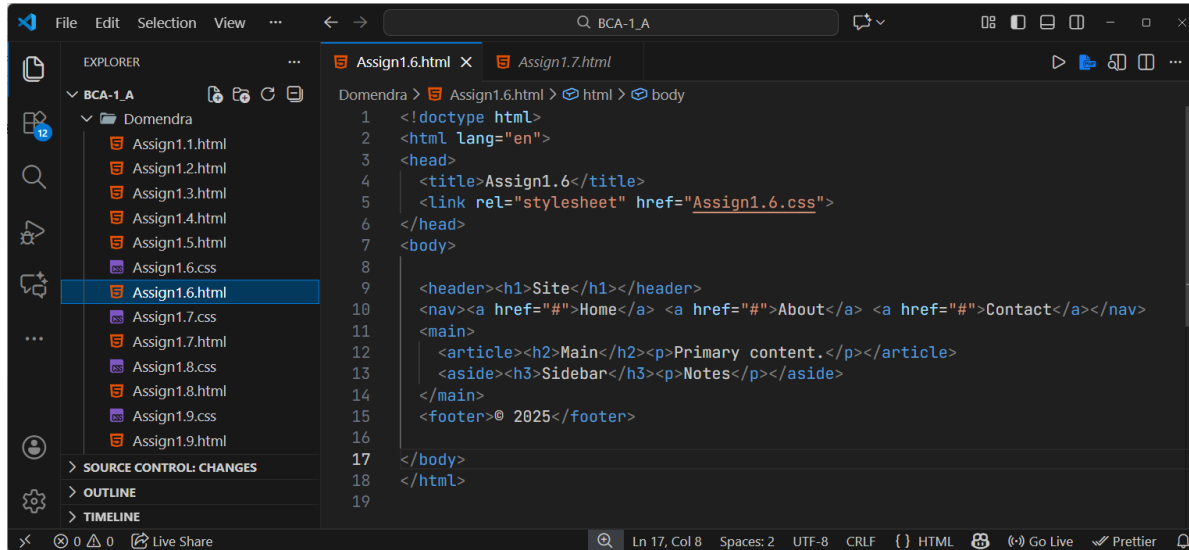
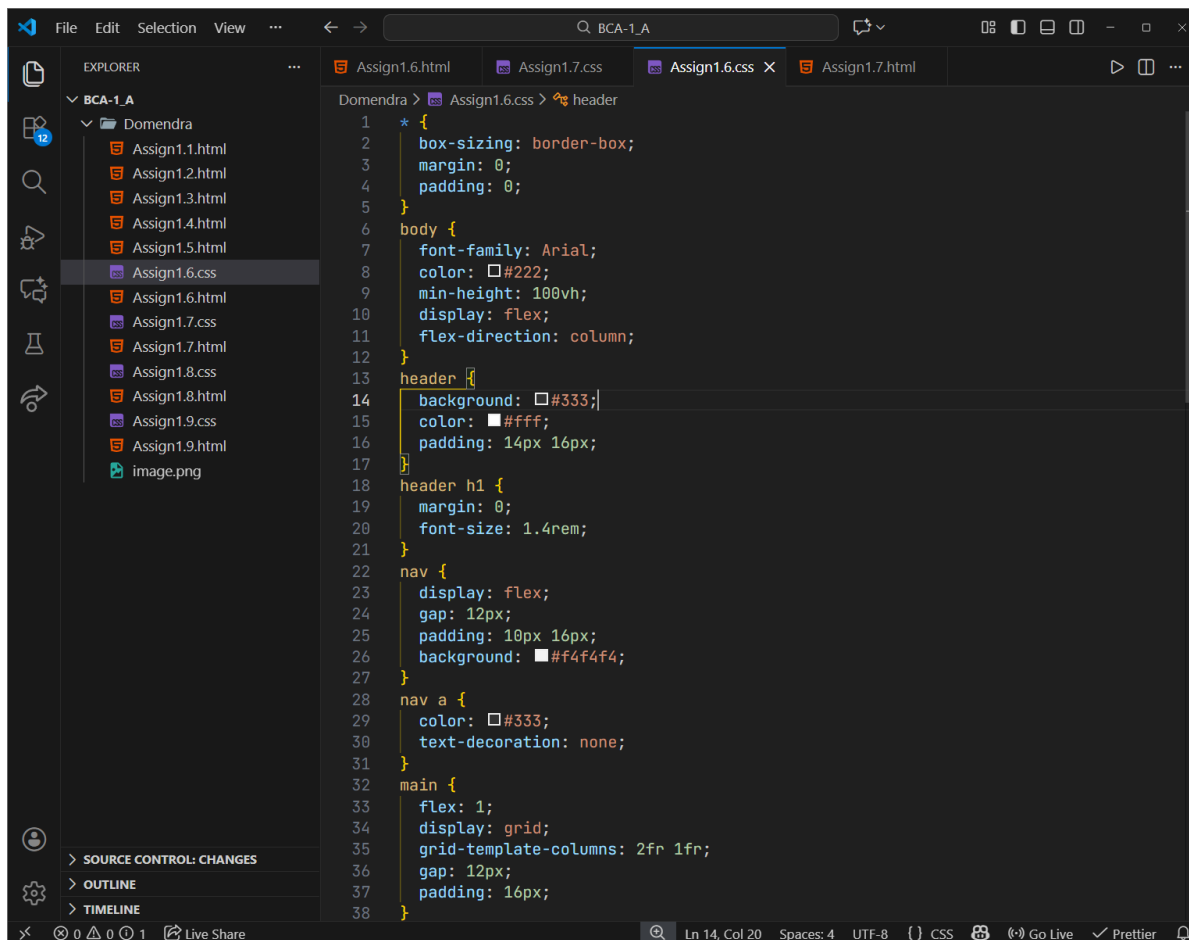


