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**

**Create responsive Image gallery using CSS Grid & Media Queries**

**Objective:**

The objective of this activity is to teach learners how to design a fully responsive image gallery using CSS Grid combined with media queries. Through this exercise, participants will learn how to define a grid layout, control column and row sizing, and use breakpoints to adjust the number of columns for various screen sizes. By the end, learners will be able to create an image gallery that looks aesthetically pleasing and works well across desktops, tablets, and mobile devices.

**Equipment Required:**

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

**Prerequisites:**

* Basic HTML knowledge
* Familiarity with CSS selectors, properties, and values
* Understanding of CSS Grid basics and media queries
* Ability to save and open .html files in a browser

**Problem Statement:**

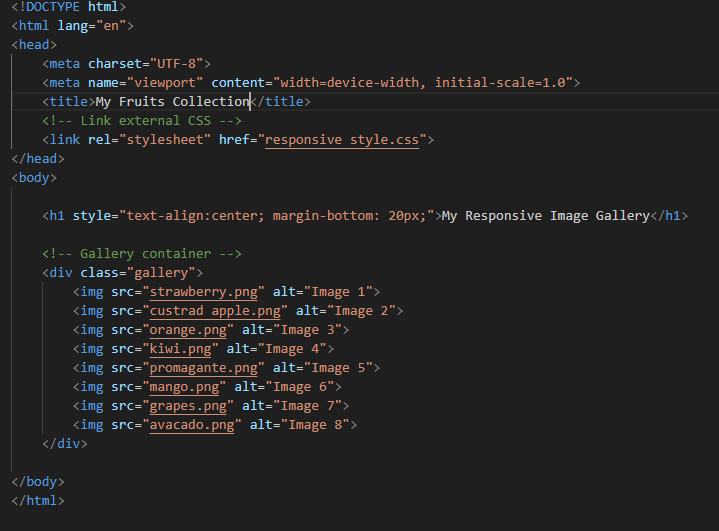
You are required to design a web page that displays a responsive image gallery using CSS Grid. The gallery should consist of multiple images arranged neatly in rows and columns, automatically adjusting the number of columns depending on the viewer’s screen size. On large screens, more columns should be visible, while fewer columns should appear on smaller devices for optimal viewing. The gallery should also feature clean spacing and consistent image presentation.

**Procedure:**

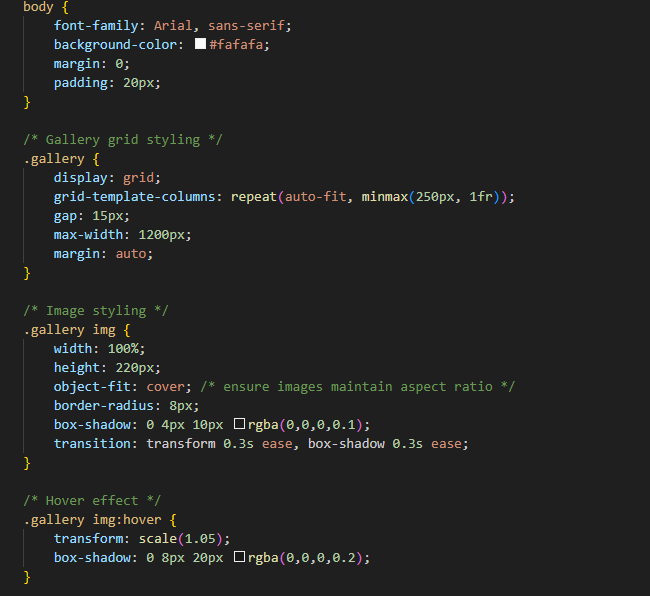
1. Create a new HTML file named ***responsive\_gallery.html***.
2. In the HTML file, add a <div> container to hold your images.
3. Insert multiple <img> elements inside the container, each representing a gallery item.
4. Create a CSS file named gallery\_styles.css and link it to your HTML file.
5. In CSS, use display: grid on the container to define a flexible grid layout.
6. Use the grid-template-columns property with the repeat() and minmax() functions for automatic column sizing.
7. Add gap to space out images evenly.
8. Set width: 100% and object-fit: cover for images to ensure they scale properly.
9. Use media queries to modify the number of columns on smaller screen widths.
10. Save and preview in your browser, resizing the window to test responsiveness.

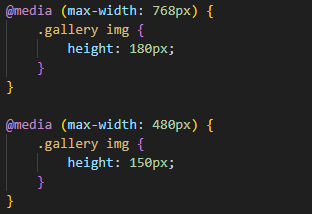
**Code**

***responsive\_gallery.html***

****

**Responsive\_style.css**

****

****

**Output**

****