

# **EXPERIMENT 1**

**Aim:** Building a responsive ui using tailwind CSS.

## **Tools, Technologies, and Prerequisites**

- **Tools/Editor:** VS Code or any modern code editor.
- **Technologies Used:**
  - HTML5 for page structure.
  - Tailwind CSS for styling and responsive design.
  - JavaScript for interactivity.
- **Prerequisites:**
  - Understanding of HTML and CSS basics.
  - Familiarity with JavaScript fundamentals.
  - Knowledge of Tailwind CSS utility classes and responsive design concepts.

## **Theory:**

Tailwind CSS is a **utility-first CSS framework** that provides small, reusable utility classes to apply styling directly in HTML. Instead of relying on predefined themes or writing large amounts of custom CSS, developers use classes like `flex`, `p-4`, `bg-indigo-500`, and `text-center` to build modern and fully customized user interfaces. It is highly **responsive by design**, as it includes breakpoint-based utilities (`sm:`, `md:`, `lg:`, `xl:`) to make layouts adapt smoothly to different devices. This approach helps in creating clean, scalable, and consistent designs efficiently.

The project consists of two pages. The first page displays a **greeting section** with motivational text, a **flip card** showing a daily mission, and navigation to the challenge page. The second page presents a **DSA coding challenge** where the user can input the expected output and receive feedback instantly. Tailwind CSS is used throughout for layout, gradients, typography, spacing, borders, shadows, rounded corners, and animations. The use of Tailwind's responsive utilities ensures that all elements, including text, buttons, and cards, are automatically optimized for different screen sizes.

The project includes features beyond basic UI styling. **Responsiveness** is implemented using Tailwind's breakpoint utilities, making the design adaptive across devices. **Page routing** is demonstrated by linking the greeting page with the challenge page. **JavaScript functionality** is integrated for interactive elements such as flipping the mission card and validating coding challenge answers. Animations like **pulse**, **spin**, **bounce**, and **transitions** are added to enhance the user experience. Together, these functionalities demonstrate how Tailwind CSS can be effectively combined with JavaScript to build an interactive and responsive web project.