2.1. Styling a Web Page Layout: Create a simple web page using HTML and CSS that includes a header, navigation bar, main content, and footer.

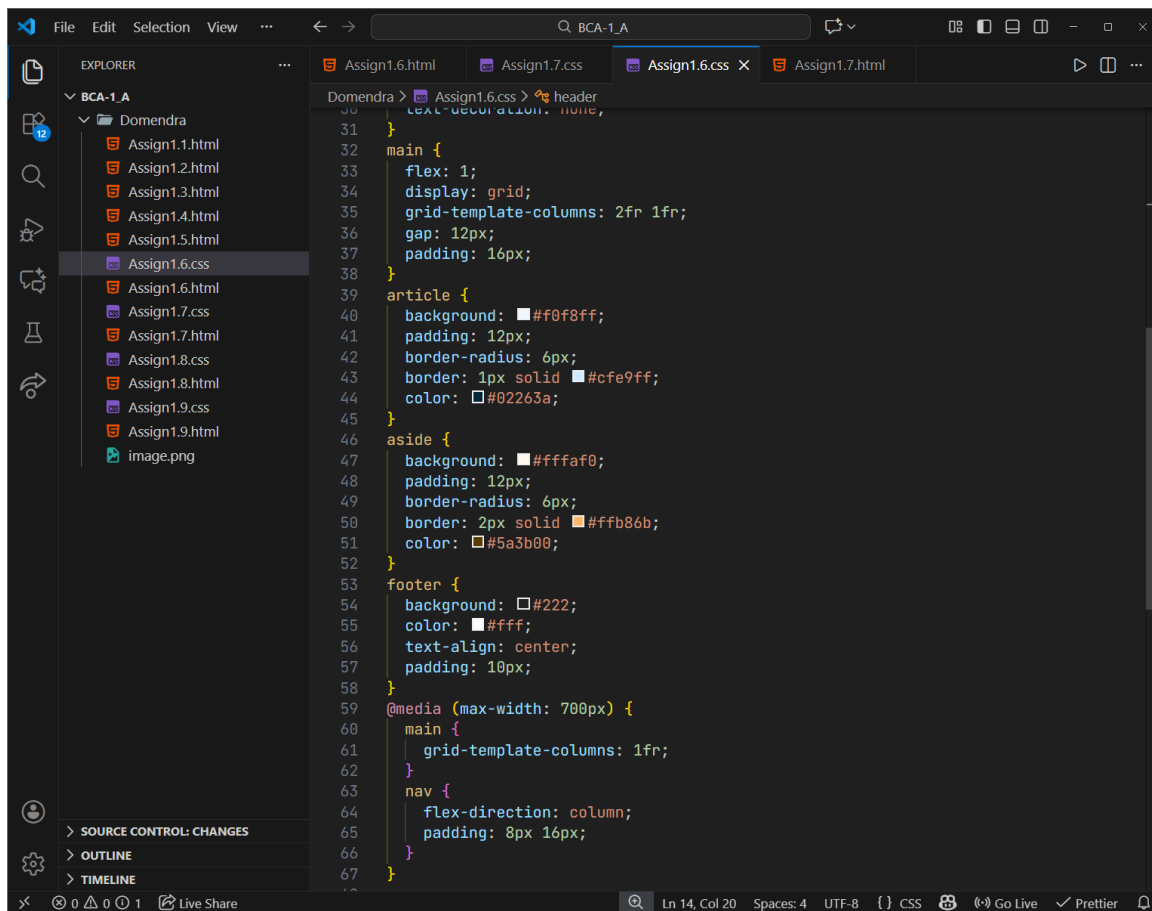
Code:



```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <title>Assign1.6</title>
5   <link rel="stylesheet" href="Assign1.6.css">
6 </head>
7 <body>
8
9   <header><h1>Site</h1></header>
10  <nav><a href="#">Home</a> <a href="#">About</a> <a href="#">Contact</a></nav>
11  <main>
12    <article><h2>Main</h2><p>Primary content.</p></article>
13    <aside><h3>Sidebar</h3><p>Notes</p></aside>
14  </main>
15  <footer>© 2025</footer>
16
17 </body>
18 </html>
19
```

