Table of Contents

[1. **Grid css** 2](#_Toc80028932)

1. **Grid css**

Container:

Lun phải có **display:grid**

Có **grid-template-columns**,

Có thể có thêm **grid-template-rows**



**Grid-template-columns**:

* **repeat**: lập lại các giá trị cho các phần tử theo cột.
* **Auto-fit**: tự động tính width để fix với các màn hình.
* **Minmax**: input1: số min: width **ko thể nhỏ hơn** , input2: số max width **ko thể lơn hơn**

(nếu **width màn hình** < **tổng width của các div** thì sẽ xuống hàng)

Vd: 4div, min: 250px.

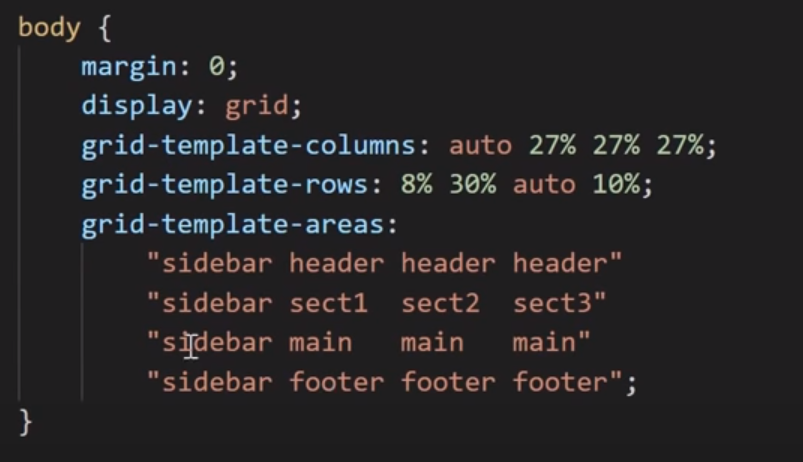
Nếu màn hình từ 1000px trở lên thì 4 div trên 1 row.

Nếu dưới 1000px thì 1 div sẽ rớt xuống row dưới.

Nếu dưới 750px thì 2 div rớt.

Nếu dưới 500px thì 3 div rớt xuống row dưới.

Nên auto-fit sẽ giúp các div lấp khoảng trống của row đều nhau.



**Grid-template-areas**:

Để tạo ra 1 cái bản đồ cho cái layout.

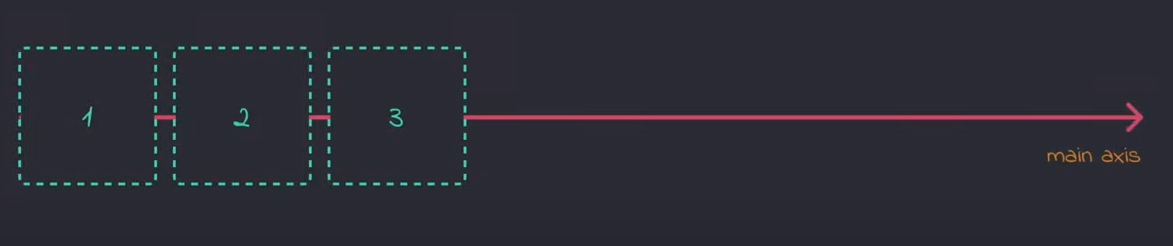
Nhìn rõ ràng, nhưng phải định nghĩa thêm **grid-template-rows** nữa, nên chỉ phù hợp với layout tổng(layout cố định các section)

Khi xuống mobile thì phải định nghĩa lại template này

So sánh flex vs grid css

**Flex**

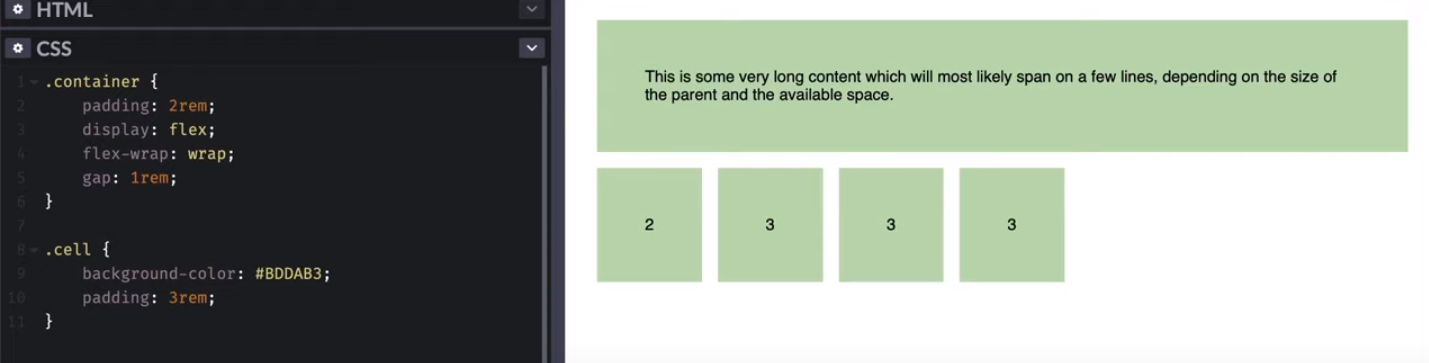
1 direction



Nên muốn đổi hưởng thì phải dùng flex-direction: row/coloumn

Flexbox cũng có thể dùng trong grid system.

