NotebookLM

You are the architect. You define an explicit grid.

You use properties like `grid-template-columns` and `grid-template-rows` to draw the initial blueprint for your content.

```
.grid-container {
  display: grid;
  grid-template-columns: repeat(3, 1fr);
  grid-template-rows: repeat(2, 150px);
}
```

NotebookLM

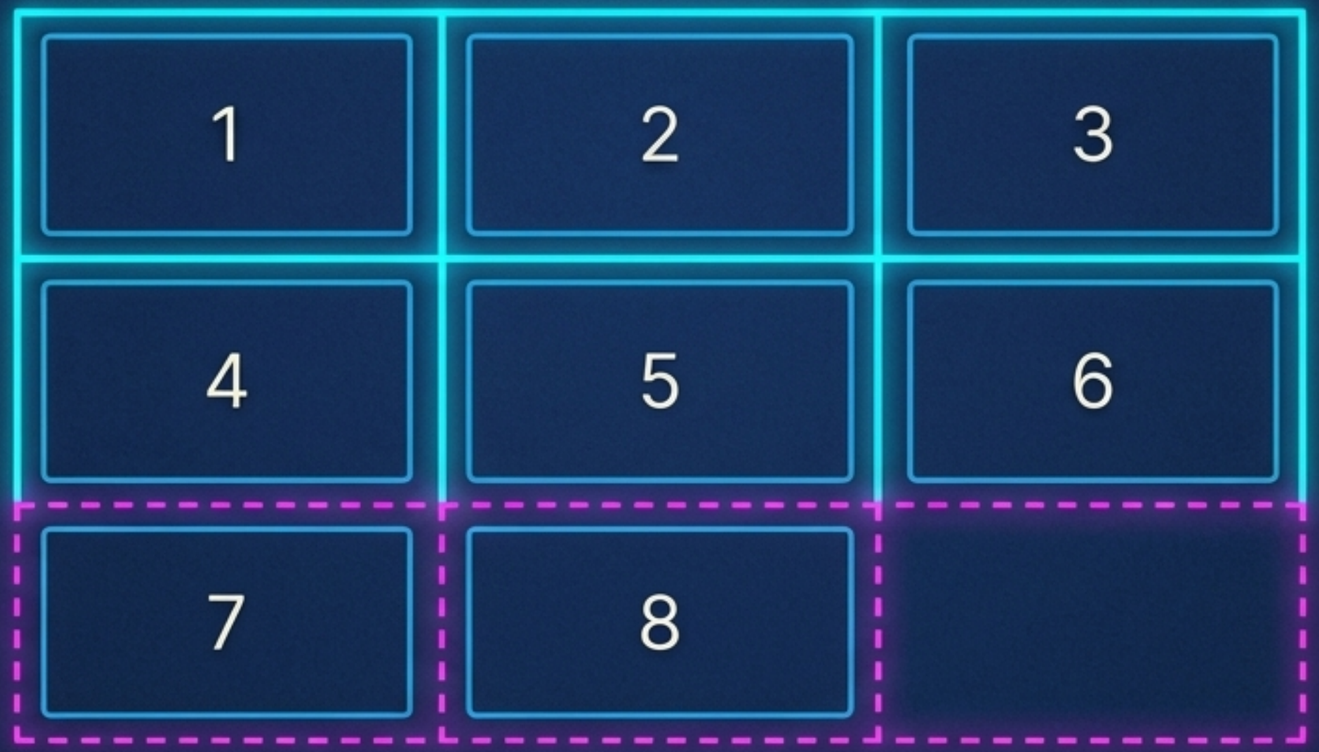
Then you add more items than the blueprint can hold.

What happens now?

NotebookLM

Grid doesn’t panic — it quietly creates more grid.

That extra structure, created automatically to hold the overflowing items, is the **implicit grid**.



