

EXP 1: Create a student registration form with HTML5 validation for name, email, and age inputs.

Code:

```
<!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>

  <!-- Tailwind CSS CDN -->

  <script src="https://cdn.tailwindcss.com"></script>

  <!-- Google Fonts: Inter -->

  <link rel="preconnect" href="https://fonts.googleapis.com">

  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700&display=swap" rel="stylesheet">

  <style>

    /* Custom styles can be added here if needed */

    body {

      font-family: 'Inter', sans-serif;

    }

    /* Style for browser validation pop-ups */

    input:invalid {

      border-color: #e53e3e; /* red-600 */

    }

    input:valid {

      border-color: #28a745; /* green-600 */

    }

  </style>

</head>

<body>
```

```

</style>
</head>
<body class="bg-gray-100 flex items-center justify-center min-h-screen">

  <!-- Main container for the form -->

  <div class="w-full max-w-md p-8 space-y-6 bg-white rounded-xl shadow-lg">

    <h1 class="text-3xl font-bold text-center text-gray-800">Student Registration</h1>

    <p class="text-center text-gray-500">Please fill out the form below to register.</p>


    <!-- The novalidate attribute prevents the browser's default validation tooltips,
         but the :invalid and :valid pseudo-classes still work.

         Remove 'novalidate' if you want the default browser pop-ups. -->

    <form action="#" method="POST" class="space-y-6">

      <!-- Name Input -->

      <div>

        <label for="name" class="text-sm font-medium text-gray-700">Full Name</label>

        <input

          type="text"

          id="name"

          name="name"

          placeholder="e.g., Jane Doe"

          required

          class="mt-1 block w-full px-4 py-3 bg-gray-50 border border-gray-300 rounded-lg
shadow-sm placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-indigo-500
focus:border-indigo-500 transition duration-150 ease-in-out"

          >

        <p class="mt-1 text-xs text-red-500 hidden peer-invalid:block">

          Full name is required.

        </p>

      </div>

```

```
<!-- Email Input -->

<div>

  <label for="email" class="text-sm font-medium text-gray-700">Email
Address</label>

  <input
    type="email"
    id="email"
    name="email"
    placeholder="you@example.com"
    required

    class="mt-1 block w-full px-4 py-3 bg-gray-50 border border-gray-300 rounded-lg
shadow-sm placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-indigo-500
focus:border-indigo-500 transition duration-150 ease-in-out"

  >

  <p class="mt-1 text-xs text-red-500 hidden peer-invalid:block">

    Please provide a valid email address.

  </p>
</div>
```

```
<!-- Age Input -->

<div>

  <label for="age" class="text-sm font-medium text-gray-700">Age</label>

  <input
    type="number"
    id="age"
    name="age"
    placeholder="Enter your age"
    required
```

```
        min="16"

        max="100"

        class="mt-1 block w-full px-4 py-3 bg-gray-50 border border-gray-300 rounded-lg
shadow-sm placeholder-gray-400 focus:outline-none focus:ring-2 focus:ring-indigo-500
focus:border-indigo-500 transition duration-150 ease-in-out"

    >

    <p class="mt-1 text-xs text-red-500 hidden peer-invalid:block">

        Age must be between 16 and 100.

    </p>

</div>


<!-- Submit Button -->

<div>

    <button

        type="submit"

        class="w-full flex justify-center py-3 px-4 border border-transparent rounded-lg
shadow-sm text-sm font-medium text-white bg-indigo-600 hover:bg-indigo-700
focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-indigo-500 transition duration-
150 ease-in-out"

    >

        Register

    </button>

</div>

</form>

</div>

</body>

</html>
```

EXP 2: Build a responsive banking UI with deposit/withdrawal buttons and balance display.