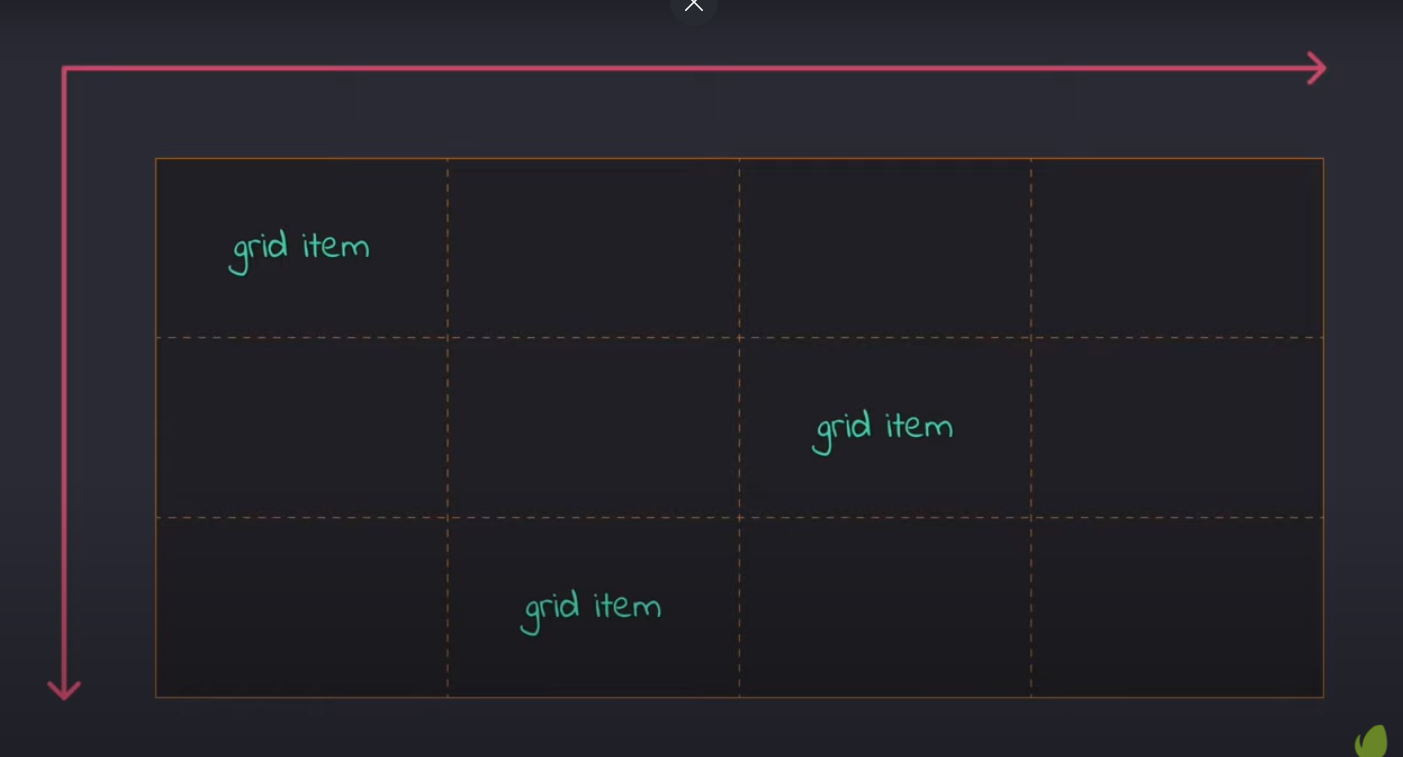
**Content shapes the layout** – nội dung định hình layout



