

CSS Grid Lecture Notes

1. Grid Container

The element where grid behavior begins. Apply `display: grid;` (or `inline-grid`) to make it a grid container. All direct children become grid items. Example:

```
.container { display: grid; }
```

Use case: Turns a normal div into a flexible layout container.

2. Grid Items

Children of a grid container. They are placed into grid cells created by rows/columns. Example:

```
.item { grid-column: 1 / 3; grid-row: 2; }
```

Use case: Gives control over where each element sits.

3. Defining Rows and Columns

Use `grid-template-columns` and `grid-template-rows` to set structure. Example:

```
.container { display: grid; grid-template-columns: 200px 1fr 1fr; grid-template-rows: 100px auto; }
```

Use case: Design specific track sizes (sidebar, main content).

4. repeat() Function

Shorthand for repeating rows/columns. Example:

```
.container { grid-template-columns: repeat(3, 1fr); }
```

Use case: Quickly define evenly spaced grids like galleries.

5. Grid Gap

Controls space between rows/columns. Example:

```
.container { gap: 20px; }
```

Use case: Avoids margin hacks.

6. Grid Template Areas

Name and arrange sections of layout. Example:

```
.container { grid-template-areas: 'header header' 'nav main' 'footer footer'; grid-template-columns: 1fr 3fr; }
```

```
header { grid-area: header; } nav { grid-area: nav; }
```

Use case: Readable page layouts.

7. Alignment

Controls how items align in cells and how grid aligns in container. Example:

```
.container { justify-items: center; align-items: start; }
```

Use case: Perfect for centering items without flexbox.

8. Nested Grid

A grid inside another grid item. Example:

```
.parent { display: grid; grid-template-columns: 1fr 1fr; } .child { display: grid; grid-template-rows: auto auto; }
```

Use case: Dashboards, cards, multi-level layouts.

9. Implicit vs Explicit Grid

Explicit: defined by grid-template. Implicit: created automatically when items exceed grid. Example:

```
.container { grid-template-columns: 100px 100px; grid-auto-rows: 50px; }
```

Use case: Handles unknown number of items.

10. fr Unit

Fractional unit for flexible sizing. Example:

```
.container { grid-template-columns: 1fr 2fr 1fr; }
```

Use case: Responsive and flexible layouts.