Code :

```
<!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>

  <!-- Tailwind CSS CDN -->

  <script src="https://cdn.tailwindcss.com"></script>

  <!-- Google Fonts: Inter -->

  <link rel="preconnect" href="https://fonts.googleapis.com">

  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700&display=swap" rel="stylesheet">

  <!-- Icon Library -->

  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/font-awesome@6.1.1/css/all.min.css">

  <style>

    body {

      font-family: 'Inter', sans-serif;

    }

  </style>

</head>

<body class="bg-gray-100 flex items-center justify-center min-h-screen">

  <!-- Main Banking App Container -->

  <div class="w-full max-w-sm mx-auto bg-white rounded-2xl shadow-2xl p-6 space-y-6">
```

```
<!-- Header -->

<div class="flex justify-between items-center">

  <h1 class="text-xl font-bold text-gray-800">My Account</h1>

  <i class="fa-solid fa-bell text-gray-400"></i>

</div>


<!-- Balance Display -->

<div class="bg-indigo-600 text-white p-6 rounded-xl">

  <p class="text-sm text-indigo-200">Current Balance</p>

  <p id="balance-display" class="text-4xl font-bold mt-1">$1,250.75</p>

</div>


<!-- Transaction Controls -->

<div class="space-y-4">

  <h2 class="text-lg font-semibold text-gray-700">New Transaction</h2>

  <!-- Amount Input -->

  <div>

    <label for="amount" class="sr-only">Amount</label>

    <div class="relative">

      <div class="pointer-events-none absolute inset-y-0 left-0 pl-3 flex items-center">

        <span class="text-gray-500">$</span>

      </div>

      <input

        type="number"

        id="amount"

        name="amount"

        placeholder="0.00"

      >

    </div>

  </div>

</div>
```

```
        class="block w-full pl-7 pr-12 py-3 bg-gray-50 border border-gray-300 rounded-
lg focus:outline-none focus:ring-2 focus:ring-indigo-500 transition"
```

```
        >
```

```
    </div>
```

```
</div>
```

```
<!-- Action Buttons -->
```

```
<div class="grid grid-cols-2 gap-4">
```

```
    <button id="deposit-btn" class="w-full flex items-center justify-center gap-2 py-3
px-4 border border-transparent rounded-lg text-sm font-medium text-white bg-green-500
hover:bg-green-600 focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-green-500
transition">
```

```
        <i class="fa-solid fa-arrow-down"></i>
```

```
        Deposit
```

```
    </button>
```

```
    <button id="withdraw-btn" class="w-full flex items-center justify-center gap-2 py-3
px-4 border border-transparent rounded-lg text-sm font-medium text-white bg-red-500
hover:bg-red-600 focus:outline-none focus:ring-2 focus:ring-offset-2 focus:ring-red-500
transition">
```

```
        <i class="fa-solid fa-arrow-up"></i>
```

```
        Withdraw
```

```
    </button>
```

```
</div>
```

```
</div>
```

```
<!-- Message Area -->
```

```
<div id="message-box" class="text-center text-sm font-medium text-gray-600 h-
5"></div>
```

```
</div>
```

```
<script>
```

```
// --- DOM Elements ---
```

```
const balanceDisplay = document.getElementById('balance-display');
```

```
const amountInput = document.getElementById('amount');
```

```
const depositBtn = document.getElementById('deposit-btn');
```

```
const withdrawBtn = document.getElementById('withdraw-btn');
```

```
const messageBox = document.getElementById('message-box');
```

```
// --- State ---
```

```
let currentBalance = 1250.75;
```

```
// --- Functions ---
```

```
/**
```

```
 * Updates the balance display with proper formatting.
```

```
*/
```

```
function updateDisplay() {
```

```
    balanceDisplay.textContent = `$$${currentBalance.toLocaleString('en-US', {  
minimumFractionDigits: 2, maximumFractionDigits: 2 }}}`;
```

```
}
```

```
/**
```

```
 * Displays a message to the user for a short duration.
```

```
 * @param {string} text - The message to display.
```

```
 * @param {string} color - The color of the message (e.g., 'text-red-500').
```

```
*/
```

```
function showMessage(text, color = 'text-gray-600') {
```



```

    messageBox.textContent = text;

    messageBox.className = `text-center text-sm font-medium h-5 ${color}`;

    setTimeout(() => {
        messageBox.textContent = "";
    }, 3000); // Message disappears after 3 seconds
}

/**
 * Handles the deposit logic.
 */
function handleDeposit() {
    const amount = parseFloat(amountInput.value);

    if (isNaN(amount) || amount <= 0) {
        showMessage('Please enter a valid positive amount.', 'text-red-500');

        return;
    }

    currentBalance += amount;

    updateDisplay();

    showMessage(`Successfully deposited $$${amount.toFixed(2)}.`, 'text-green-500');

    amountInput.value = "";
}

/**
 * Handles the withdrawal logic.
 */
function handleWithdraw() {
    const amount = parseFloat(amountInput.value);

    if (isNaN(amount) || amount <= 0) {