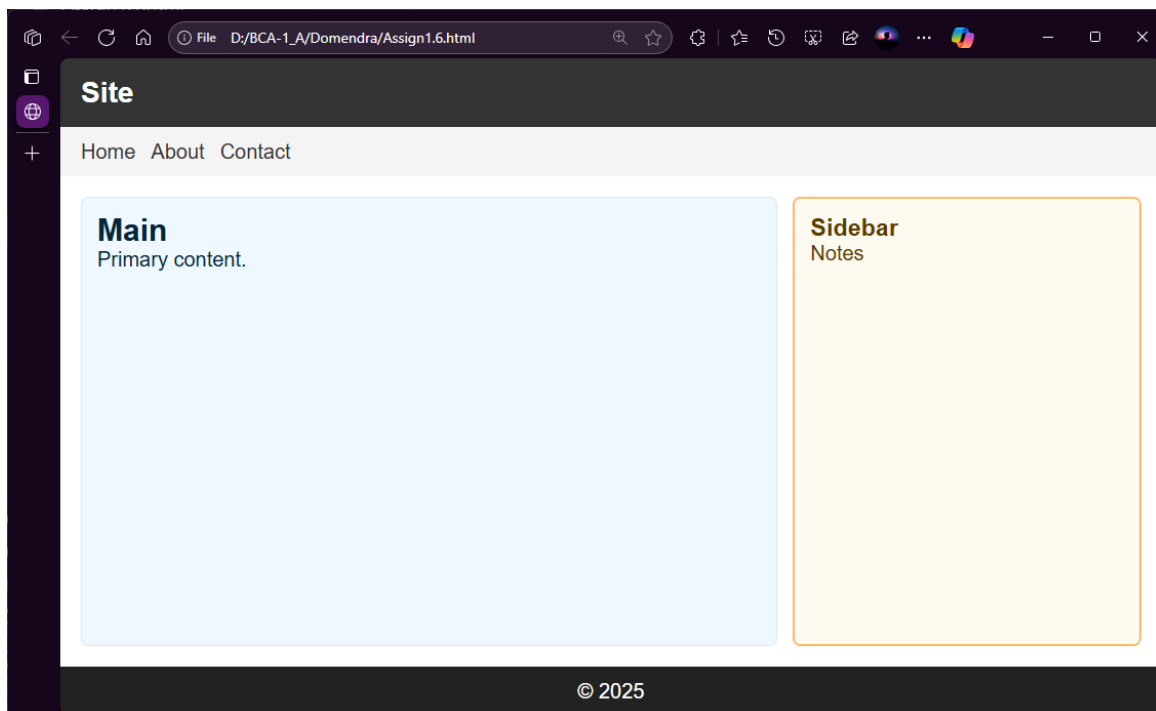


```
1 * {
2   box-sizing: border-box;
3   margin: 0;
4   padding: 0;
5 }
6 body {
7   font-family: Arial;
8   color: #222;
9   min-height: 100vh;
10  display: flex;
11  flex-direction: column;
12 }
13 header {
14   background: #333;
15   color: #fff;
16   padding: 14px 16px;
17 }
18 header h1 {
19   margin: 0;
20   font-size: 1.4rem;
21 }
22 nav {
23   display: flex;
24   gap: 12px;
25   padding: 10px 16px;
26   background: #f4f4f4;
27 }
28 nav a {
29   color: #333;
30   text-decoration: none;
31 }
32 main {
33   flex: 1;
34   display: grid;
35   grid-template-columns: 2fr 1fr;
36   gap: 12px;
37   padding: 16px;
38 }
```



```
30  text-decoration: none;
31  }
32  main {
33    flex: 1;
34    display: grid;
35    grid-template-columns: 2fr 1fr;
36    gap: 12px;
37    padding: 16px;
38  }
39  article {
40    background: #f0f8ff;
41    padding: 12px;
42    border-radius: 6px;
43    border: 1px solid #cfe9ff;
44    color: #02263a;
45  }
46  aside {
47    background: #ffffaf;
48    padding: 12px;
49    border-radius: 6px;
50    border: 2px solid #ffb86b;
51    color: #5a3b00;
52  }
53  footer {
54    background: #222;
55    color: #fff;
56    text-align: center;
57    padding: 10px;
58  }
59  @media (max-width: 700px) {
60    main {
61      grid-template-columns: 1fr;
62    }
63    nav {
64      flex-direction: column;
65      padding: 8px 16px;
66    }
67  }
```

Output:

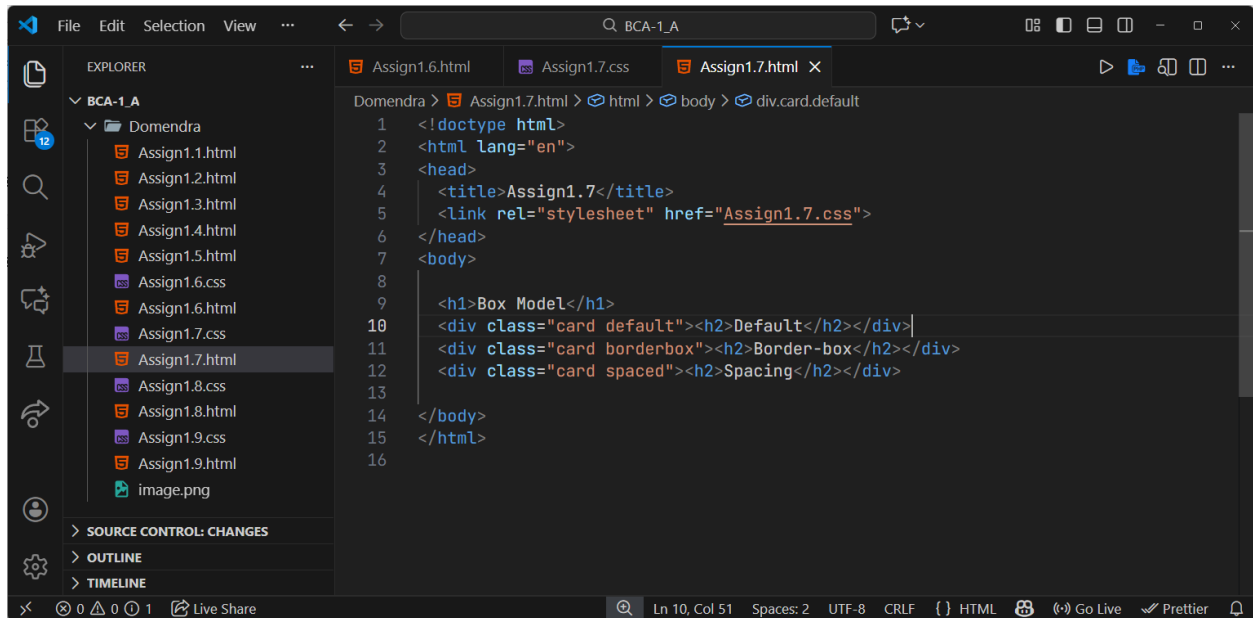


2.2. CSS Box Model & Spacing

Design a card layout using div elements to demonstrate the box model.

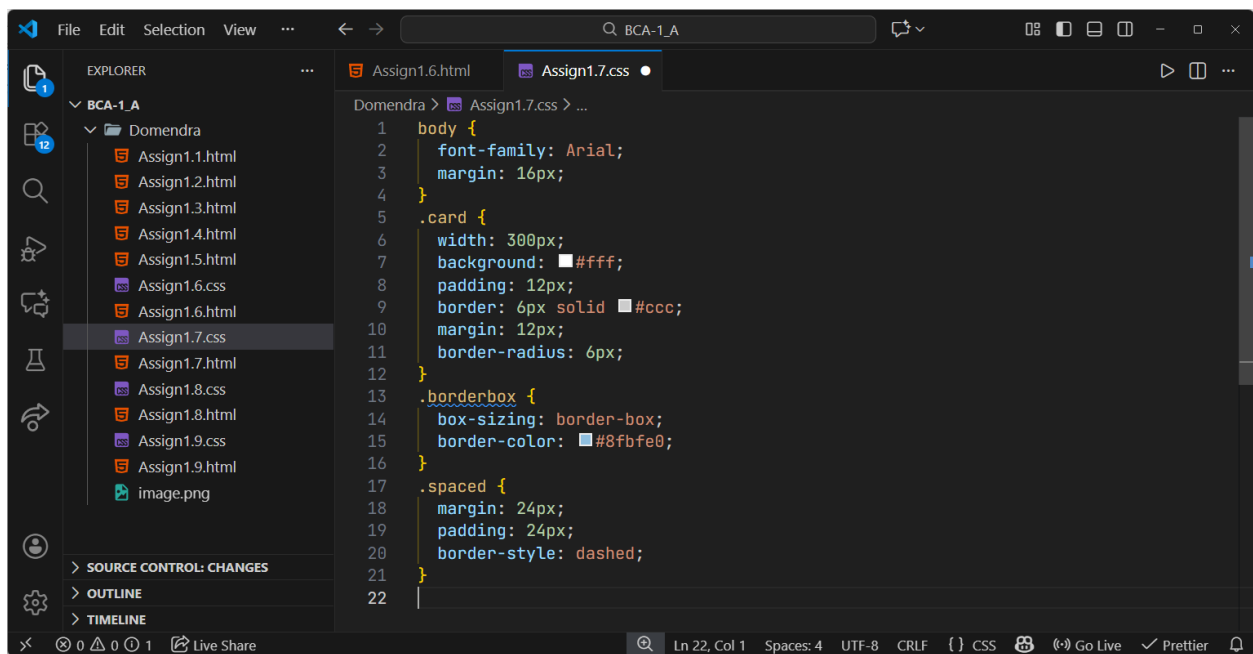
Code:

HTML



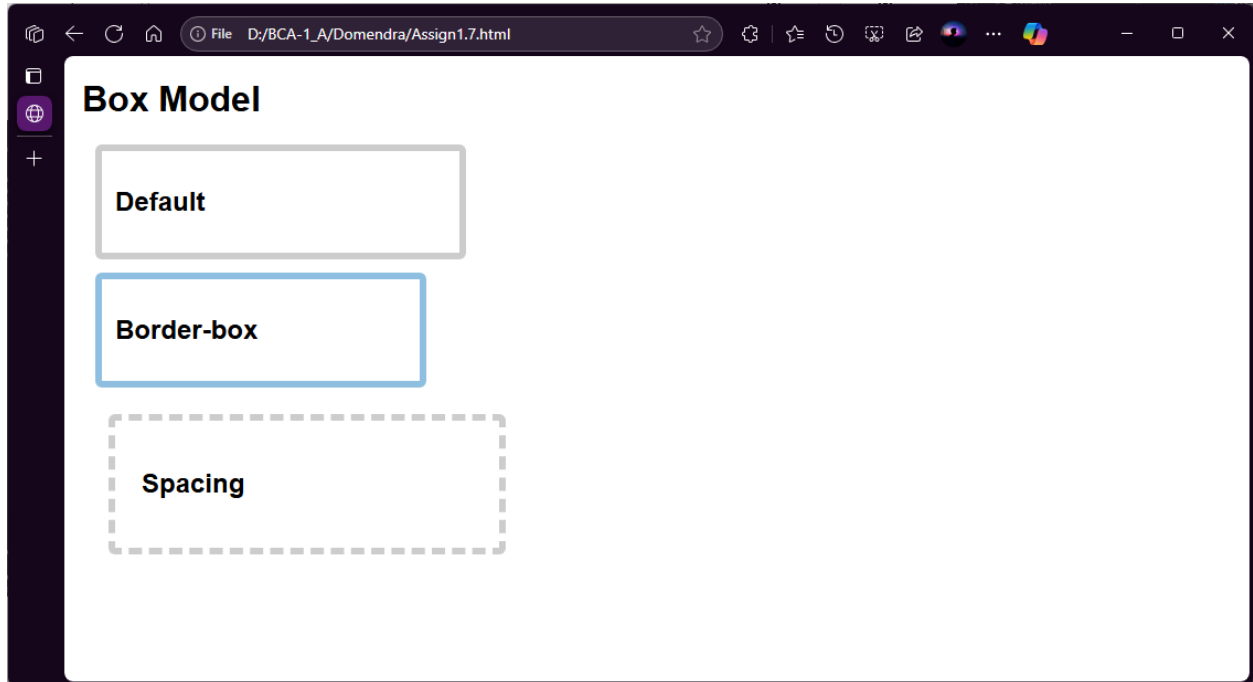
```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <title>Assign1.7</title>
5   <link rel="stylesheet" href="Assign1.7.css">
6 </head>
7 <body>
8
9   <h1>Box Model</h1>
10  <div class="card default"><h2>Default</h2></div>
11  <div class="card borderbox"><h2>Border-box</h2></div>
12  <div class="card spaced"><h2>Spacing</h2></div>
13
14 </body>
15 </html>
16
```

CSS



```
1 body {
2   font-family: Arial;
3   margin: 16px;
4 }
5
6 .card {
7   width: 300px;
8   background: #fff;
9   padding: 12px;
10  border: 6px solid #ccc;
11  margin: 12px;
12  border-radius: 6px;
13 }
14
15 .borderbox {
16   box-sizing: border-box;
17   border-color: #8fbfe0;
18 }
19
20 .spaced {
21   margin: 24px;
22   padding: 24px;
23   border-style: dashed;
24 }
```

Output:



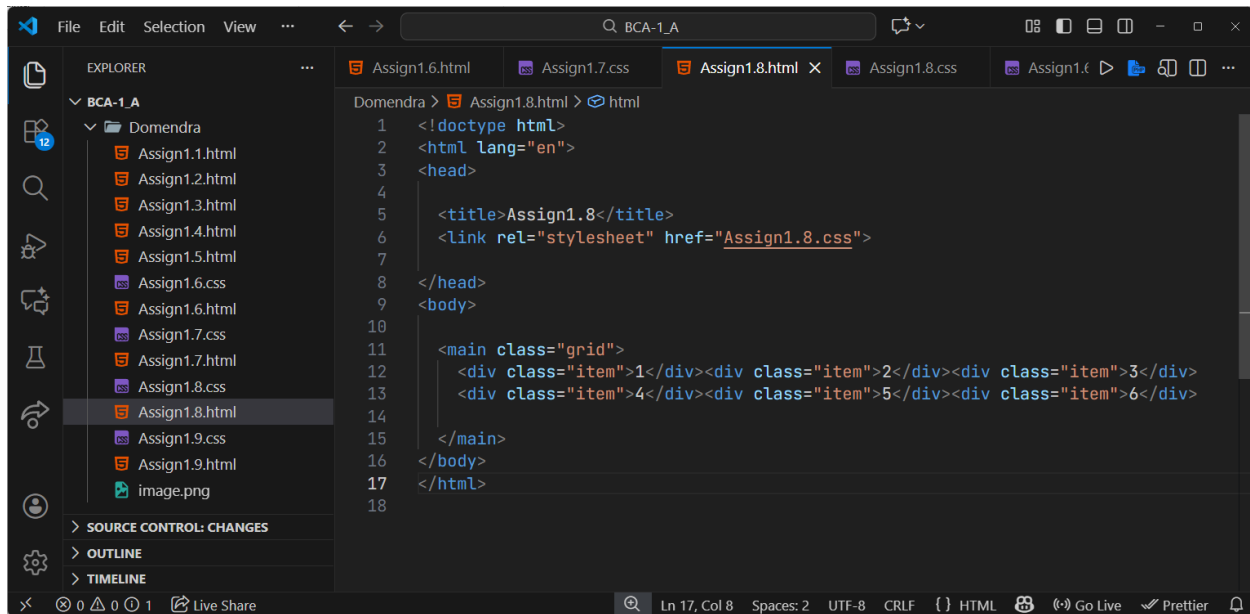
2.3. Create a webpage that adjusts its layout based on screen size using CSS media queries.

