A close up of a logo

Description automatically generated

**CSS Styling**

**Disclaimer: The content is curated from online/offline resources and used for educational purpose only**

**LAB MANUAL**

**Responsive Two-Column Layout Design using Flexbox and Media Queries**

**Objective:**

The objective of this activity is to teach learners how to build a responsive two-column layout using CSS Flexbox combined with media queries. This exercise will help participants understand how flex containers and flex items behave, how to distribute space between columns, and how to ensure the layout adapts gracefully to different screen sizes, stacking columns vertically on smaller devices. By the end, learners should be able to create flexible, clean, and responsive two-column web designs suitable for web pages and applications.

**Equipment Required:**

* A computer with a text editor (e.g., VS Code, Sublime Text, Notepad++)
* A modern web browser such as Chrome, Firefox, or Edge with developer tools

**Prerequisites:**

* Basic knowledge of HTML structure and tags
* Familiarity with CSS selectors, properties, and values
* Basic understanding of CSS Flexbox concepts and syntax
* Basic understanding of CSS media queries for responsive design
* Ability to save and open .html files in a web browser

**Problem Statement:**

Your task is to design a web page that features a two-column layout using CSS Flexbox. The layout should divide the page into two sections arranged horizontally side-by-side on larger screens, with equal or adjustable widths. On smaller screen widths (such as mobile devices), the two columns should stack vertically to enhance readability and usability. You will implement the flex container and flex items, apply necessary flex properties, and write media queries to adapt the layout based on the screen size.

**Procedure:**

1. Create a new file named ***responsive\_two\_column.html***.
2. Define the basic HTML structure including <head> and <body>.
3. Inside the <body>, create a container <div> representing the flex container.
4. Add two child <div> elements for the left and right columns within the container.
5. Add content placeholders such as headings, paragraphs, or images in each column.
6. In the <head>, include a <style> section or link an external CSS file for styling.
7. Use Flexbox properties on the container (display: flex; flex-direction: row;) and set widths or flex-grow for the columns.
8. Create a media query targeting screen widths less than 768px (or your chosen breakpoint) that changes the flex container direction to column and sets the columns to full width.
9. Save and open the file in a browser, then resize the browser window to test responsiveness.

**Code**

***responsive\_two\_column.html***

 <!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

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

  <title>Responsive Two-Column Layout</title>

  <style>

    body {

      margin: 0;

      font-family: Arial, sans-serif;

      background: #f2f2f2;

    }

    .container {

      display: flex;

      flex-wrap: wrap;

      gap: 20px;

      padding: 20px;

    }

    .column {

      flex: 1;

      min-width: 300px;

      padding: 26px 22px 22px 22px;

      border-radius: 14px;

      box-shadow: 0 4px 18px rgba(79,140,255,0.10);

      border: 2px solid #a5b4fc;

      transition: box-shadow 0.2s, border-color 0.2s;

    }

    .column:first-child {

      background: linear-gradient(135deg, #fff7fa 60%, #f3ffe6 100%);

      border-color: #fbcfe8;

    }

    .column:last-child {

      background: linear-gradient(135deg, #e6f7ff 60%, #f3e8ff 100%);

      border-color: #bae6fd;

    }

    .column h2 {

      color: #0e7490;

      margin-top: 0;

      letter-spacing: 1px;

    }

    .column:last-child h2 {

      color: #7c3aed;

    }

    .column p {

      color: #374151;

      font-size: 1.08em;

    }

    /\* Two-column layout for larger screens \*/

    @media (min-width: 768px) {

      .container {

        flex-direction: row;

      }

    }

    /\* One-column layout for small screens \*/

    @media (max-width: 767px) {

      .container {

        flex-direction: column;

      }

    }

  </style>

</head>

<body>

  <div class="container">

    <div class="column">

      <h2>Favorite Comfort Foods</h2>

      <img src="https://images.unsplash.com/photo-1504674900247-0877df9cc836?auto=format&fit=crop&w=400&q=80" alt="Pizza" style="width:100%; border-radius:8px; margin-bottom:12px;">

      <ul style="font-size:1.08em; color:#374151; margin:0 0 10px 18px;">

        <li>Cheesy Margherita Pizza</li>

        <li>Chocolate Lava Cake</li>

        <li>Spicy Masala Maggi</li>

        <li>Classic French Fries</li>

        <li>Homemade Paneer Tikka</li>

      </ul>

      <a href="https://www.bonappetit.com/recipes" target="\_blank" style="color:#f59e42; text-decoration:underline; font-size:0.98em;">Try a new recipe</a>

    </div>

    <div class="column">

      <h2>Simple Joys</h2>

      <img src="https://images.unsplash.com/photo-1464983953574-0892a716854b?auto=format&fit=crop&w=400&q=80" alt="Sunset" style="width:100%; border-radius:8px; margin-bottom:12px;">

      <ul style="font-size:1.08em; color:#374151; margin:0 0 10px 18px;">

        <li>Watching sunsets from the rooftop</li>

        <li>Finding a new favorite song</li>

        <li>Freshly brewed coffee on a rainy day</li>

        <li>Laughing with friends until it hurts</li>

        <li>Receiving a handwritten letter</li>

      </ul>

      <a href="https://www.ted.com/topics/happiness" target="\_blank" style="color:#f472b6; text-decoration:underline; font-size:0.98em;">Explore happiness talks</a>

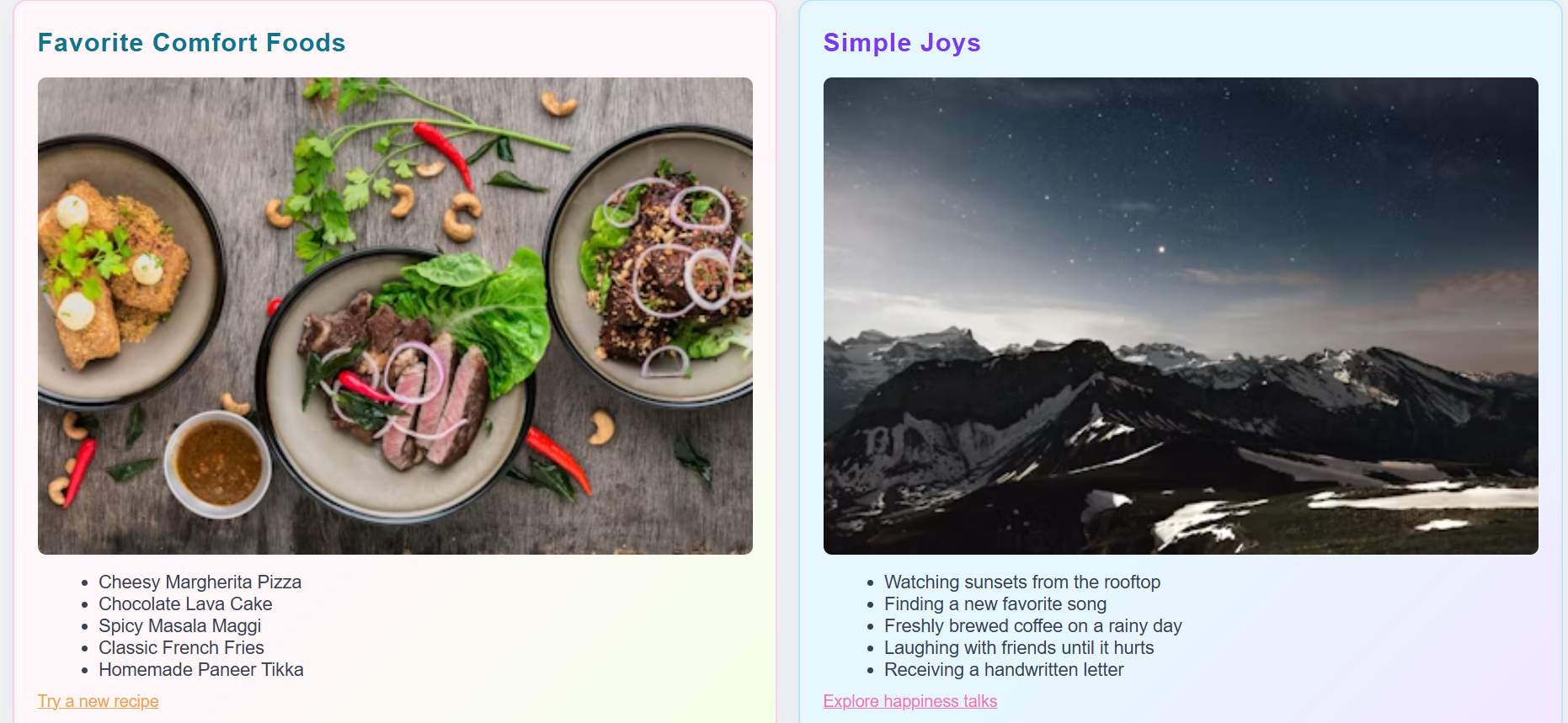
    </div>

  </div>

</body>

</html>

**Output**

****