



## Day 26

# “Web Development + Security”

## CSS Grids:

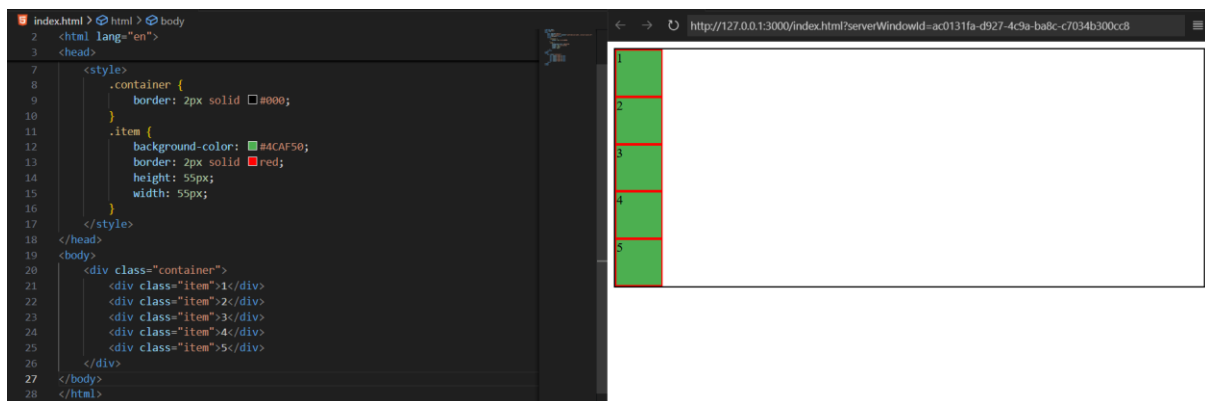
### What is CSS Grid?

CSS Grid allows you to divide a container into rows and columns and place items precisely in these rows and columns.

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

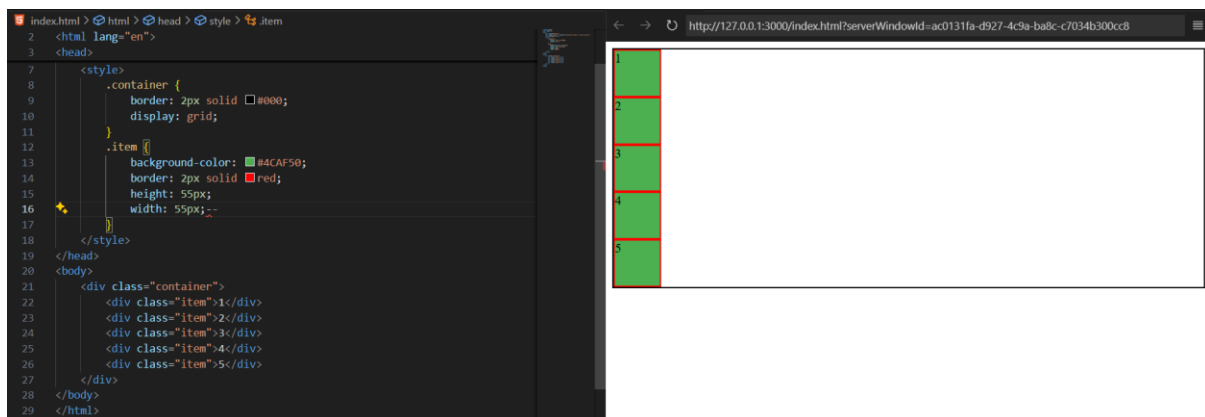
All direct children of .container become grid items.

Let's first create 5 <div> inside one parent <div> with some basic CSS from our end:

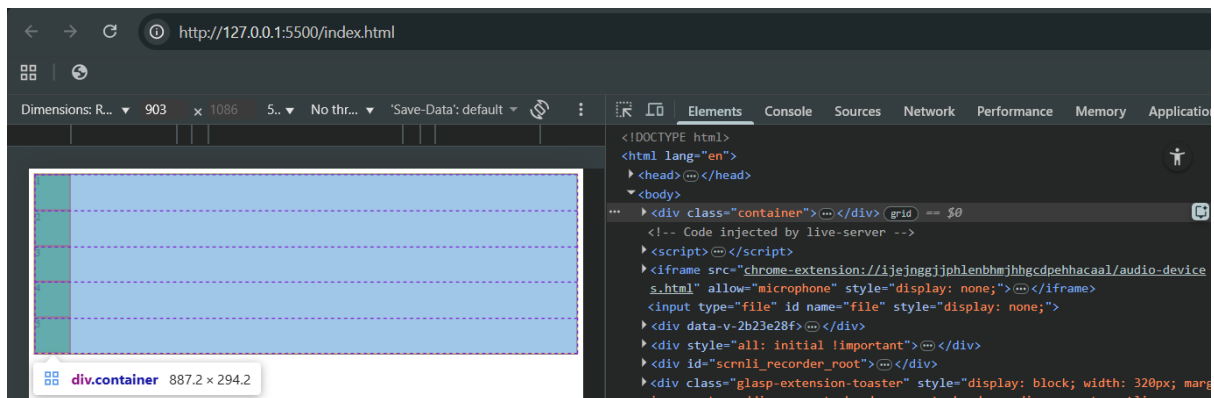


Clearly, we have not used the display:grid yet, but what if we do so? Then also, everything be as it is, but when you inspect, you will see that each of the box will be treated as row and columns. So, basically it divides the containers into rows and columns.

For example:

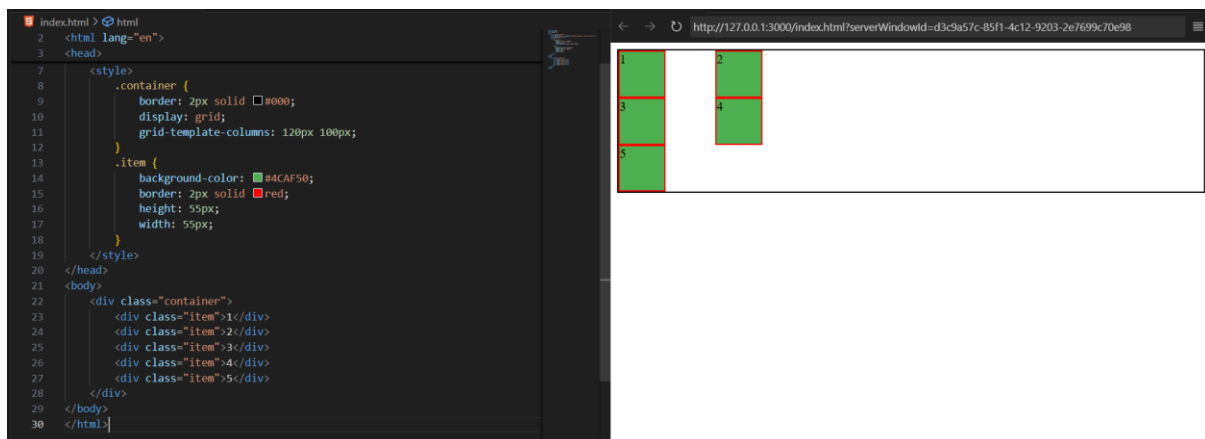


Inspect:

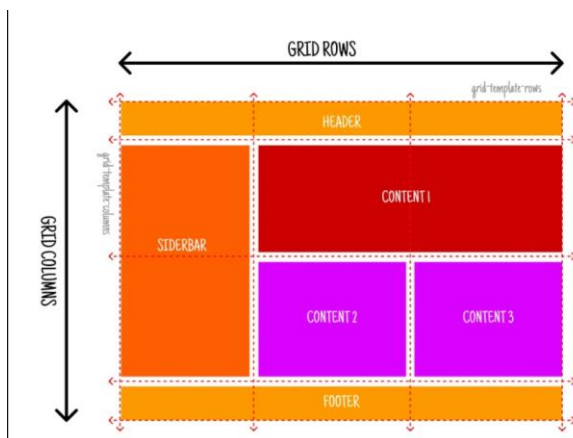


Now, we will see how to set the column width which we get by changing the display to grid, for which we will do, `grid-template-columns: 120px 100px`.

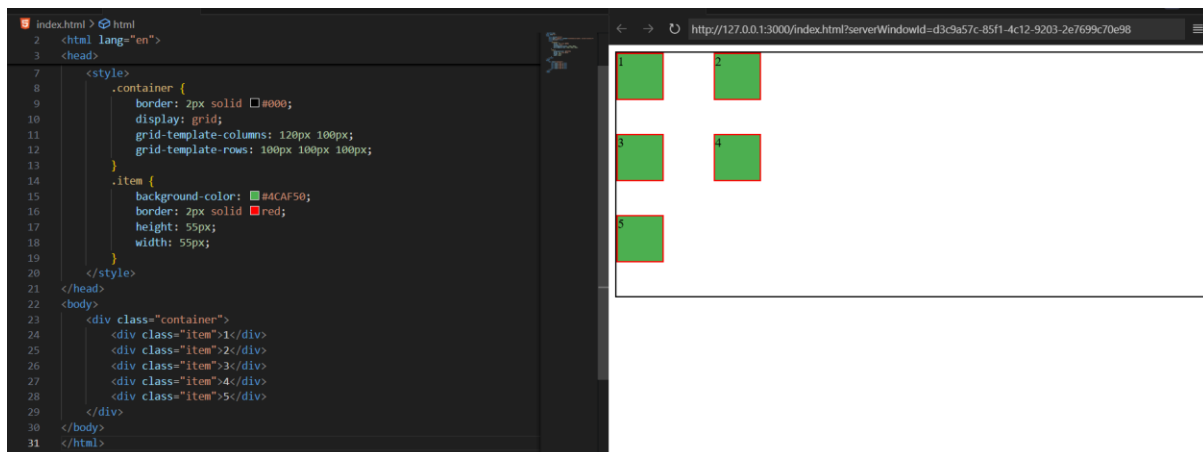
For example:



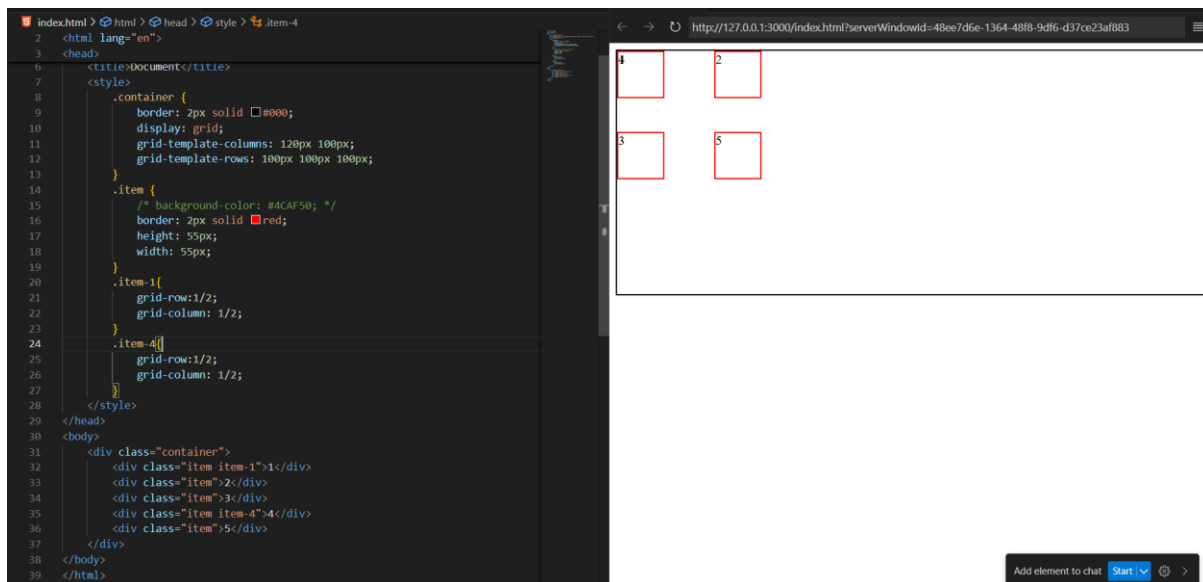
Now, concept of grid rows and grid columns:



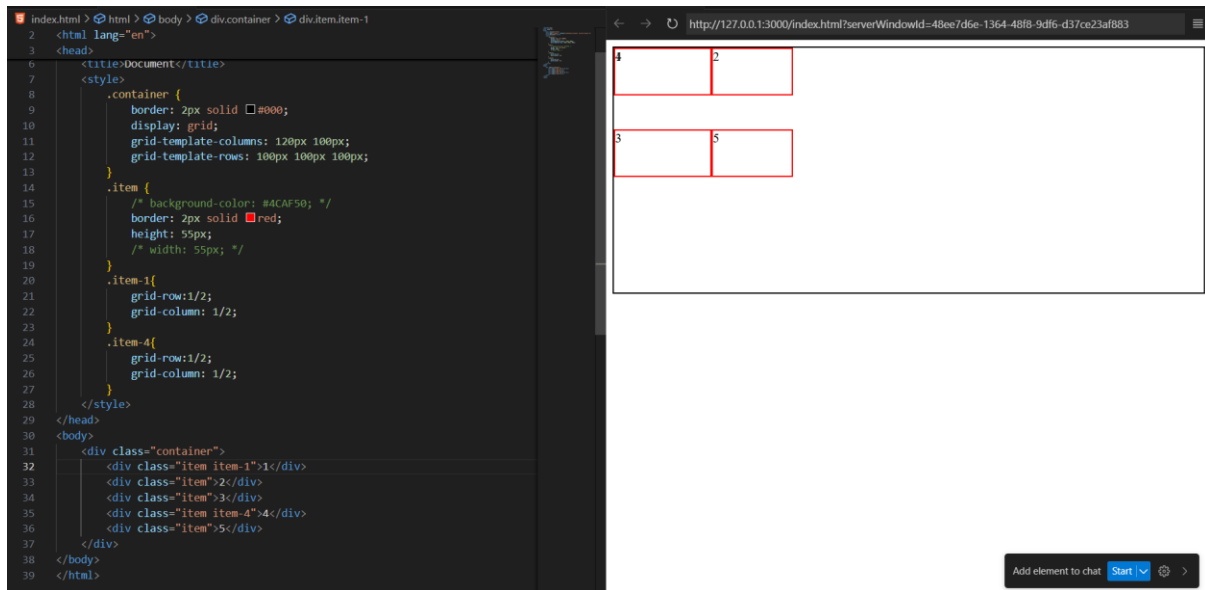
Example: showing use of grid-template-rows:



