FULL STACK

NAME :AMBIKA KASHYAP

CLASS:23BCC1-A

UID:23BCC70026

EXPERIMENT-3

**AIM**

To design and implement a responsive **Admin Dashboard** using HTML and CSS Grid Layout, incorporating a light/dark theme switching feature to enhance user experience and modern web design practices.

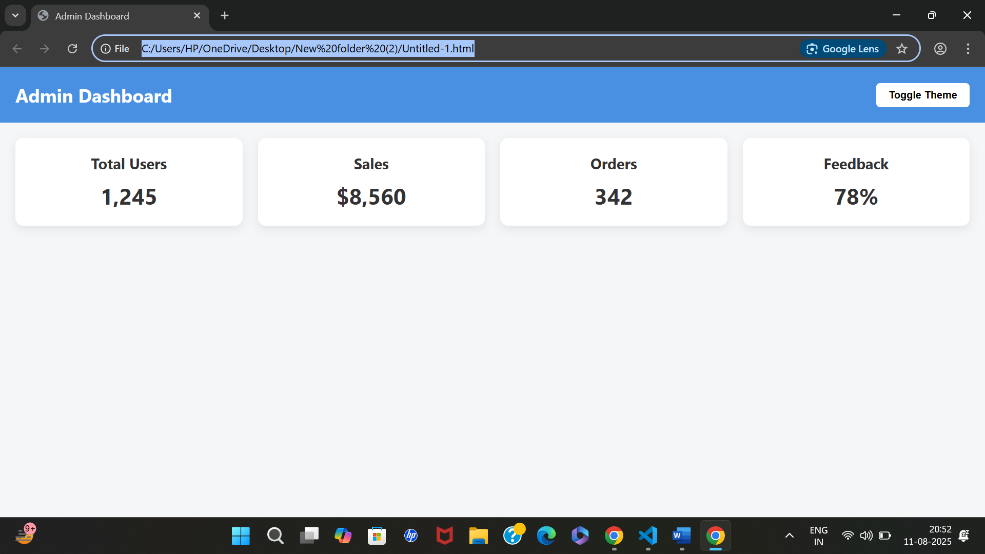
**THEORY**

1. **HTML Structure**
   * HTML defines the dashboard sections: sidebar navigation, top header, main content, and stats/cards.
   * Semantic tags like <header>, <aside>, <main>, and <section> improve accessibility and structure.
2. **CSS Grid Layout**
   * CSS Grid allows creating complex layouts with rows and columns.
   * Areas can be named for easier placement (e.g., grid-template-areas).
   * Grid is ideal for responsive, two-dimensional layouts.
3. **Theme Switching**
   * By defining **CSS variables** (--background-color, --text-color, etc.), theme colors can be toggled by adding/removing a class to the <body>.
   * This changes styles without rewriting all CSS.
4. **Responsive Design**
   * Media queries adjust grid layout for smaller devices, collapsing the sidebar into a top menu.
   * The dashboard adapts for desktop, tablet, and mobile views.
5. **Modern UI Principles**
   * Consistent spacing and typography.
   * Hover effects for interactive elements.
   * Minimalist, content-first design.

**PROCEDURE**

1. **Plan the Layout**
   * Sidebar navigation (links to sections).
   * Header bar with theme toggle button.
   * Main content with cards/statistics.
2. **Write the HTML Structure**
   * Use a grid container with sidebar, header, and main content.
   * Add cards for stats inside main content.
3. **Apply CSS Styling**
   * Define grid areas for layout.
   * Style sidebar, header, and cards.
   * Use CSS variables for theme colors.
4. **Add Theme Switching**
   * Define two sets of CSS variables (light & dark).
   * Use JavaScript to toggle a dark-theme class on <body>.
5. **Test and Optimize**
   * Check layout and theme switching on different devices.

**OUTPUT:**

****

**LEARNING OUTCOMES:**

After completing this experiment, the learner will be able to:

1. **Use CSS Grid** to create a responsive, multi-section web layout.
2. **Organize HTML** into semantic sections (header, sidebar, main content).
3. **Apply CSS variables** to create themeable designs.
4. **Implement theme switching** by toggling classes with JavaScript.
5. **Design responsive dashboards** that adapt to various devices.