Requirements:

- On large screens: Display three columns.
- On tablets: Display two columns.
- On mobile: Display one column.
- Add different background colors for each view.

Code:

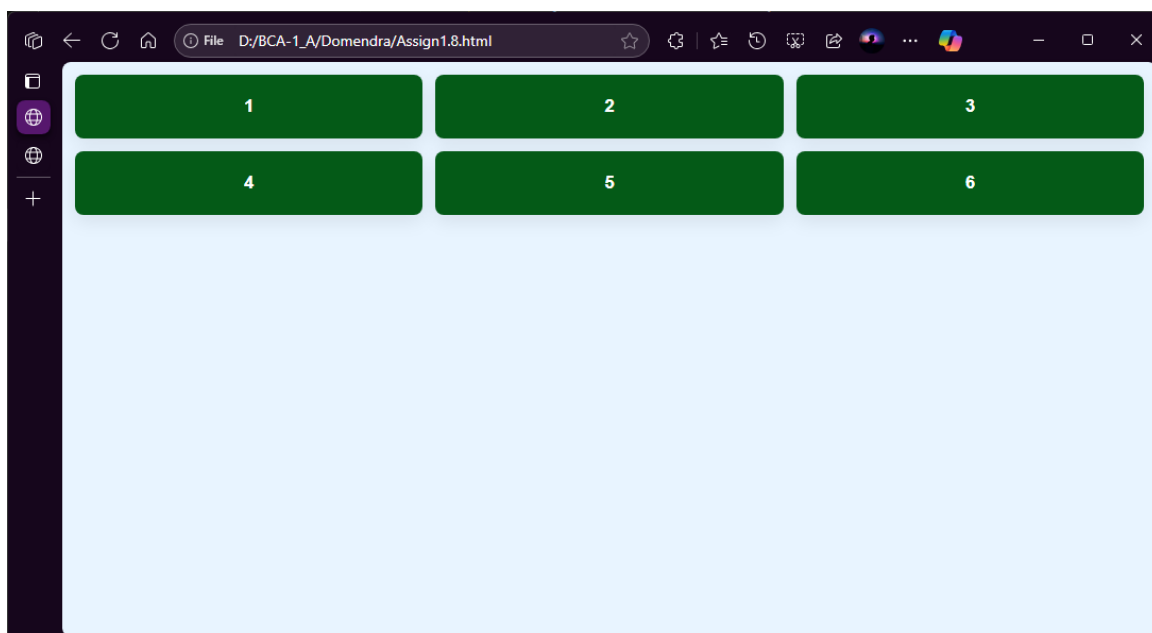
HTML

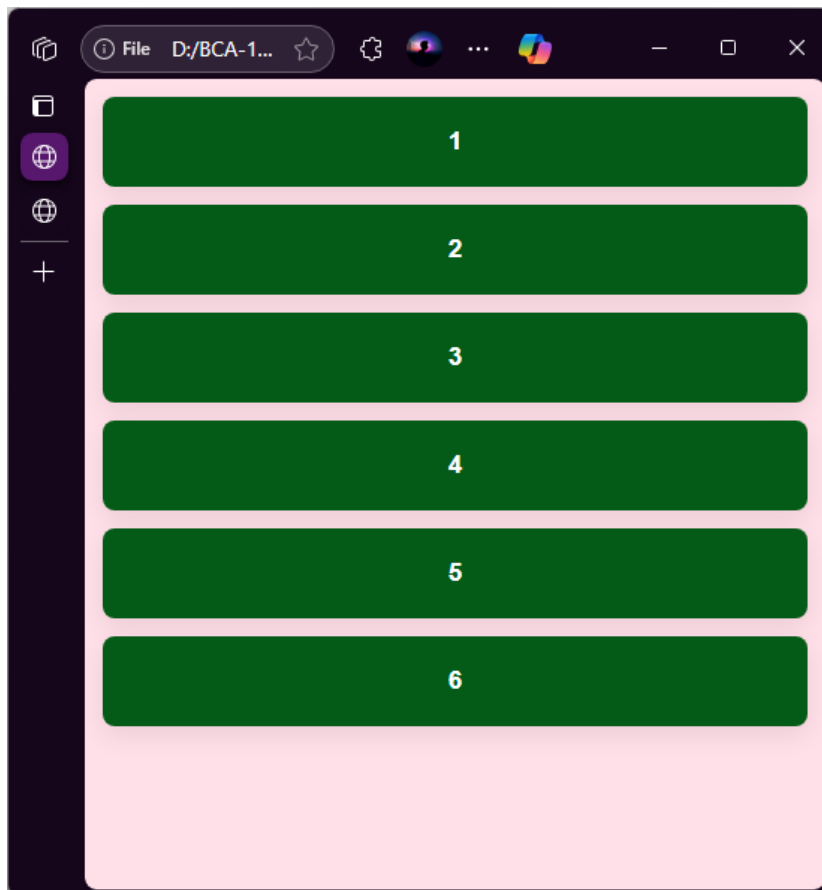
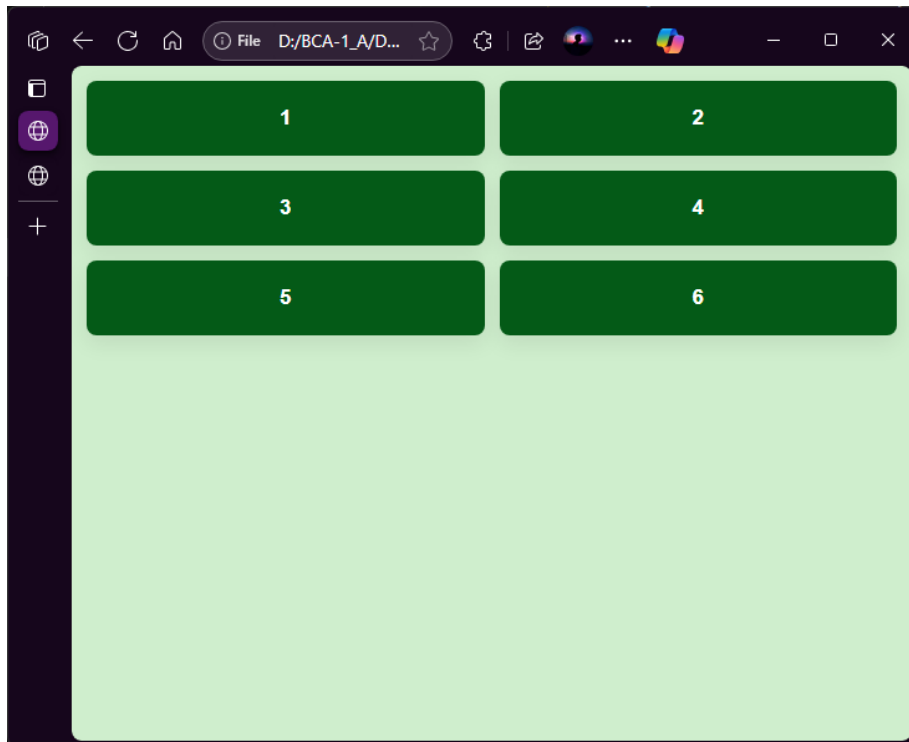


```
1 <!doctype html>
2 <html lang="en">
3 <head>
4
5     <title>Assign1.8</title>
6     <link rel="stylesheet" href="Assign1.8.css">
7
8 </head>
9 <body>
10
11     <main class="grid">
12         <div class="item">1</div><div class="item">2</div><div class="item">3</div>
13         <div class="item">4</div><div class="item">5</div><div class="item">6</div>
14
15     </main>
16 </body>
17 </html>
18
```

```
1 body {
2   font-family: Arial;
3   margin: 0;
4   padding: 12px;
5   background: #e8f4ff;
6   color: #111;
7 }
8 .grid {
9   display: grid;
10  grid-template-columns: repeat(3, 1fr);
11  gap: 12px;
12 }
13 .item {
14   padding: 20px;
15   background: #045a17;
16   color: #fff;
17   border-radius: 8px;
18   text-align: center;
19   font-weight: 700;
20   font-size: 1.05rem;
21   box-shadow: 0 6px 18px rgba(0, 0, 0, 0.08);
22 }
23 @media (max-width: 1024px) {
24   .grid {
25     grid-template-columns: repeat(2, 1fr);
26   }
27   body {
28     background: #cfeecd;
29   }
30 }
31 @media (max-width: 600px) {
32   .grid {
33     grid-template-columns: 1fr;
34   }
35   body {
36     background: #ffdfe8;
37   }
38 }
```

Output:





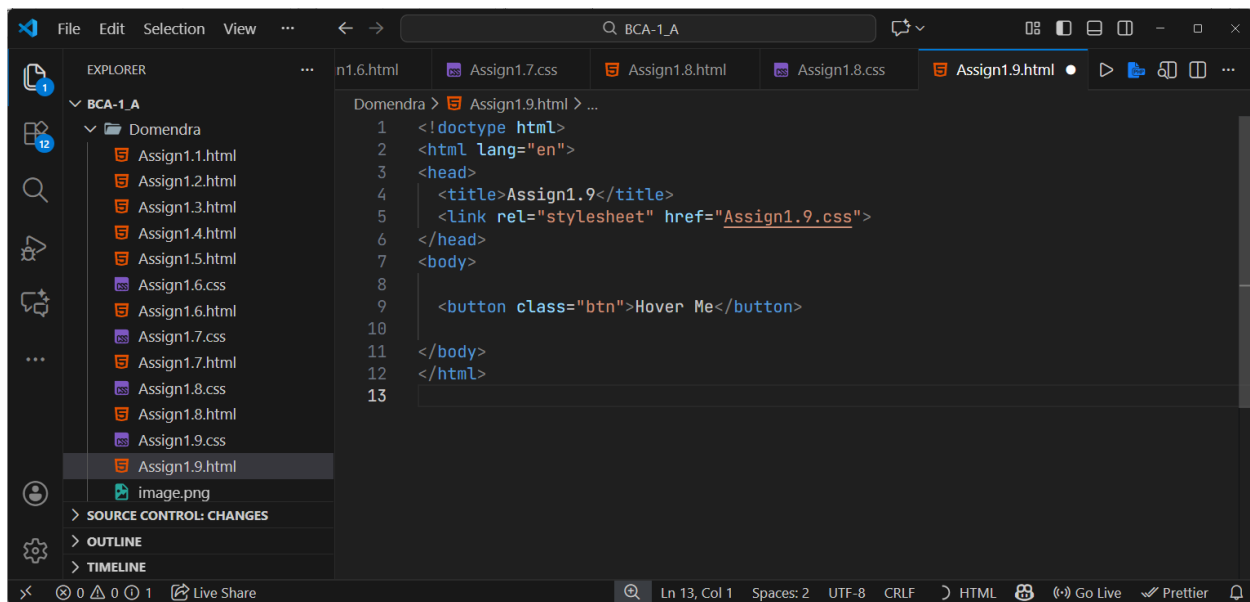
2.4. CSS Animations and Transitions

Create an animated button using CSS.

Requirements: On hover, the button should smoothly change color and increase in size.

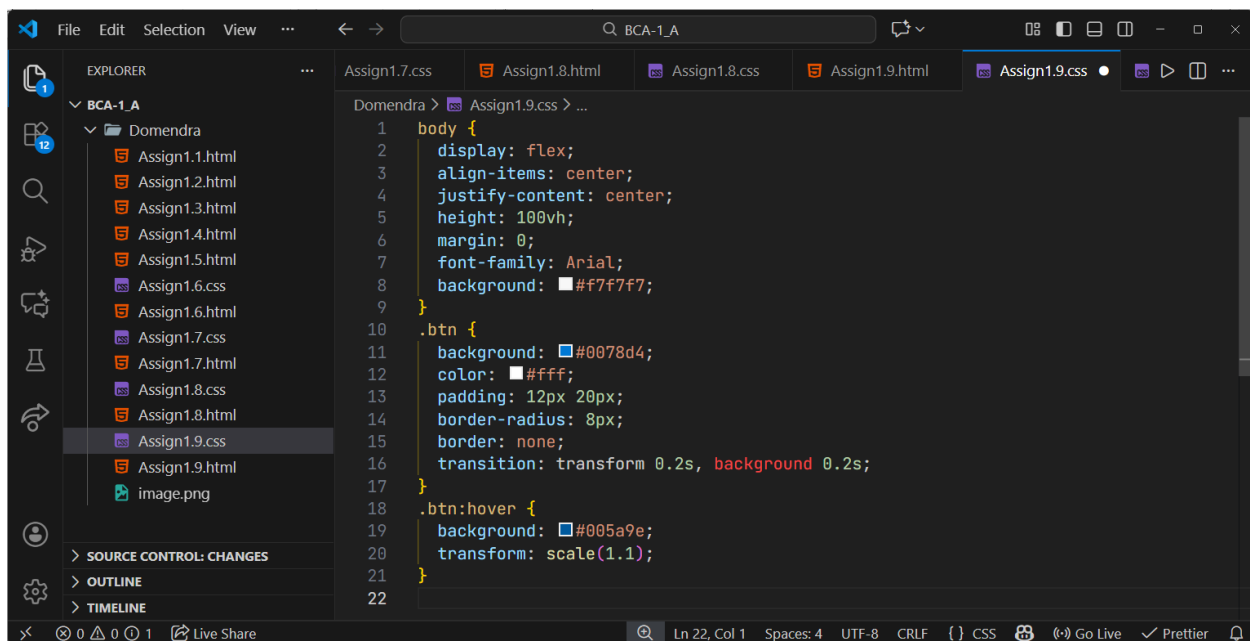
Code:

HTML



```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <title>Assign1.9</title>
5   <link rel="stylesheet" href="Assign1.9.css">
6 </head>
7 <body>
8
9   <button class="btn">Hover Me</button>
10
11 </body>
12 </html>
13
```

CSS



```
1 body {
2   display: flex;
3   align-items: center;
4   justify-content: center;
5   height: 100vh;
6   margin: 0;
7   font-family: Arial;
8   background: #f7f7f7;
9 }
10
11 .btn {
12   background: #0078d4;
13   color: #fff;
14   padding: 12px 20px;
15   border-radius: 8px;
16   border: none;
17   transition: transform 0.2s, background 0.2s;
18 }
19
20 .btn:hover {
21   background: #005a9e;
22   transform: scale(1.1);
23 }
```


Output:

