
      background-color: #D8B48C;
    }
  


  <header>
    <h1>Product Catalog</h1>
    <div class="cart-counter">Cart: <span id="cartCount">0</span></div>
  </header>

  <div class="catalog" id="catalog"></div>

  <script>
    const products = [
      { id: 1, name: "Laptop", price: "$999", emoji: "💻" },
      { id: 2, name: "Headphones", price: "$79", emoji: "🎧" },
      { id: 3, name: "Smartphone", price: "$699", emoji: "📱" },
      { id: 4, name: "Tablet", price: "$399", emoji: "💻" },
      { id: 5, name: "Camera", price: "$599", emoji: "📷" },
      { id: 6, name: "Watch", price: "$199", emoji: "⌚" },
    ];

    let cartCount = 0;

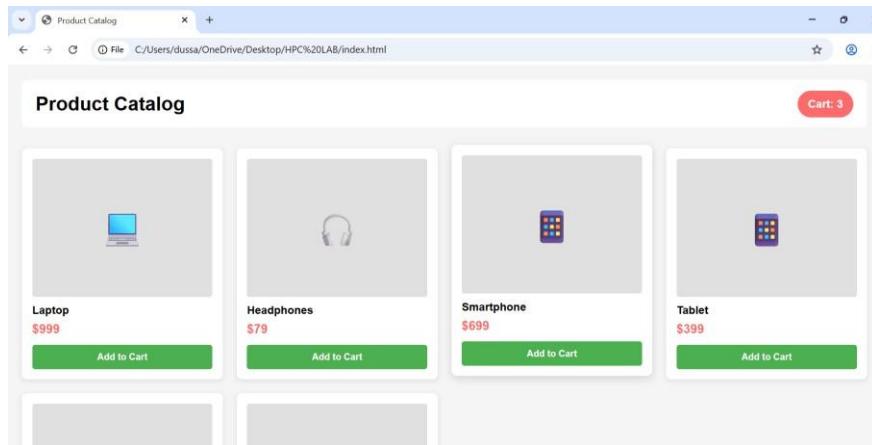
    function renderCatalog() {
      const catalog = document.getElementById("catalog");
      catalog.innerHTML = products.map(product =>
        `<div class="product-card">
          <div class="product-image">${product.emoji}</div>
          ${product.name} ${product.price}
          <div class="product-price">${product.price}</div>
          <button class="btn-add" onclick="addtocart(${product.id})">Add to Cart</button>
        </div>`);
    }

    function addtocart(productId) {
      cartCount++;
      document.getElementById("cartCount").textContent = cartCount;
    }

