Q1. Explain CSS Grid and how it differs from Flexbox. When would you use Grid over Flexbox?

Ans: CSS Grid: CSS Grid is a powerful layout system that allows you to create two-dimensional layouts (both rows and columns). It gives full control over arranging items in a structured grid, similar to tables but more flexible and responsive.

How It Differs from Flexbox: Flexbox, on the other hand, is mainly used for one-dimensional layouts (either a row *or* a column)

Difference Between CSS Grid and Flexbox:

Feature	CSS Grid	Flexbox
Layout Type	2-D (rows + columns)	1-D (row **or** column)
Alignment	Controls alignment in both directions simultaneously	Alignment along one axis at a time
Item Placement	Can place items anywhere in the grid	Items flow naturally in one direction
Best For	Complete page layouts	Aligning items inside a small container

When to Use CSS Grid Over Flexbox

Use CSS Grid when:

- You need a full webpage layout with rows and columns.
- You want to create a fixed or responsive grid (e.g., galleries, dashboards).
- You need to position items precisely (spanning rows/columns).

Use Flexbox when:

- You are aligning items in a single row or column.
- You need simple horizontal/vertical centering.
- You are designing nav bars, buttons, or small UI components

Q2. Describe the grid-template-columns, grid-template-rows, and grid-gap properties. Provide examples of how to use them.

Ans: Grid Properties in CSS Grid: CSS Grid uses specific properties to define the structure of rows and columns. Three important ones are grid-template-columns, grid-template-rows, and grid-gap.

1. grid-template-columns

This property defines how many columns a grid will have and how wide each column should be.

Example:

its Creates 3 columns, each 200px wide:-

.container {

  display: grid;

  grid-template-columns: 200px 200px 200px;

}

Another example using fractions (fr):

grid-template-columns: 1fr 2fr;

First column takes 1 part, second takes 2 parts

2. grid-template-rows

This property sets the number and height of rows.

Example:

Creates 2 rows, first 100px tall, second 150px:-

.container {

  display: grid;

  grid-template-rows: 100px 150px;

}

we can also use auto:

grid-template-rows: auto auto;

Rows grow based on content.

3. grid-gap (or gap)

This property adds space between rows and columns in the grid.

Example:

Adds 20px space between all grid items:-

.container {

  display: grid;

  gap: 20px;

}

We can set row and column gaps separately:

gap: 10px 30px;

Complete Example:-

.container {

  display: grid;

  grid-template-columns: 1fr 1fr 1fr;

  grid-template-rows: 100px auto;

  gap: 15px;

}