NotebookLM

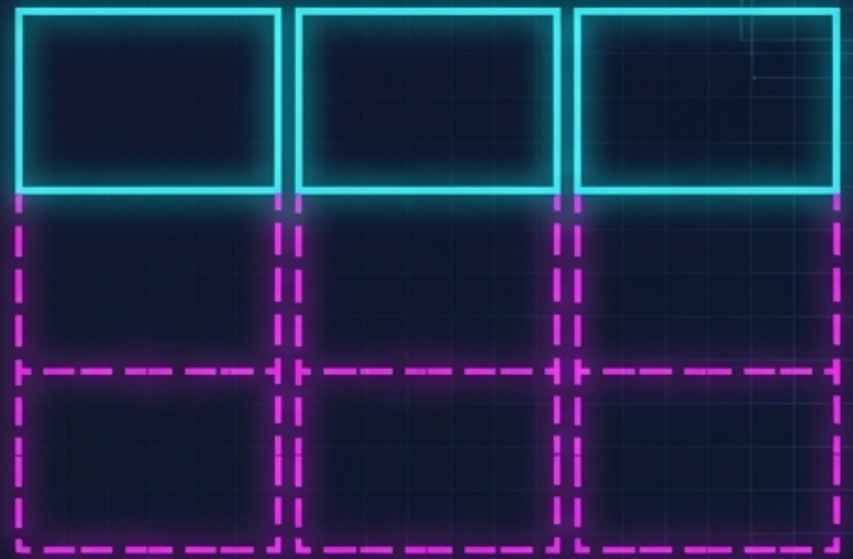
Explicit vs. Implicit

Explicit Grid

The tracks *you* define with `grid-template-*` properties. This is your planned blueprint.

Implicit Grid

The tracks Grid creates automatically when items overflow your blueprint.

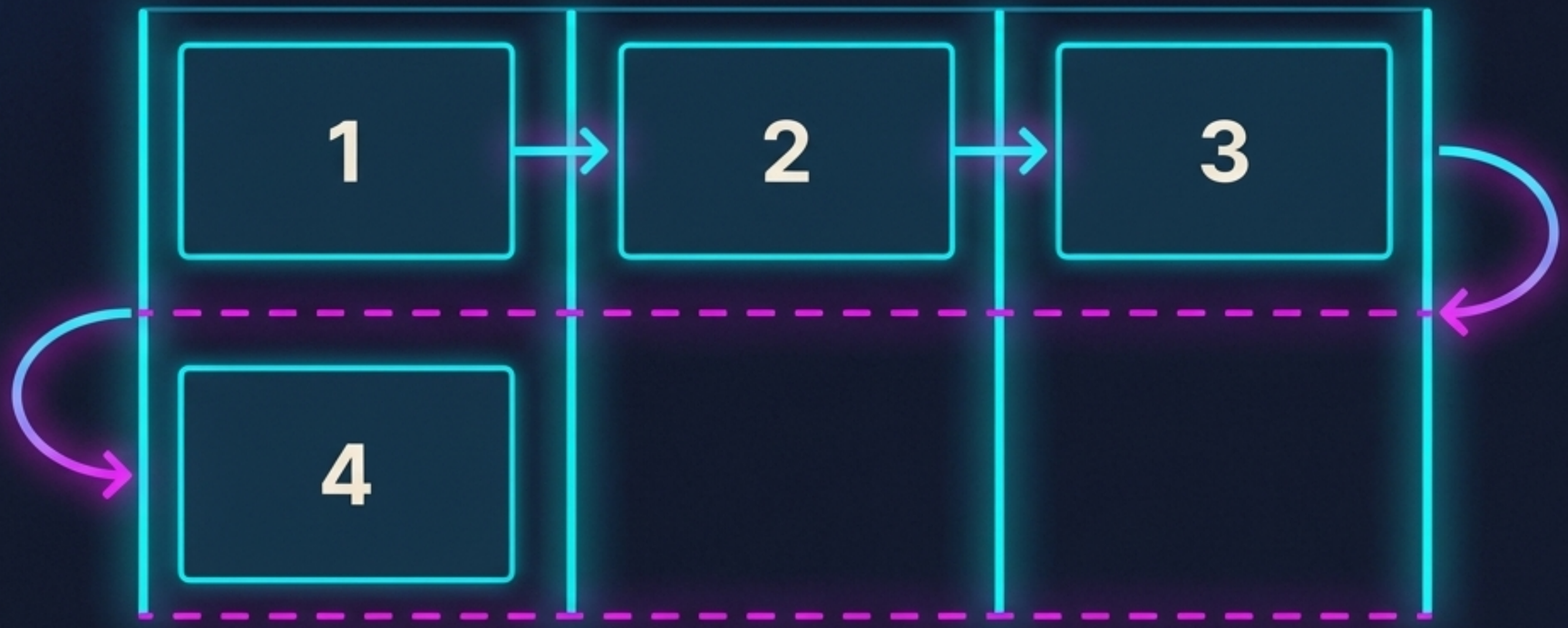


You define 3 columns. You place 12 items.
Grid creates as many rows as needed to fit them.

NotebookLM

Auto-Placement: The Default Flow

If you don’t place items manually, Grid auto-places them from left to right, then wraps to the next available row.



“That’s why it feels like ‘a smarter inline-block system’.”

NotebookLM

By default, implicit rows are `auto` height.

This means their height is determined by the content inside them, which can lead to an inconsistent layout.


```
/* Default behavior */
grid-auto-rows: auto;
```

NotebookLM

You control the implicit blueprint with `grid-auto-rows`.

Use this property to set a consistent size for every implicit row Grid creates.

Before

After


```
/* Your command */
grid-auto-rows: 10rem;
```

Before

After

NotebookLM

Putting It All Together

Explicit tracks are what you define. **Implicit tracks** are what Grid creates to finish the job.

Explicit Row

Implicit Rows
(sized by `grid-auto-rows`)

Auto-placement fills rows left-to-right, then adds new rows as needed.

NotebookLM

Implicit Columns Exist Too

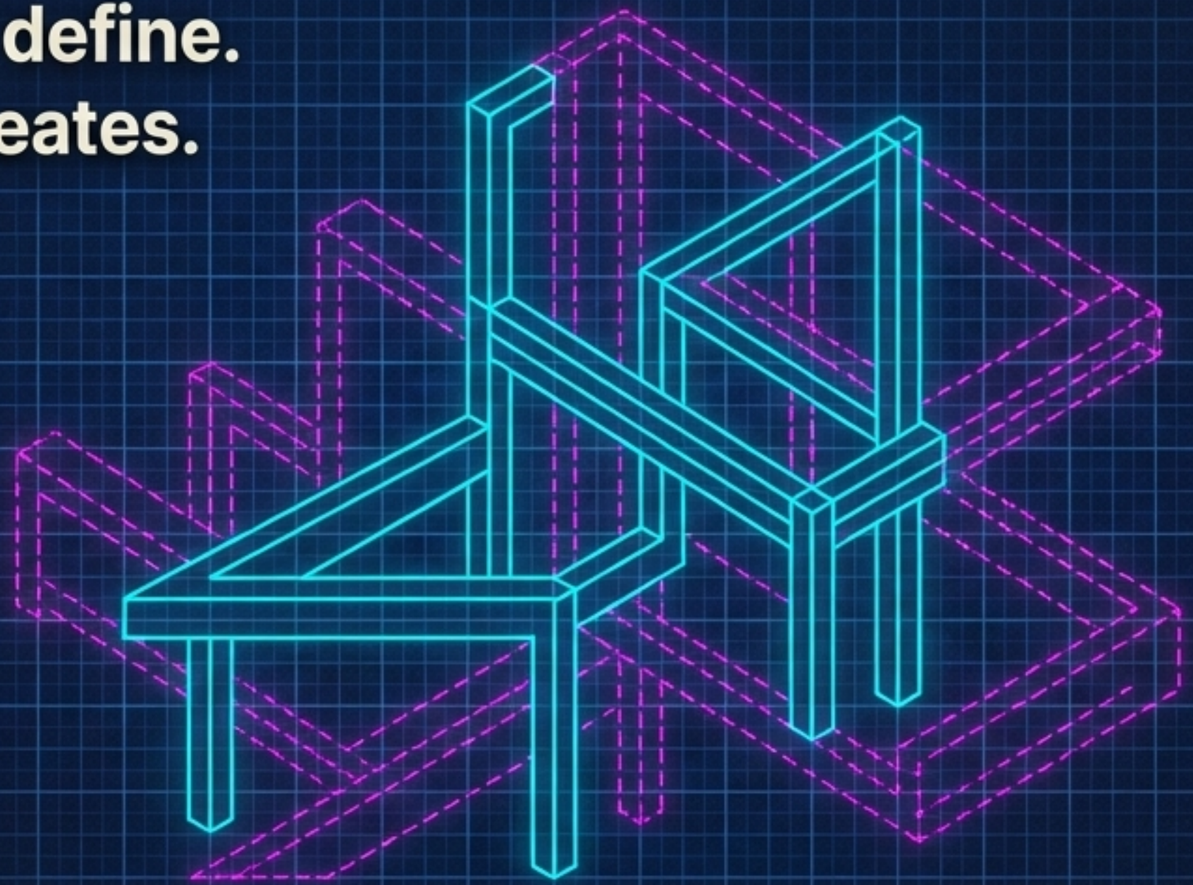
If your auto-placement flow is column-based, Grid can create implicit columns as well. The same principle applies, but the property to control them is `grid-auto-columns`.

We'll keep that one holstered for later, but it's the same idea.

NotebookLM

The Grid always has two layers.

The structure **you define.**
The structure **it creates.**



Learn both, and **auto-placement** stops being “mysterious”.

NotebookLM

p.s., keep learning!

NotebookLM