```

```
        showMessage('Please enter a valid positive amount.', 'text-red-500');
        return;
    }
    if (amount > currentBalance) {
        showMessage('Insufficient funds for this withdrawal.', 'text-red-500');
        return;
    }
    currentBalance -= amount;
    updateDisplay();
    showMessage(`Successfully withdrew $$${amount.toFixed(2)}.`, 'text-blue-500');
    amountInput.value = '';
}

// --- Event Listeners ---
depositBtn.addEventListener('click', handleDeposit);
withdrawBtn.addEventListener('click', handleWithdraw);

// --- Initial Display Update ---
// This ensures the initial balance is correctly formatted on page load.
updateDisplay();
</script>
</body>
</html>
```

EXP 3: Develop an admin dashboard with CSS Grid layout and dynamic theme switching.

Code:

```
<!DOCTYPE html>

<html lang="en" class=""> <!-- The class on the HTML element will be toggled for themes -->

<head>

  <meta charset="UTF-8">

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Admin Dashboard</title>

  <!-- Tailwind CSS CDN -->

  <script src="https://cdn.tailwindcss.com"></script>

  <!-- Google Fonts: Inter -->

  <link rel="preconnect" href="https://fonts.googleapis.com">

  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

  <link
href="https://fonts.googleapis.com/css2?family=Inter:wght@400;500;600;700&display=swap" rel="stylesheet">

  <!-- Icon Library -->

  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/@fortawesome/fontawesome-free@6.1.1/css/all.min.css">

  <style>

    /* Custom CSS for layout and theming */

    body {

      font-family: 'Inter', sans-serif;

    }

    /* CSS Variables for theming. Default is light mode. */
```

```
:root {  
  --bg-primary: #ffffff;  
  --bg-secondary: #f3f4f6; /* gray-100 */  
  --text-primary: #1f2937; /* gray-800 */  
  --text-secondary: #6b7280; /* gray-500 */  
  --border-color: #e5e7eb; /* gray-200 */  
  --accent-color: #4f46e5; /* indigo-600 */  
}
```

/\* Dark theme variables, applied when the 'dark' class is on the html element \*/

```
html.dark {  
  --bg-primary: #1f2937; /* gray-800 */  
  --bg-secondary: #374151; /* gray-700 */  
  --text-primary: #f9fafb; /* gray-50 */  
  --text-secondary: #d1d5db; /* gray-300 */  
  --border-color: #4b5563; /* gray-600 */  
  --accent-color: #818cf8; /* indigo-400 */  
}
```

/\* Applying variables to elements \*/

```
body {  
  background-color: var(--bg-secondary);  
  color: var(--text-primary);  
  transition: background-color 0.3s, color 0.3s;  
}
```

```
.dashboard-grid {  
  display: grid;
```

```
grid-template-columns: 250px 1fr; /* Sidebar and main content */
grid-template-rows: auto 1fr; /* Header and content area */
grid-template-areas:
    "sidebar header"
    "sidebar main";
height: 100vh;
}
```

```
/* Mobile layout adjustments */
```

```
@media (max-width: 768px) {
    .dashboard-grid {
        grid-template-columns: 1fr;
        grid-template-rows: auto auto 1fr; /* Header, Sidebar, Main */
        grid-template-areas:
            "header"
            "sidebar"
            "main";
    }
    #sidebar {
        position: static;
        height: auto;
    }
}
```

```
#header { grid-area: header; background-color: var(--bg-primary); border-bottom: 1px solid var(--border-color); }
```

```
#sidebar { grid-area: sidebar; background-color: var(--bg-primary); border-right: 1px solid var(--border-color); }
```

```
#main-content { grid-area: main; overflow-y: auto; }
```

```

/* General card styling */

.card {
    background-color: var(--bg-primary);
    border: 1px solid var(--border-color);
    transition: background-color 0.3s, border-color 0.3s;
}

</style>
</head>
<body>
    <div class="dashboard-grid">
        <!-- Header -->
        <header id="header" class="p-4 flex justify-between items-center">
            <h1 class="text-xl font-bold">Dashboard</h1>
            <div class="flex items-center gap-4">
                <span class="text-sm font-medium" style="color: var(--text-secondary);">Admin
User</span>
                <!-- Theme Switcher -->
                <button id="theme-switcher" class="p-2 rounded-full focus:outline-none focus:ring-
2 focus:ring-offset-2" style="background-color: var(--bg-secondary); focus:ring-color: var(--
accent-color);">
                    <i id="theme-icon" class="fa-solid fa-moon text-lg" style="color: var(--text-
primary);"></i>
                </button>
            </div>
        </header>

        <!-- Sidebar -->
        <aside id="sidebar" class="p-4 flex flex-col gap-4">

