

## Report Task-1

- **Mention Name or email id :**

veenumaheshwari99@gmail.com

- **Task Title :**

Responsive Company Member Team Page

- **Task Description :**

To develop a responsive web page for the company Member Team Page using HTML and CSS for implementation.

- **Steps Taken :**

**HTML Structure:**

- Document Structure:

Used the `<!DOCTYPE html>` declaration.

Created the HTML document with the `<html>` tag and set the language attribute to "en".

Defined the head section with meta tags for character set and viewport, a title for the page, and a link to an external CSS file.

Structured the body with various sections and buttons.

- Buttons and Sections:

- Added buttons for different sections such as "ER Info Tech," "Leadership / Management Team," "Development Team," and "Sales and Marketing Team."

- Used `<div>` elements with the class "founder" to represent team members.

- Included images, headings, and paragraphs for each team member.

### **CSS Styling:**

- Body Styling:

Set display: flex to the body to use flexbox for layout.

Aligned items in the center vertically and horizontally.

Applied a margin at the top.

- Button Styling:

Styled the buttons with padding, font size, background color, and border radius.

- Introduction Styling:

Styled the introduction paragraph with a larger font size and centered text.

- Divider Styling:

Created a dotted line divider with a width of 50% and a margin.

- Founder Styling:

Styled the founder div with text alignment, margin, and image size.  
Used media queries for responsive design based on screen width.

### **Responsive Design:**

- Media Queries:

Defined different styles for the ".founder" class based on screen width.

Adjusted margins for better spacing on smaller screens.

- Flexbox:

Utilized display: flex for containers to create a flexible layout.

Used flex-wrap and justify-content for responsive item arrangement.

- Recommendations:

Alt Text for Images:

Consider adding meaningful alternative text to images for accessibility.

- External CSS:

Move inline styles to the external CSS file for better organization.

- Consistency:

Ensure consistent styling across sections for a unified visual appearance.

- **Challenges Faced :**

Responsive Design:

Ensuring a consistent and appealing layout across various screen sizes can be challenging. It looks like you've used media queries effectively to address this.

Flexbox Usage:

Working with flexbox for layout can be challenging, especially when trying to align items both vertically and horizontally. It seems like you've handled this well.

Image Handling:

Managing images, especially ensuring they are properly sized and have appropriate alt text for accessibility, can be a consideration.

Maintainability:

As your project grows, maintaining and organizing your CSS code becomes crucial. Consider using separate CSS files for better organization and readability.

#### Cross-Browser Compatibility:

Ensuring that your design looks consistent across different web browsers can sometimes be challenging. Testing on various browsers is essential.

#### Accessibility:

Providing meaningful alternative text for images is important for accessibility. Ensure that all users, including those using screen readers, can understand the content.

#### Consistency in Styling:

Maintaining a consistent visual style across different sections and components can be challenging. It's essential to have a cohesive design.

- **Solutions Implemented :**

- Effective use of flexbox to create a centered and responsive layout.
- Media queries to handle responsiveness for different screen sizes.
- Consistent button styling for a cohesive design.

- **Learnings :**

- ✓ Flexbox Layout:

Effective use of flexbox properties to create a centered and vertically aligned layout.

Flex containers and items are used to organize content in a flexible and responsive manner.

✓ Media Queries:

Utilization of media queries to make the design responsive to different screen sizes.

Adjustments in margins and styles based on the screen width.

✓ Button Styling:

Consistent styling for buttons using padding, font size, background color, and border-radius.

✓ CSS Styling:

Implementation of CSS styles for various elements, including buttons, dividers, and footer details.

Balancing aesthetic design with readability and responsiveness.

✓ HTML Structure:

Proper HTML structure with a clear separation of head and body sections.

Inclusion of appropriate tags for semantic HTML.

✓ Responsive Design:

Responsive adjustments for footer details and team sections based on different screen sizes.

✓ Project Organization:

Incorporation of a separate CSS file for styling, promoting a more organized project structure.

✓ Alt Text for Images:

Inclusion of alternative text for images, enhancing accessibility.

- **Project Updates :**

This task is implemented using HTML and CSS code.