Ans.1 a. Flexbox Layout :

* Flexbox is primarily designed for one-dimensional layouts, such as arranging items in a row or a column.
* It excels at distributing space along a single axis and is great for handling the alignment of items within that axis.
* It is suitable for creating dynamic and flexible layouts when the size of content is unknown or changes dynamically.

b. Grid Layout :

* CSS Grid is designed for two-dimensional layouts, where you have both rows and columns.
* It allows for precise control over both the rows and columns, making it well-suited for creating grid-like structures with complex layouts.
* It's often used for creating responsive web designs with defined rows and columns.

When to choose one over other:

Flexbox is suitable for creating flexible, one-dimensional layouts where content may vary in size or order. Use it for things like navigation menus, flexible content containers, and handling content that flows in a single direction (horizontally or vertically).

Grid is a better choice when you need to create complex, two-dimensional layouts, such as grids of images or content, where items align both horizontally and vertically, or when you have a fixed grid structure in mind.

Ans.02

1. **Justify-content:** This property defines how flex items are aligned along the main axis (the primary direction of the flex container).
2. **Align-items:** This property defines how flex items are aligned along the cross-axis (the perpendicular direction to the main axis).
3. **Gap:** It is the property to specify the size of the gap or spacing between rows and columns in a grid container.
4. **Flex-direction:** This property specifies the direction in which flex items are placed within the flex container.
5. **Flex-wrap:** This property determines whether flex items are allowed to wrap onto new lines if they cannot fit in the container.