Question 1: Explain CSS Grid and How It Differs from Flexbox

CSS Grid:

CSS Grid is a two-dimensional layout system that allows you to create layouts by defining rows and columns. It provides a powerful way to control both horizontal and vertical alignment simultaneously.

Flexbox:

Flexbox is a one-dimensional layout system used for aligning items either in a row (**main axis**) or a column (**cross axis**). It is ideal for layouts where elements align in a single direction.

Differences Between Grid and Flexbox:

Aspect	**CSS Grid**	**Flexbox**
Dimension	Two-dimensional (rows and columns).	One-dimensional (row **or** column).
Use Case	Complex layouts with multiple rows/columns.	Aligning or distributing items in a single direction.
Parent-Child Relation	Defines layout for both parent and child items.	Focuses on aligning child items.
Examples	Webpage layouts, dashboards.	Navigation bars, buttons, toolbars.

When to Use Grid Over Flexbox:

1. **Use Grid**:
    - When designing a complex layout with both rows and columns.
    - For creating grids, dashboards, or galleries.
2. **Use Flexbox**:
    - When aligning elements in a single row or column.
    - For small-scale alignment like menus, buttons, or forms.

Question 2: Describe grid-template-columns, grid-template-rows, and grid-gap

1. grid-template-columns:

- **Definition**: Specifies the number and size of columns in the grid.
- **Syntax**:
- grid-template-columns: <size> <size> ...;
- **Example**:

- .grid {
-    display: grid;
-    grid-template-columns: 200px 1fr 200px;
- }

Creates three columns: two fixed at 200px and one flexible (1fr).

---

2. grid-template-rows:

- **Definition**: Specifies the number and size of rows in the grid.
- **Syntax**:
- grid-template-rows: <size> <size> ...;
- **Example**:
- .grid {
-    display: grid;
-    grid-template-rows: 100px auto 100px;
- }

Creates three rows: a fixed height of 100px for the first and third rows, and the second row takes up remaining space.

---

3. grid-gap (or gap):

- **Definition**: Defines the spacing between rows and columns in the grid.
- **Syntax**:
- grid-gap: <row-gap> <column-gap>;
- **Example**:
- .grid {
-    display: grid;
-    grid-template-columns: 1fr 1fr 1fr;
-    grid-template-rows: auto auto;
-    grid-gap: 20px 10px;
- }

Adds 20px space between rows and 10px space between columns.

Example Combining Properties:

```
.container {
    display: grid;
    grid-template-columns: repeat(3, 1fr);
    grid-template-rows: 150px 150px;
    gap: 20px;
}
```

This creates a grid with:

- **3 equal-width columns**.
- **2 rows of 150px height**.
- A uniform 20px gap between both rows and columns.

Summary Table:

Property	**Purpose**	**Example**
grid-template-columns	Defines the number and width of columns.	grid-template-columns: 1fr 2fr;
grid-template-rows	Defines the number and height of rows.	grid-template-rows: auto 200px;
grid-gap / gap	Sets the spacing between grid rows/columns.	grid-gap: 10px 20px;