    renderCatalog();
  </script>
</body>
</html>

```

Output:



Explanation:

- HTML Structure → Creates a Product Catalog page with header and product grid.
- Header Section → Displays page title and cart counter (Cart: 0).
- CSS Styling →
 - Grid layout for products ○ White product cards with shadow
 - Hover effect (card lifts up) ○ Green “Add to Cart” button
 - Red cart counter badge

- Products Array (JavaScript) → Stores product details (id, name, price, emoji).
- renderCatalog() Function → Dynamically displays all products using map() and innerHTML.
- addToCart() Function → Increases cart count and updates it on the page.
- onclick Event → Adds selected product to cart when button is clicked.
- Initial Call → renderCatalog() loads products when page opens.

Task 5: Create a Responsive Web Page Layout Instructions:

- Design a basic web page layout with a header, main content area, and footer using HTML and CSS.
- Use AI to assist in generating responsive CSS for different screen sizes.
- Ensure the layout is clean and visually organized.

Expected Output:

- A responsive web page with:
- Header with navigation links
- Main section with placeholder text/images
- Footer with copyright or contact info
- Layout adapts correctly to desktop, tablet, and mobile screen sizes.

Code:

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Responsive Web Layout</title>
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <style>
      * {
        margin: 0;
        padding: 0;
        box-sizing: border-box;
        font-family: Arial, sans-serif;
      }
      body {
        line-height: 1.6;
      }
      header {
        background: #222;
        color: white;
        padding: 20px 0;
        text-align: center;
      }
      nav {
        display: flex;
        justify-content: center;
        gap: 25px;
        margin-top: 10px;
      }
      nav a {
        color: white;
        text-decoration: none;
        font-weight: bold;
      }
      nav a.active {
        color: #00adb5;
      }
      section {
        padding: 60px 20px;
        min-height: 100vh;
      }
      #home { background: #f0f0f0; }
      #about { background: #e0e0e0; }
      #projects { background: #f0e6e6; }
      #contact { background: #f0f8e9; }

      .project-card, .skill-tag {
        background: white;
        padding: 10px;
        margin: 10px 0;
        border-radius: 8px;
        box-shadow: 0 4px 8px rgba(0,0,0,0.1);
      }
      footer {
        background: #222;
        color: white;
        text-align: center;
        padding: 15px;
      }
    /* Responsive Layout */
    @media (max-width: 768px) {
      nav {
        flex-direction: column;
        align-items: center;
      }
      section {
        padding: 40px 15px;
      }
    }
  </style>
</head>
<body>

  <header id="header">
    <h1>My Responsive Website</h1>
    <nav>
      <a href="#home" class="nav-link">Home</a>
      <a href="#about" class="nav-link">About</a>
      <a href="#projects" class="nav-link">Projects</a>
      <a href="#contact" class="nav-link">Contact</a>
    </nav>
  </header>

  <section id="home">
    <h2>Home</h2>

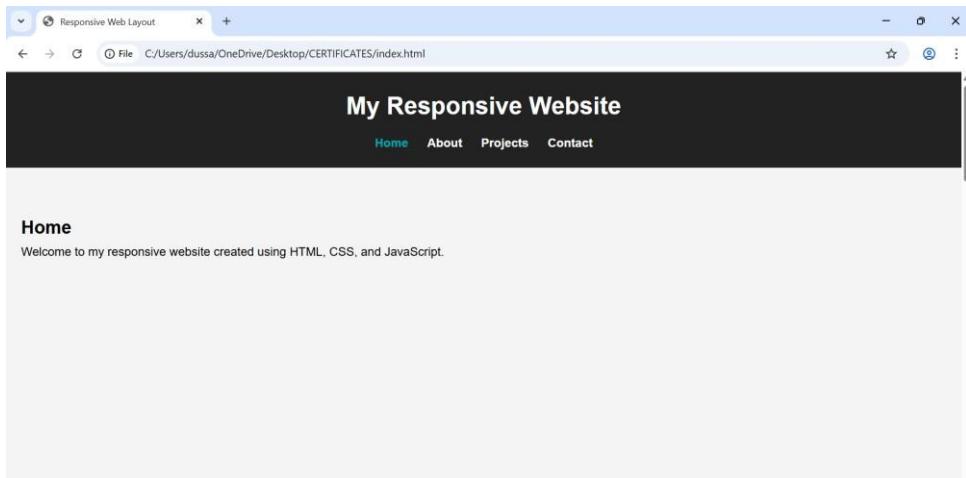
```

```

1<!DOCTYPE html>
2<html>
3    <p>Welcome to my responsive website created using HTML, CSS, and JavaScript.</p>
4</html>
5<body>
6    <div id="header">
7        <h1>My Portfolio</h1>
8        <nav>
9            <a href="#">Home</a>
10           <a href="#">About</a>
11           <a href="#">Skills</a>
12           <a href="#">Projects</a>
13           <a href="#">Contact</a>
14       </nav>
15   </div>
16   <div id="about">
17       <h2>About</h2>
18       <div class="skill-tag">HTML</div>
19       <div class="skill-tag">CSS</div>
20       <div class="skill-tag">JavaScript</div>
21   </div>
22   <div id="projects">
23       <h2>Projects</h2>
24       <div class="project-card">Project 1 - Portfolio Website</div>
25       <div class="project-card">Project 2 - E-commerce Page</div>
26   </div>
27   <div id="contact">
28       <h2>Contact</h2>
29       <form id="contactForm">
30           <input type="text" id="name" placeholder="Name">
31           <input type="email" id="email" placeholder="Email">
32           <input type="text" id="subject" placeholder="Subject">
33           <textarea id="message" placeholder="Message"></textarea>
34           <button type="submit">Send Message</button>
35       </form>
36   </div>
37   </div>
38   <div id="footer">
39       <p>© 2026 My Website | All Rights Reserved</p>
40   </div>
41 </body>
42 <script>
43     // Smooth Scrolling
44     document.querySelectorAll('a[href="#"]')
45     .forEach(anchor => {
46         anchor.addEventListener('click', function (e) {
47             e.preventDefault();
48             const target = document.querySelector(this.getAttribute('href'));
49             target.scrollIntoView({ behavior: 'smooth' });
50         });
51     });
52     // Contact Form Validation
53     const contactForm = document.getElementById('contactForm');
54
55     contactForm.addEventListener('submit', function(e) {
56         e.preventDefault();
57
58         const name = document.getElementById('name').value;
59         const email = document.getElementById('email').value;
60         const subject = document.getElementById('subject').value;
61         const message = document.getElementById('message').value;
62
63         if (!name || !email || !subject || !message) {
64             alert('Please fill all fields');
65             return;
66         }
67
68         const emailPattern = /^[^\s@]+@[^\s@]+\.[^\s@]{2,}$/i;
69         if (!emailPattern.test(email)) {
70             alert('Invalid Email');
71             return;
72         }
73
74         alert('Message Sent Successfully!');
75         contactForm.reset();
76     });
77
78     // Active Navigation Highlight
79     window.addEventListener('scroll', () => {
80         let current = '';
81
82         document.querySelectorAll('section')
83         .forEach(section => {
84             if (section.offsetTop <= window.pageYOffset + 200) {
85                 current = section.getAttribute('id');
86             }
87         });
88
89         document.querySelectorAll('.nav-link')
90         .forEach(link => {
91             link.classList.remove('active');
92             if (link.getAttribute('href').includes(current)) {
93                 link.classList.add('active');
94             }
95         });
96     });
97
98     // Scroll Animations
99     const observer = new IntersectionObserver(entries => {
100         entries.forEach(entry => {
101             if (entry.isIntersecting) {
102                 entry.target.style.opacity = "1";
103                 entry.target.style.transform = "translateY(0)";
104             }
105         }, { threshold: 0.1 });
106
107         document.querySelectorAll('.project-card, .skill-tag')
108         .forEach(el => {
109             el.style.opacity = "0";
110             el.style.transform = "translateY(20px)";
111             el.style.transition = "0.6s";
112             observer.observe(el);
113         });
114
115     console.log("Responsive Web Page Loaded Successfully!");
116
117     </script>
118 </body>
119 </html>

```

Output:



Explanation

- The HTML creates a page with header, navigation, sections, contact form, and footer.
- CSS uses Flexbox for navigation and a media query to make the layout responsive.
- JavaScript enables smooth scrolling when clicking menu links.
- The contact form includes basic validation for empty fields and email format.
- Active navigation links highlight based on scroll position.
- Scroll animations make project cards and skills appear smoothly.