

`/* grid-template-columns: repeat(auto-fit,minmax(100px,200px)); */`

`/* use either auto-fill or auto-fit */` differnec between them

The difference between `auto-fill` and `auto-fit` in the CSS grid property `grid-template-columns: repeat(auto-fill, minmax(100px, 200px));` (or `auto-fit`) lies in how they handle empty grid tracks when there is extra space.

`auto-fill`

- It **fills the row with as many columns as possible**, even if some of those columns end up being empty.
- This means if the container's width allows space for 5 columns, it will create 5 columns, **regardless of how many actual grid items there are**.
- Empty columns remain visible and take up space.

`auto-fit`

- It also tries to fit as many columns as possible.
- However, **empty columns collapse**, meaning if there are fewer grid items than potential columns, the extra columns do not take up space.
- This allows the existing items to expand and fill the available space.

Visual difference example

Assuming you have 3 grid items and the container could fit 5 column slots:

- With `auto-fill`, you get 5 columns sized 100–200px each, including 2 empty columns.
- With `auto-fit`, you only get 3 columns sized to fill available space, no empty columns.

Summary

keyword	Behavior with extra space and empty columns
<code>auto-fill</code>	Creates full number of columns, empty columns remain visible
<code>auto-fit</code>	Creates columns only for content, empty columns collapse

Use `auto-fill` if you want consistent track count on the grid line regardless of content count.

Use `auto-fit` to have grid items stretch and avoid empty gaps.