```

```
<nav>

  <ul class="space-y-2">

    <li><a href="#" class="flex items-center gap-3 p-2 rounded-lg"
style="background-color: var(--bg-secondary); color: var(--accent-color);"><i class="fa-solid
fa-chart-pie w-6 text-center"></i> Overview</a></li>

    <li><a href="#" class="flex items-center gap-3 p-2 rounded-lg hover:bg-gray-200
dark:hover:bg-gray-700"><i class="fa-solid fa-users w-6 text-center"></i> Users</a></li>

    <li><a href="#" class="flex items-center gap-3 p-2 rounded-lg hover:bg-gray-200
dark:hover:bg-gray-700"><i class="fa-solid fa-box w-6 text-center"></i> Products</a></li>

    <li><a href="#" class="flex items-center gap-3 p-2 rounded-lg hover:bg-gray-200
dark:hover:bg-gray-700"><i class="fa-solid fa-gear w-6 text-center"></i> Settings</a></li>

  </ul>

</nav>

</aside>
```

```
<!-- Main Content -->
```

```
<main id="main-content" class="p-6">
```

```
  <div class="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-4 gap-6">
```

```
    <!-- Stat Card 1 -->
```

```
    <div class="card p-5 rounded-xl shadow">
```

```
      <h3 class="font-semibold" style="color: var(--text-secondary);">Total Sales</h3>
```

```
      <p class="text-3xl font-bold mt-1">$24,805</p>
```

```
    </div>
```

```
    <!-- Stat Card 2 -->
```

```
    <div class="card p-5 rounded-xl shadow">
```

```
      <h3 class="font-semibold" style="color: var(--text-secondary);">New Users</h3>
```

```
      <p class="text-3xl font-bold mt-1">1,204</p>
```

```
    </div>
```

```
    <!-- Stat Card 3 -->
```

```
<div class="card p-5 rounded-xl shadow">

  <h3 class="font-semibold" style="color: var(--text-secondary);">Orders
Pending</h3>

  <p class="text-3xl font-bold mt-1">82</p>

</div>

<!-- Stat Card 4 -->

<div class="card p-5 rounded-xl shadow">

  <h3 class="font-semibold" style="color: var(--text-secondary);">Support
Tickets</h3>

  <p class="text-3xl font-bold mt-1">15</p>

</div>

<!-- Chart Card -->

<div class="card p-5 rounded-xl shadow lg:col-span-2">

  <h3 class="font-semibold mb-4">Sales Over Time</h3>

</div>

<!-- Table Card -->

<div class="card p-5 rounded-xl shadow lg:col-span-2">

  <h3 class="font-semibold mb-4">Recent Orders</h3>

  <div class="overflow-x-auto">

    <table class="w-full text-left">

      <thead>

        <tr style="border-bottom: 1px solid var(--border-color);">

          <th class="p-2">Order ID</th>

          <th class="p-2">Customer</th>

          <th class="p-2">Amount</th>

        </tr>

      </thead>

    </table>

  </div>

</div>
```



```

        </thead>

        <tbody>

            <tr><td class="p-2">#1024</td><td class="p-2">John Doe</td><td
class="p-2">$129.00</td></tr>

            <tr><td class="p-2">#1025</td><td class="p-2">Jane Smith</td><td
class="p-2">$49.99</td></tr>

            <tr><td class="p-2">#1026</td><td class="p-2">Sam Wilson</td><td
class="p-2">$250.50</td></tr>

        </tbody>

    </table>

</div>

</div>

</div>

</main>

</div>