## Code:

```
index.html > html > head > style > .preserve-3d
1  index.html<!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>Good Morning</title>
7    <script src="https://cdn.tailwindcss.com"></script>
8    <link href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&display=swap" rel="stylesheet">
9    <style>
10     .perspective {
11       perspective: 1000px;
12     }
13     .preserve-3d {
14       transform-style: preserve-3d;
15     }
16     .backface-hidden {
17       backface-visibility: hidden;
18     }
19     .rotate-y-180 {
20       transform: rotateY(180deg);
21     }
22     .animate-spin-slow {
23       animation: spin 20s linear infinite;
24     }
25     .animate-spin-reverse-slow {
26       animation: spinReverse 25s linear infinite;
27     }
28     @keyframes spin {
29       from { transform: rotate(0deg); }
30       to { transform: rotate(360deg); }
31     }
32     @keyframes spinReverse {
33       from { transform: rotate(360deg); }
34       to { transform: rotate(0deg); }
35     }
36     .flip-card-inner {
37       transition: transform 0.8s;
```

```
C:\Users\Hitika\Desktop\fsd\index.html style > .preserve-3d
2 <html lang="en">
3 <head>
9 <style>
36 .flip-card-inner {
37   transition: transform 0.8s;
38   transform-style: preserve-3d;
39   position: relative;
40   width: 100%;
41   height: 100%;
42 }
43 .flipped {
44   transform: rotateY(180deg);
45 }
46 .flip-card-front, .flip-card-back {
47   position: absolute;
48   width: 100%;
49   height: 100%;
50   backface-visibility: hidden;
51   border-radius: 1rem;
52   box-shadow: 0 10px 15px rgba(0, 0, 0, 0.1);
53 }
54 .flip-card-back {
55   transform: rotateY(180deg);
56 }
57 </style>
58 </head>
59 <body class="bg-gradient-to-r from-blue-100 via-indigo-200 to-purple-200 min-h-screen flex flex-col items-center justify-center">
60 <!-- Header Section -->
61 <div class="text-center z-10">
62 <h1 class="text-6xl font-extrabold text-indigo-800 mb-4 animate-pulse">Good Morning, Everyone! 🌞</h1>
63 <p class="text-lg text-gray-800 mb-4">New day, new lines of code, new chances to build something amazing! 💡🚀
64 <p class="text-md text-indigo-900 italic mb-8 max-w-xl mx-auto">"Don't wait for opportunity. Compile it."</p>
65
66 <!-- Engineering Flip Card with Mission -->
67 <div id="card" class="relative w-96 h-60 perspective mx-auto cursor-pointer">
68 <div id="flipInner" class="flip-card-inner">
69 <!-- Front Side -->
```

```
C:\Users\Hitika\Desktop\fsd\index.html style > .preserve-3d
59 <body class="bg-gradient-to-r from-blue-100 via-indigo-200 to-purple-200 min-h-screen flex flex-col items-center justify-center">
61 <div class="text-center z-10">
62 <div id="card" class="relative w-96 h-60 perspective mx-auto cursor-pointer">
63 <div id="flipInner" class="flip-card-inner">
76 <div class="flip-card-back bg-gradient-to-br from-indigo-100 to-purple-200 border border-indigo-400 flex flex-col items-center justify-center">
77 <p class="text-center text-gray-800 text-lg font-medium">
78 🌐 Visit a random GitHub repo. Identify one issue. Try to solve it. Welcome to Open Source!
79 </p>
80 <button onclick="window.open('https://github.com/explore', '_blank'); event.stopPropagation();" class="mt-4">
81 </div>
82 </div>
83 </div>
84
85 <script>
86 const card = document.getElementById('card');
87 const flipInner = document.getElementById('flipInner');
88
89 card.addEventListener('click', () => {
90   flipInner.classList.toggle('flipped');
91 });
92 </script>
93
94 <!-- Puzzle Button -->
95 <button onclick="window.location.href='page2.html'" class="mt-10 bg-purple-600 hover:bg-purple-700 text-white">
96 </div>
97
98 <!-- Animated Background Elements -->
99 <div class="absolute top-10 left-10 text-6xl text-indigo-300 animate-spin-slow">⚙️</div>
100 <div class="absolute bottom-10 right-10 text-6xl text-indigo-300 animate-spin-reverse-slow">⚙️</div>
101 <div class="absolute top-1/3 left-1/4 text-3xl text-purple-300 animate-bounce">💡</div>
102 <div class="absolute bottom-1/4 right-1/3 text-3xl text-blue-300 animate-bounce">🚀</div>
103 </body>
104 </html>
```

```

page2.html > html > body.bg-gradient-to-br from-yellow-100 via-pink-100 to-purple-100.min-h-screen.flex.flex-col.items-center.justify-center.font-[Orbitron]
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>🔥 DSA Challenge</title>
7    <script src="https://cdn.tailwindcss.com"></script>
8    <link href="https://fonts.googleapis.com/css2?family=Orbitron:wght@500&display=swap" rel="stylesheet">
9  </head>
10 <body class="bg-gradient-to-br from-yellow-100 via-pink-100 to-purple-100 min-h-screen flex flex-col items-center justify-center font-[Orbitron]">
11
12   <h1 class="text-4xl font-bold text-purple-800 mb-4 animate-pulse">🔥 Think Like a Dev</h1>
13
14   <div class="bg-white border border-purple-300 shadow-xl rounded-xl p-6 w-full max-w-md">
15     <h2 class="text-xl font-semibold text-purple-700 mb-2">🔥 Can You Solve This?</h2>
16     <pre class="bg-gray-100 text-left p-3 rounded text-sm mb-4">
17 int arr[] = {2, 3, 4, 6, 8};
18 int x = arr[0];
19 for (int i = 1; i < 5; i++) {
20   x ^= arr[i];
21 }
22 cout << x;
23 </pre>
24     <input id="answer" type="text" placeholder="Enter output..." class="w-full p-2 border border-gray-300 rounded mb-2">
25     <button onclick="checkAnswer()" class="bg-purple-600 hover:bg-purple-700 text-white px-4 py-2 rounded w-full">Check Answer</button>
26     <p id="result" class="mt-4 text-lg font-semibold"></p>
27     <button onclick="next()" class="text-sm text-purple-600 mt-4 underline hover:text-purple-800">🔄 Try another (soon)</button>
28   </div>
29
30   <a href="index.html" class="mt-8 inline-block text-purple-700 underline">🏠 Back to Debug Page</a>
31
32   <script>
33     function checkAnswer() {
34       const ans = document.getElementById("answer").value.trim();
35       const result = document.getElementById("result");
36       if (ans === "1") {
37         result.textContent = "✅ Correct! XOR can be powerful.";