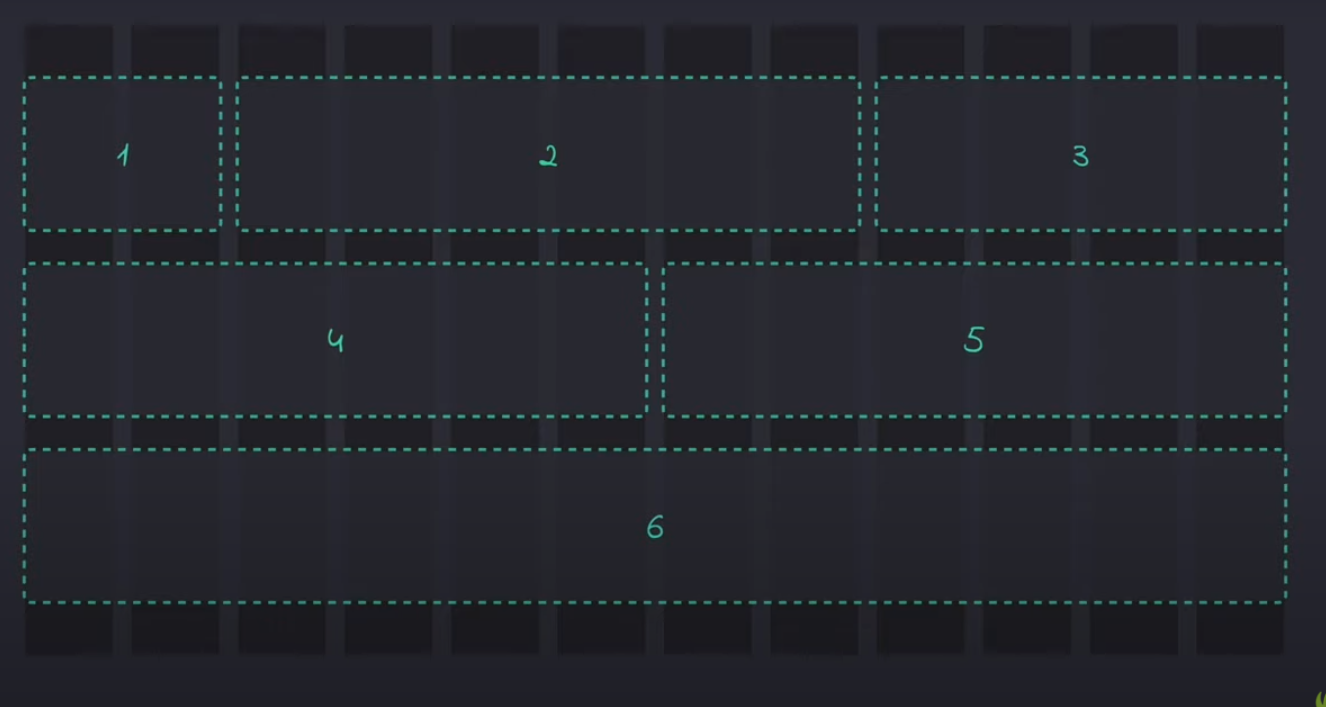
Nội dung được chạy từ trái sang phải cho full width.

Flex-box ko thể responsive được (phải dùng @media) , nó chỉ wrap các items ở trong thôi. Khi ko width ko đủ chỗ thì flex-box wrap sẽ tự động xuống hàng. **grid**

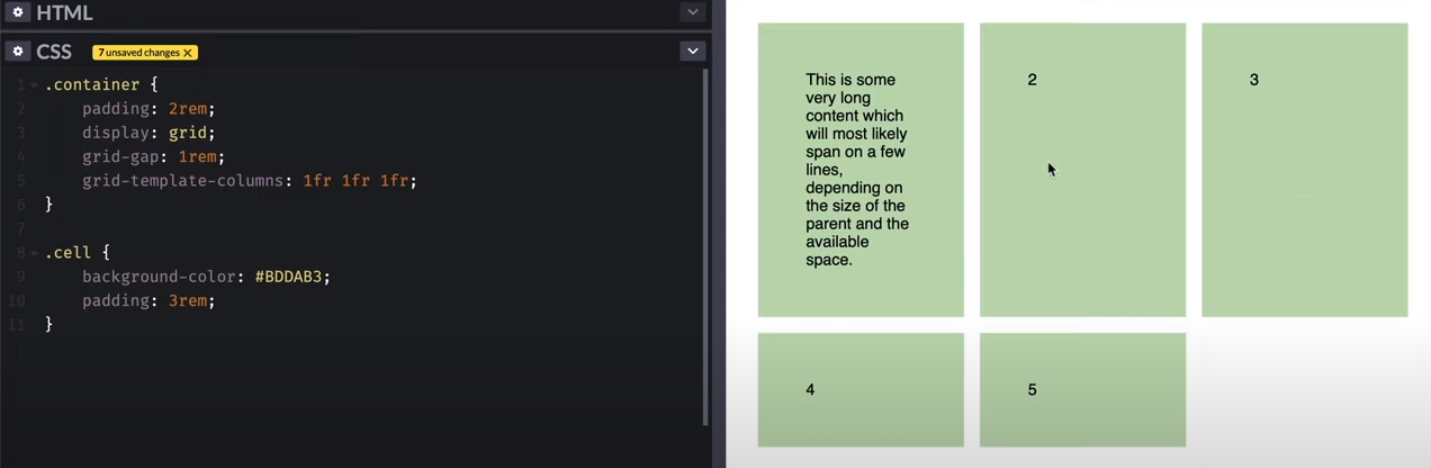
2 directions



12 coloumns grid in grid system



**Layout shapes the content** – layout định hình cho nội dung



Nội dung chỉ chạy ở trong 1 section mà grid đã định nghĩa, ko chạy hết width

Grid-css có thể tự responsive dc(ko dùng @media) với thuộc tính **auto-fit/auto-fill**.