CSS Flexbox

Question 1: What is CSS Flexbox, and how is it useful for layout design? Explain the terms flex-container and flex-item.
Ans:

CSS Flexbox is a modern CSS layout system designed to help developers create responsive and flexible layouts more easily compared to older methods like floats or tables. It allows elements within a container to align, distribute space, and adjust size dynamically depending on the screen size or available space.

Flexbox is Useful for Layout Design :-
1. Responsive design - adjusts automatically to different screen sizes.
2. Easy alignment - vertically and horizontally center elements without complex CSS.
3. Flexible sizing - items can grow/shrink relative to available space.
4. Order control - change the visual order of items without changing the HTML structure.
5. Space distribution - evenly distribute space between items or around them.

Terms Flex-container and Flex-items :-

1. Flex-container :
   The parent element that holds the flex items.
   Declared by setting its display property to flex or inline-flex.
   Controls the layout behavior of its children.
   Example:
   .container {
     display: flex;
   }

2. Flex-items :

The direct child elements inside a flex container.
They automatically become flexible and can grow, shrink, or align based on the container's properties.

Example:
```
<div class="container">
  <div class="item">Item 1</div>
  <div class="item">Item 2</div>
  <div class="item">Item 3</div>
</div>
```

Question 2: Describe the properties justify-content, align-items, and flex direction used in Flexbox.

Ans :

1. <u>Justify-content :</u>

This property controls horizontal alignment of flex items along the main axis, the direction in which items are laid out — by default, left-to-right. It defines how the extra space in the container is distributed between flex items.

<u>Common Value :-</u>

1. flex-start - Items packed at the start default.
2. flex-end - Items packed at the end.
3. center - Items centered.
4. space-between - Equal space *between* items first at start, last at end.
5. Space-around - Equal space *around* each item including edges.
6. Space-evenly - Equal spacing *everywhere*.

2. <u>Align-itens :</u>

Defines how flex items are aligned along the cross axis vertical by default. It controls the vertical positioning of items inside the container.

Common values:-

1. Stretch - Items stretch to fill container height default.
2. flex-start - Items aligned at the top.
3. flex-end - Items aligned at the bottom.
4. Center - Items vertically centered.
5. Baseline - Items aligned by their text baseline.

3. Flex-direction :
        Defines the main axis direction, whether items are laid out in rows or columns.

Common value :-

1. Row - Items placed left to right default.
2. Row-reverse - Items placed right to left.
3. Column - Items placed top to bottom.
4. Column-reverse - Items placed bottom to top.