```

```

<script>

    // --- DOM Elements ---

    const themeSwitcher = document.getElementById('theme-switcher');
    const themeIcon = document.getElementById('theme-icon');
    const rootElement = document.documentElement; // The <html> element

    // --- Functions ---

    /**
     * Toggles the theme between light and dark.
     */
    function toggleTheme() {

        // Toggle the 'dark' class on the <html> element
    }

```

```

rootElement.classList.toggle('dark');

// Check if the dark theme is now active
const isDarkMode = rootElement.classList.contains('dark');

// Update the icon accordingly
if (isDarkMode) {
    themelcon.classList.remove('fa-moon');
    themelcon.classList.add('fa-sun');
} else {
    themelcon.classList.remove('fa-sun');
    themelcon.classList.add('fa-moon');
}

// Optional: Save the user's preference in localStorage
localStorage.setItem('theme', isDarkMode ? 'dark' : 'light');
}

/**
 * Checks for a saved theme in localStorage and applies it on load.
 */
function applyInitialTheme() {
    const savedTheme = localStorage.getItem('theme');
    if (savedTheme === 'dark') {
        rootElement.classList.add('dark');
        themelcon.classList.remove('fa-moon');
        themelcon.classList.add('fa-sun');
    }
}

```

```
}

// --- Event Listeners ---

themeSwitcher.addEventListener('click', toggleTheme);

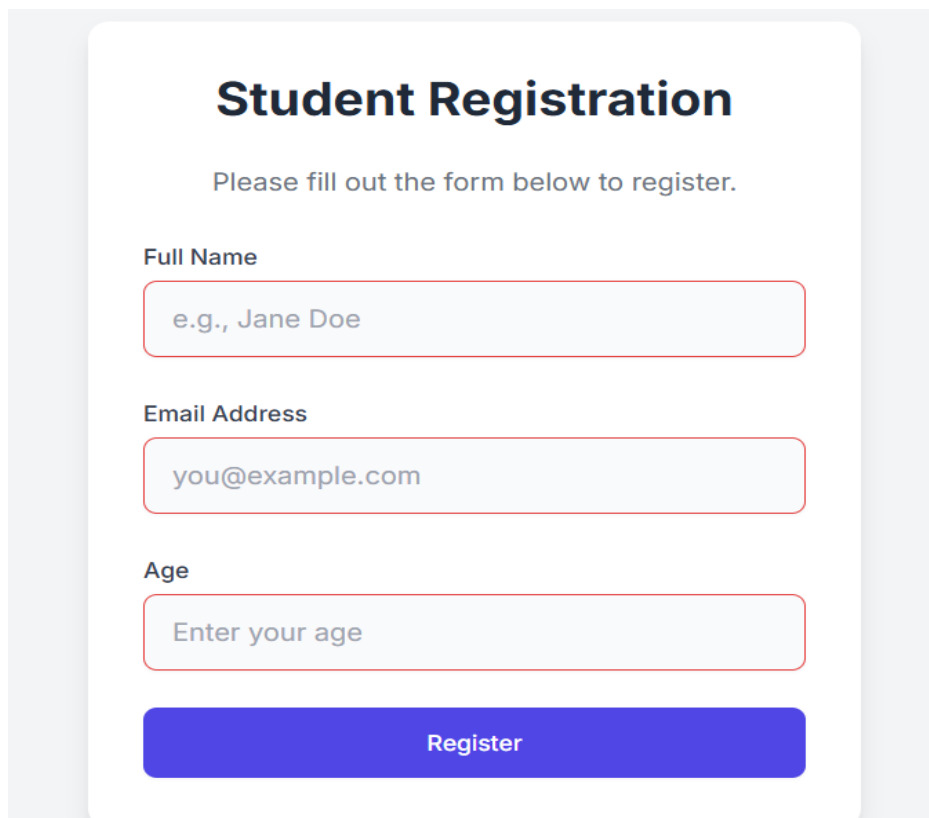

// --- Initial Setup ---

applyInitialTheme();

</script>
</body>
</html>
```

## OUTPUTS:

EXP 1:

A student registration form with a white background and rounded corners, set against a light gray background. The form contains a title, a instruction, three input fields, and a submit button.

