FULL STACK -1

AIM: Create a student registration form with HTML5 validation for name, email, and age inputs. Medium: Build a responsive banking UI with deposit/withdrawal buttons and balance display. Hard: Develop an admin dashboard with CSS Grid layout and dynamic theme switching.

1.Experiment-1

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Student Registration Form</title>
</head>
<body>
  <h2>Student Registration</h2>
  <form>
    <label for="name">Name:</label>
    <input type="text" id="name" name="name" required pattern="[A-Za-z ]+" /><br><br>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email" required /><br><br>

    <label for="age">Age:</label>
    <input type="number" id="age" name="age" required min="5" max="100" /><br><br>

    <button type="submit">Register</button>
  </form>
</body>
</html>
```

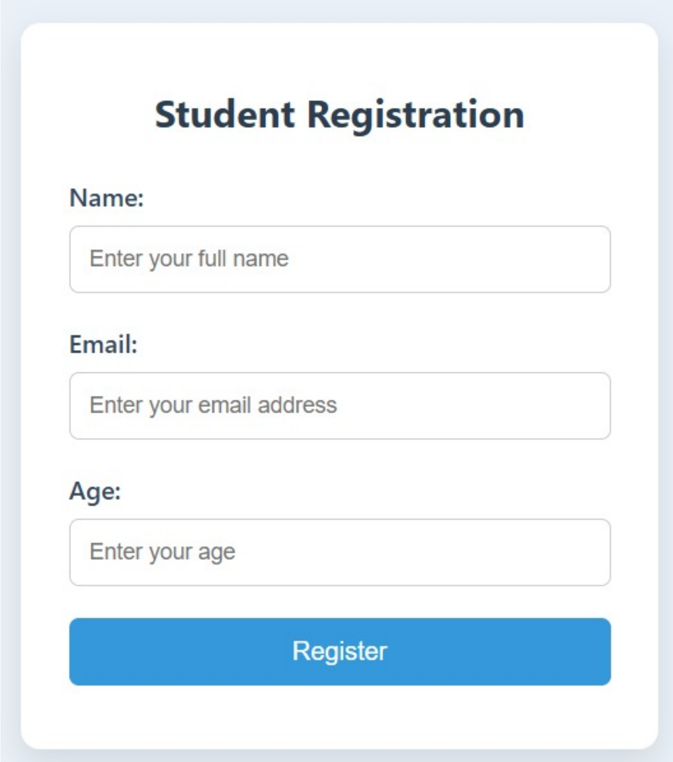


Figure 1: Experiment 1 Student registration Output

2.Experiment 2

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Banking UI</title>
  <link href="https://fonts.googleapis.com/css2?family=Poppins&display=swap" rel="stylesheet">
  <style>
    body {
      font-family: 'Poppins', sans-serif;
      background: #e6f0ff;
      margin: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }
```

```
 .bank-container {
   background: white;
   padding: 30px;
   border-radius: 12px;
   box-shadow: 0 4px 20px rgba(0, 0, 0, 0.1);
   text-align: center;
   max-width: 400px;
   width: 100%;
 }

 h2 {
   margin-bottom: 20px;
 }

 .balance {
   font-size: 2.5rem;
   font-weight: bold;
   color: #333;
   margin-bottom: 30px;
 }

 .actions button {
   padding: 12px 20px;
   font-size: 1rem;
   margin: 5px;
   border: none;
   border-radius: 8px;
   cursor: pointer;
   transition: 0.2s ease;
 }

 .actions button:nth-child(1) {
   background: #28a745;
   color: white;
 }

 .actions button:nth-child(2) {
   background: #dc3545;
   color: white;
 }
.actions button:hover {
      opacity: 0.9;
    }
  </style>
</head>
<body>
  <div class="bank-container">
    <h2>My Bank Account</h2>
    <div class="balance">$<span id="balance">1000</span></
div>
    <div class="actions">
      <button onclick="deposit()">Deposit $100</button>
      <button onclick="withdraw()">Withdraw $100</button>
    </div>
  </div>

  <script>
    let balance = 1000;
    const balanceDisplay =
document.getElementById("balance");

    function deposit() {
      balance += 100;
      balanceDisplay.textContent = balance;
    }

    function withdraw() {
      if (balance >= 100) {
        balance -= 100;
        balanceDisplay.textContent = balance;
      } else {
        alert("Insufficient funds!");
      }
    }
  </script>
</body>
</html>
```

