

Q1. Describe the main difference between the CSS flexbox layout model and the CSS grid layout model. When would you choose to use one over another?

Ans. The main difference between the CSS Flexbox and CSS grid layout model is :

Flexbox: Designed for one-dimensional layouts, either a row or a column.

It's best for aligning items in a single direction and distributing space within a container.

While

Grid: Designed for two-dimensional layouts, both rows and columns. It's ideal for creating complex layouts involving both horizontal and vertical alignments.

To use:

Flexbox is used when you need to align items in a single row or column, such as a navigation bar or a toolbar.

Grid is Used when you need to create a full-page layout or complex grid-based designs, such as a photo gallery or a web page structure.

Q2. Explain the role of the following key properties in the flexbox layout model.

1. Justify-content
2. Align-items
3. gap
4. Flex-direction
5. Flex-wrap

Ans. **1 .Justify-content**

Values:

- flex-start: Items are packed at the start of the main axis.
- flex-end: Items are packed at the end of the main axis.
- center: Items are centered along the main axis.
- space-between: Items are evenly distributed with the first item at the start and the last item at the end.
- space-around: Items are evenly distributed with equal space around them.

2. align-items:

- Role: Aligns flex items along the cross-axis (vertical by default).

Values:

- flex-start: Items are aligned at the start of the cross-axis.
- flex-end: Items are aligned at the end of the cross-axis.
- center: Items are centered along the cross-axis.
- baseline: Items are aligned along their baselines.
- stretch: Items are stretched to fill the container.

3. gap:

- **Role:** Defines the space between flex items.
- **Values:** A length value (e.g., `10px`, `1rem`) specifying the gap between items.

4. flex-direction:

- **Role:** Specifies the direction of the main axis, determining the direction in which flex items are placed in the flex container.
- **Values:**

row: Items are placed in a row, from left to right.

row-reverse: Items are placed in a row, from right to left.

column: Items are placed in a column, from top to bottom.

column-reverse: Items are placed in a column, from bottom to top.

5.flex-wrap:

- **Role:** Controls whether flex items should wrap onto multiple lines along the cross axis.
- **Values:**
 - nowrap: All flex items are placed in a single line (default).
 - wrap: Flex items wrap onto multiple lines from top to bottom.
 - wrap-reverse: Flex items wrap onto multiple lines from bottom to top.

Q3. write the code to center a div using CSS flexbox.

Ans. <https://amanisl.github.io/CSS-flexbox/>

Q4.

Ans. <https://amanisl.github.io/customerpr/>

Q5.Build a clone of the IRCTC Ticket booking page.

Ans. code and output – click the link below

<https://amanisl.github.io/irctc-clone/>