**Student Registration**

Please fill out the form below to register.

**Full Name**

**Email Address**

**Age**

**Register**

*Figure 1 Student Registration*



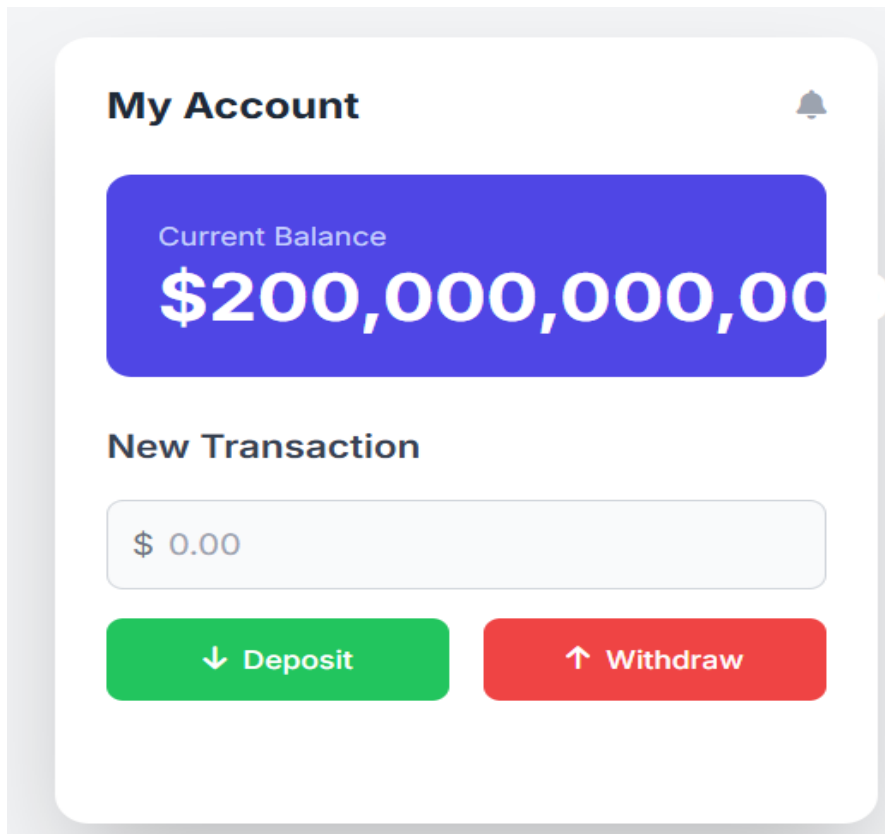


Figure 4 Account After Deposit

EXP 3:

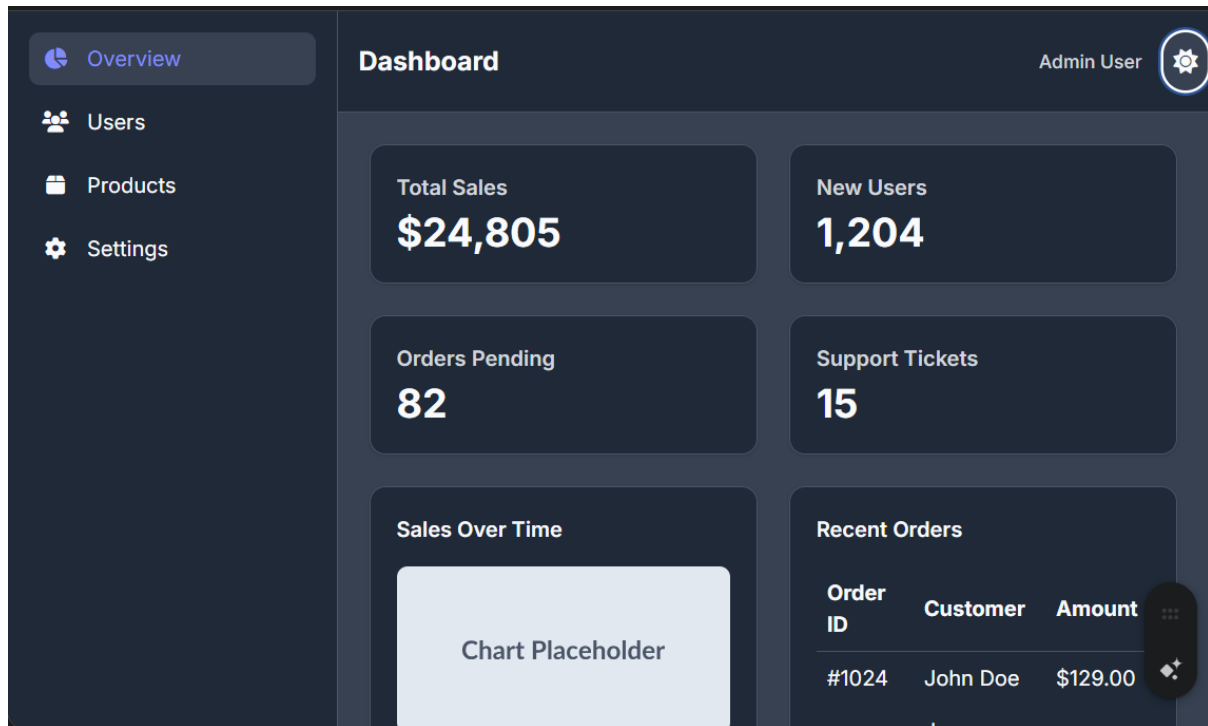


Figure 5 Dark Mode

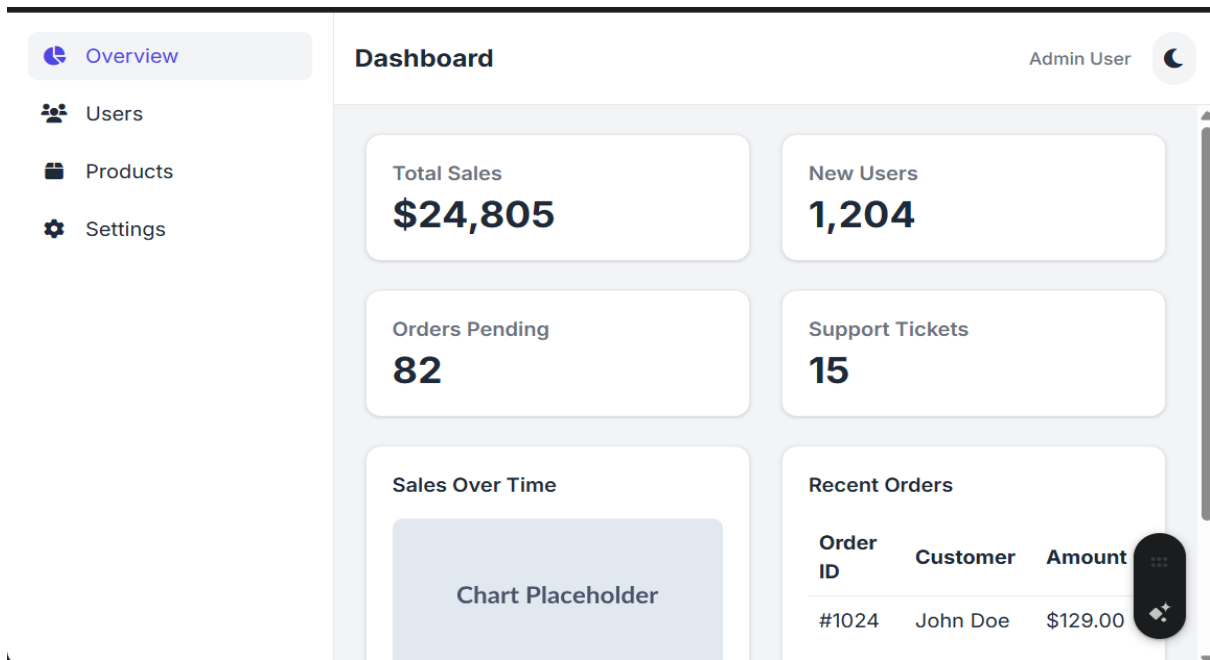


Figure 6 Light mode