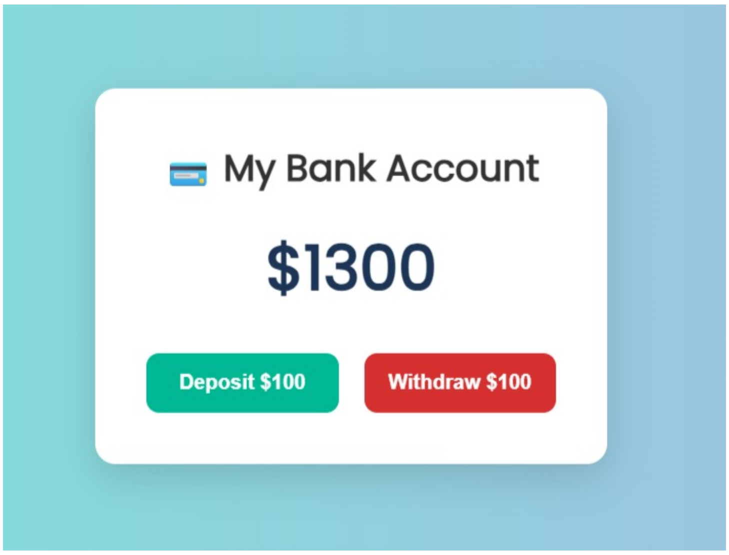


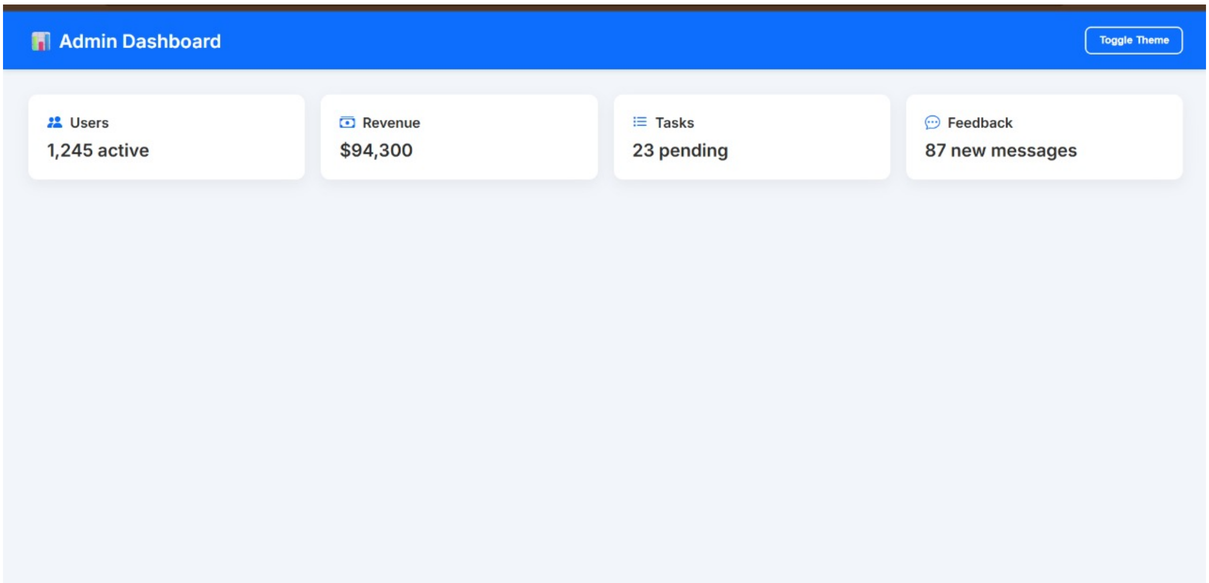
Figure 2: Experiment 2 Banking UI output

3.Experiment 3

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1.0"/>
  <title>Admin Dashboard</title>
  <link href="https://fonts.googleapis.com/css2?family=Inter:wght@400;600&display=swap" rel="stylesheet">
  <style>
    :root {
      --bg: #f9f9f9;
      --text: #222;
      --card: #fff;
      --primary: #007bff;
    }

    [data-theme="dark"] {
      --bg: #121212;
      --text: #f0f0f0;
      --card: #1e1e1e;
      --primary: #4e9eff;
```

```css
}

body {
  margin: 0;
  background-color: var(--bg);
  color: var(--text);
  font-family: 'Inter', sans-serif;
}

header {
  background-color: var(--primary);
  color: white;
  padding: 1em 2em;
  display: flex;
  justify-content: space-between;
  align-items: center;
}

.dashboard {
  display: grid;
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));
  gap: 20px;
  padding: 2em;
}

.card {
  background: var(--card);
  padding: 20px;
  border-radius: 12px;
  box-shadow: 0 4px 12px rgba(0,0,0,0.1);
  transition: background 0.3s;
}

.card h3 {
  margin: 0 0 10px;
}

.toggle-theme {
  background: transparent;
  border: 2px solid white;
  color: white;
```

```
      padding: 6px 12px;
      border-radius: 8px;
      cursor: pointer;
      font-weight: 600;
    }

    .toggle-theme:hover {
      background: rgba(255, 255, 255, 0.2);
    }
  </style>
</head>
<body data-theme="light">
  <header>
    <h1>Admin Dashboard</h1>
    <button class="toggle-theme" onclick="toggleTheme()">Toggle
Theme</button>
  </header>

  <main class="dashboard">
    <div class="card">
      <h3>Users</h3>
      <p>1,245 active</p>
    </div>
    <div class="card">
      <h3>Revenue</h3>
      <p>$94,300</p>
    </div>
    <div class="card">
      <h3>Tasks</h3>
      <p>23 pending</p>
    </div>
    <div class="card">
      <h3>Feedback</h3>
      <p>87 new messages</p>
    </div>
  </main>

  <script>
    function toggleTheme() {
      const current = document.body.getAttribute('data-theme');
      document.body.setAttribute('data-theme', current === 'dark' ?
'light' : 'dark');
    }
  </script>
</body>
</html>
```



Admin Dashboard

Users
1,245 active

Revenue
$94,300

Tasks
23 pending

Feedback
87 new messages

Figure 3: Experiment 3 Admin Dashboard Output