```

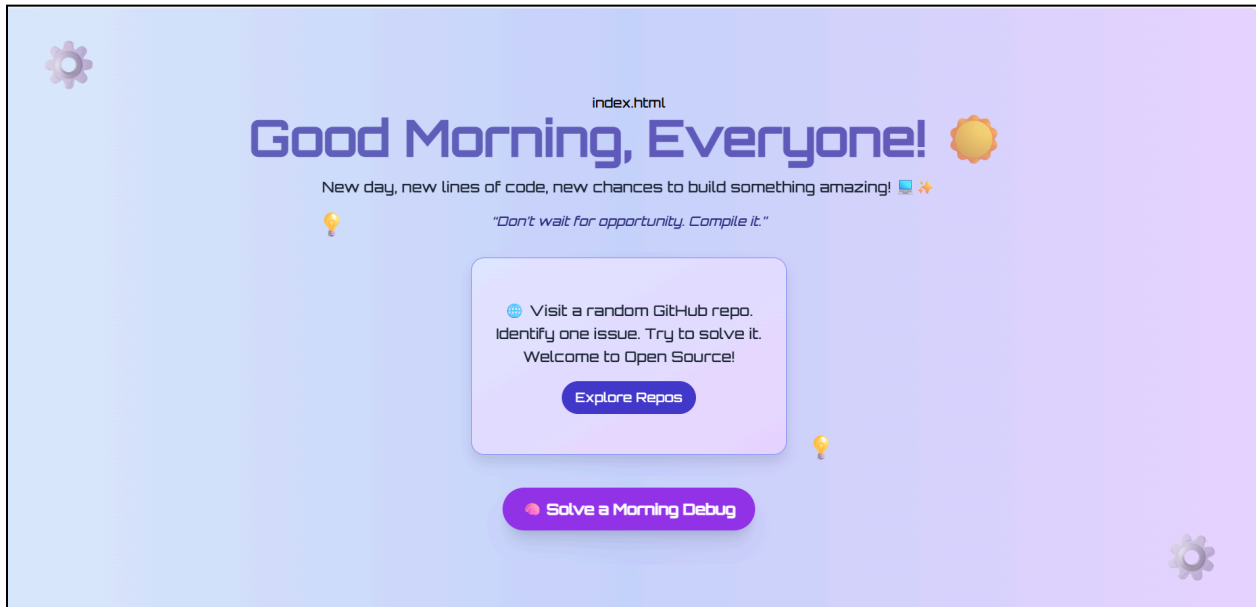
```

</script>
    function checkAnswer() {
      const ans = document.getElementById("answer").value.trim();
      const result = document.getElementById("result");
      if (ans === "1") {
        result.textContent = "✅ Correct! XOR can be powerful.";
        result.className = "text-green-600 mt-4 text-lg font-semibold";
      } else if (ans.length > 0) {
        result.textContent = "❌ Nope. Try understanding XOR on bits.";
        result.className = "text-red-600 mt-4 text-lg font-semibold";
      } else {
        result.textContent = "Please enter something.";
        result.className = "text-yellow-600 mt-4 text-lg font-semibold";
      }
    }

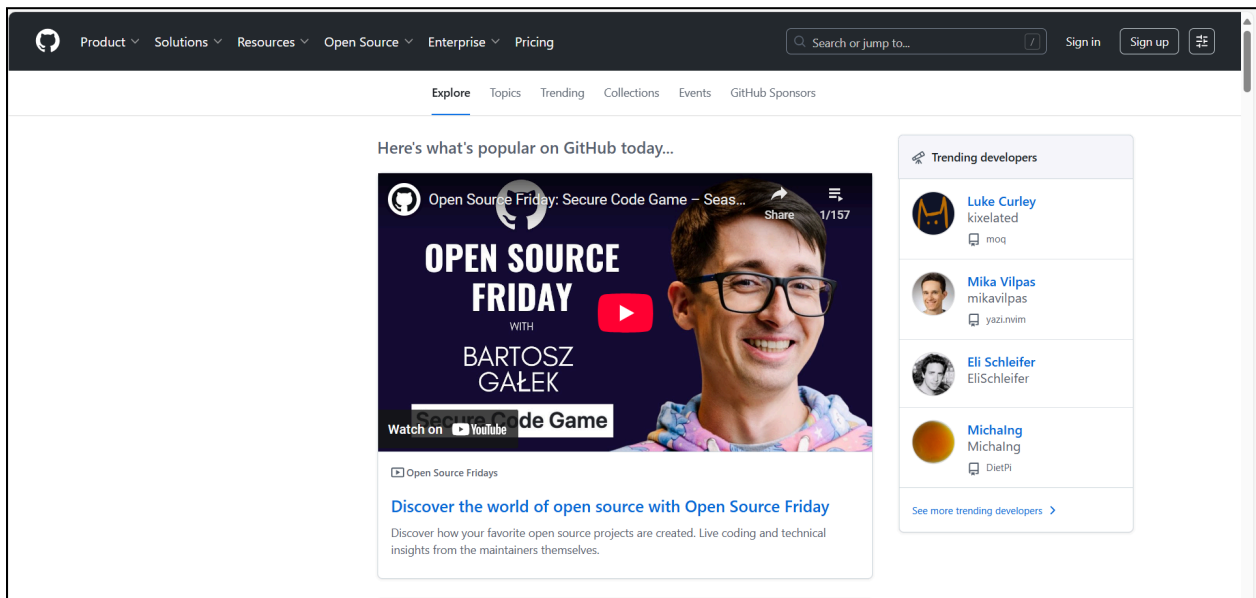
    function next() {
      alert("More challenges coming soon... Stay tuned! 🤖");
    }
  </script>
</body>
</html>

```

## Output:



a. Greeting page



b. Github navigation



c. DSA coding challenge pages

## Conclusion:

The experiment demonstrates that Tailwind CSS is an efficient framework for developing modern, responsive, and visually appealing user interfaces. By utilizing its utility-first classes, layouts and styles can be built quickly without extensive custom CSS. The project highlights how Tailwind's responsive utilities, animations, and design flexibility can be combined with JavaScript to create interactive and engaging web

pages. Overall, this approach results in a clean, scalable, and device-friendly UI that meets the objectives of building a responsive frontend.