Question 1 and 2

Q1. Differences between CSS Flexbox and CSS Grid Layout Models.

Answer:

Flexbox:

- **One-dimensional layout**: Flexbox is designed for laying out items in a single direction (either row or column).
- **Content alignment**: It excels at aligning items along a single axis and distributing space within a container.
- **Use case**: Ideal for simple layouts, such as navigation bars, aligning items in a row or column, and centering content.

Grid:

- **Two-dimensional layout**: CSS Grid is designed for layouts in both rows and columns.
- **Complex layouts**: It allows for more complex layouts with precise control over the placement of items.
- **Use case**: Best for creating entire page layouts, complex grids, and when you need to control both rows and columns simultaneously.

Q2. Key Properties in the Flexbox Layout Model.

Answer:

1. **justify-content**: Aligns the flex items along the main axis (horizontal by default). Values include `flex-start`, `flex-end`, `center`, `space-between`, `space-around`, and `space-evenly`.
2. **align-items**: Aligns the flex items along the cross axis (vertical by default). Values include `flex-start`, `flex-end`, `center`, `baseline`, and `stretch`.
3. **gap**: Defines the space between flex items. It can be set in any CSS length unit (e.g., `10px`, `1rem`).
4. **flex-direction**: Specifies the direction of the flex items. Values include `row`, `row-reverse`, `column`, and `column-reverse`.
5. **flex-wrap**: Controls whether the flex items should wrap or not. Values include `nowrap`, `wrap`, and `wrap-reverse`.