**Assignment 4**

**Problem statement :**

Program on HTML basic tags to create forms and UI elements.

The goal of this project is to create a responsive web page that effectively adjusts its layout according to different screen sizes, ensuring an optimal viewing experience across devices (desktops, tablets, and smartphones). The page will utilize HTML for structure and CSS (specifically Flexbox) for layout management, allowing users to easily navigate the content without losing accessibility or aesthetics.

### Steps to Create the Responsive Web Page

#### Step 1: Create the HTML File

1. **Structure the HTML Document**:
   * Start with the <!DOCTYPE html> declaration to define the document type.
   * Use the <html> tag to create the root element of the document and specify the language.
   * Inside the <head> section, include metadata (like charset and viewport settings) and a link to the CSS stylesheet.
2. **Build the Header**:
   * Create a <header> section containing a main title and a navigation menu with links to different sections (Home, About, Services, Contact).
3. **Set Up the Main Content**:
   * Use the <main> tag to encapsulate the main content.
   * Create a hero section that welcomes users and briefly describes the website's purpose.
   * Add a features section that includes multiple feature boxes, each highlighting different aspects of the responsive design.
4. **Create the Footer**:
   * Add a <footer> section with copyright information.

#### Step 2: Create the CSS File

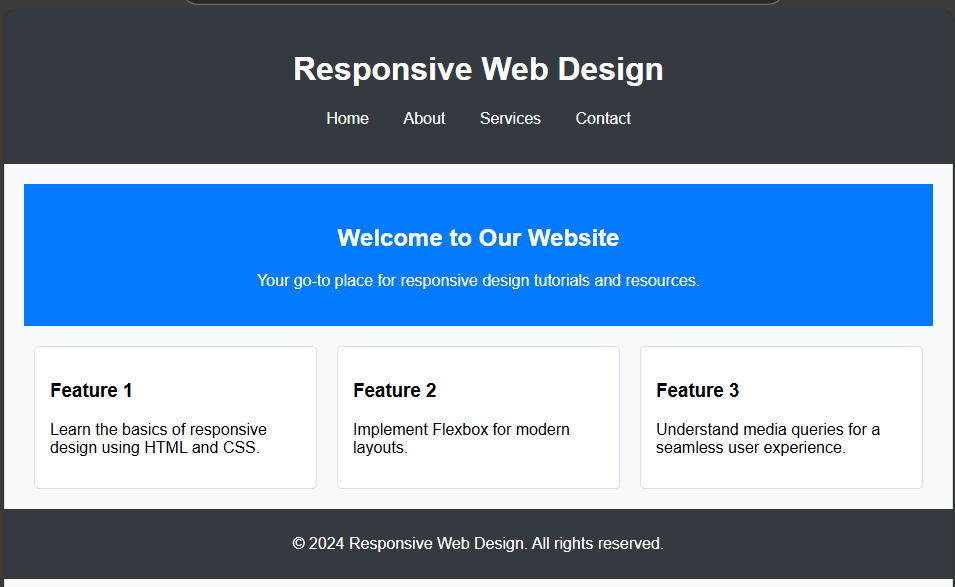
1. **Define General Styles**:
   * Set a base font family, margin, padding, and background color for the body to ensure consistency across the page.
2. **Style the Header**:
   * Apply background color, text color, padding, and center-align the text in the header section.
3. **Style the Navigation Menu**:
   * Remove default list styles and define how list items are displayed in the navigation.
4. **Define Styles for Main Content**:
   * Set maximum width for the main container and center it using auto margins.
   * Style the hero section with a background color and text styling.
5. **Implement Flexbox for Features**:
   * Use Flexbox to create a responsive layout for the features section, allowing boxes to be displayed in a row on larger screens.
6. **Style the Footer**:
   * Give the footer a dark background and center-align the text, ensuring it spans the full width of the page.

#### Step 3: Add Responsive Design

1. **Implement Media Queries**:
   * Use a media query to adjust styles for smaller screens (max-width of 768px).
   * Change the navigation menu to stack vertically instead of horizontally.
   * Change the layout of the features section from a row to a column.

#### Step 4: Save and Test

1. **Save the Files**:
   * Save the HTML code in a file named responsive\_web\_page.html.
   * Save the CSS code in a file named styles.css.
2. **Open in a Web Browser**:
   * Open the responsive\_web\_page.html file in a web browser to observe the responsive design in action.

**Screenshot of output:**