Now, we can even position the child div using the class and grid-row and grid-column property for the child element: in the below code we asked that the item whose class is item-4 and item-1 should be between grid number 1 to 2, as shown below: clearly you can see the overlap

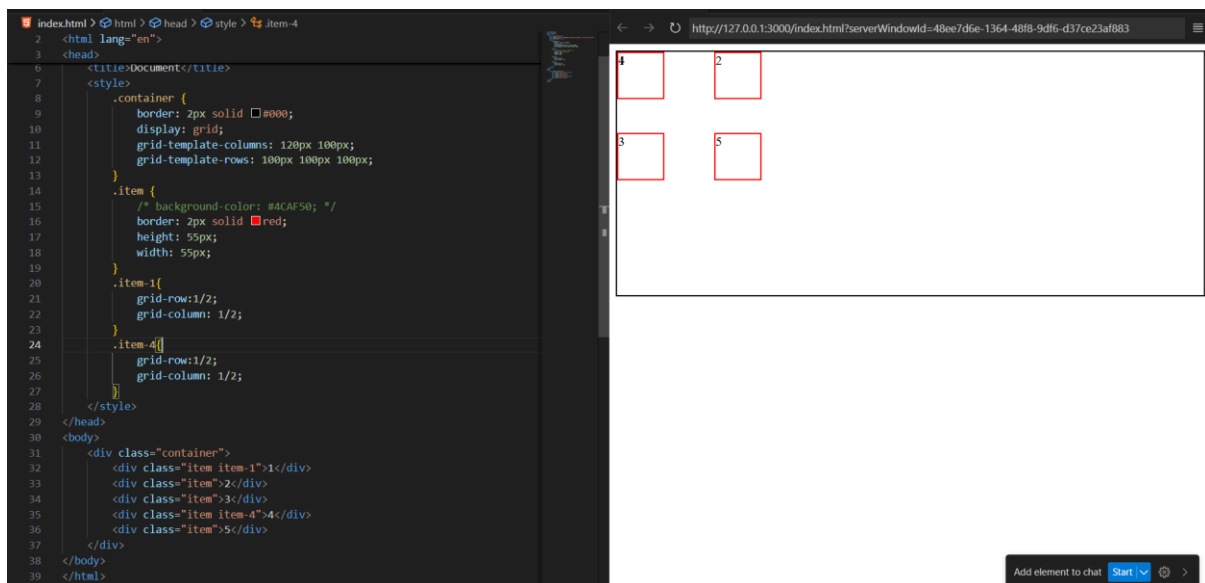


In case we don't specify the width of the items, we can see that those boxes even stretched themselves to occupy the full width of that column. As shown below:

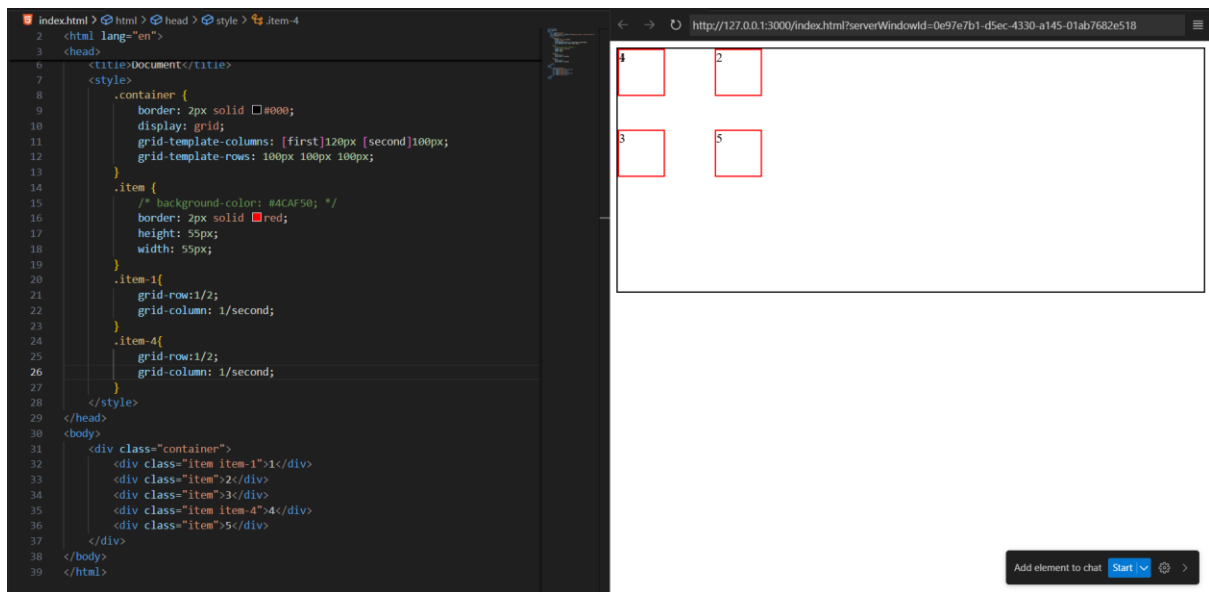


Example: we can even name the columns in the parent class and then tackle the child to expand with the name:

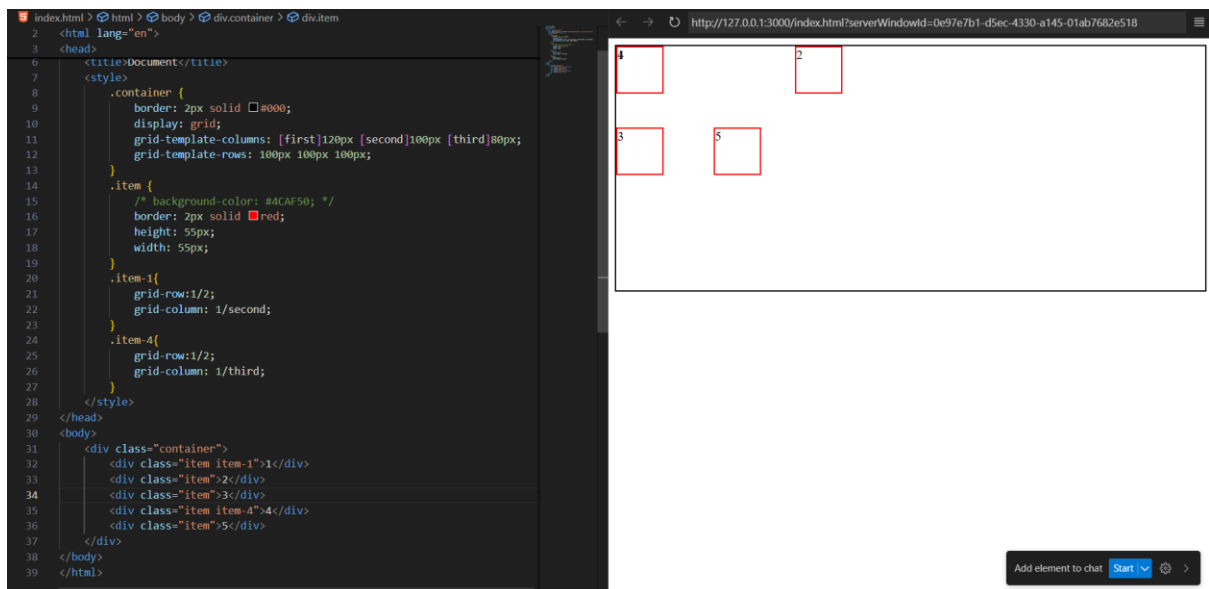
Without name:



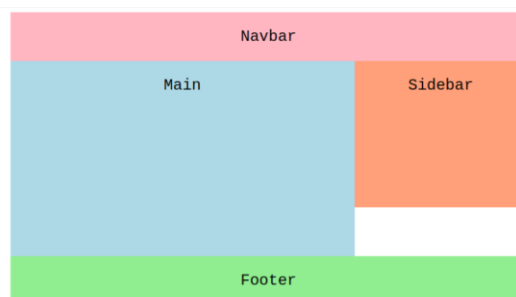
With name:



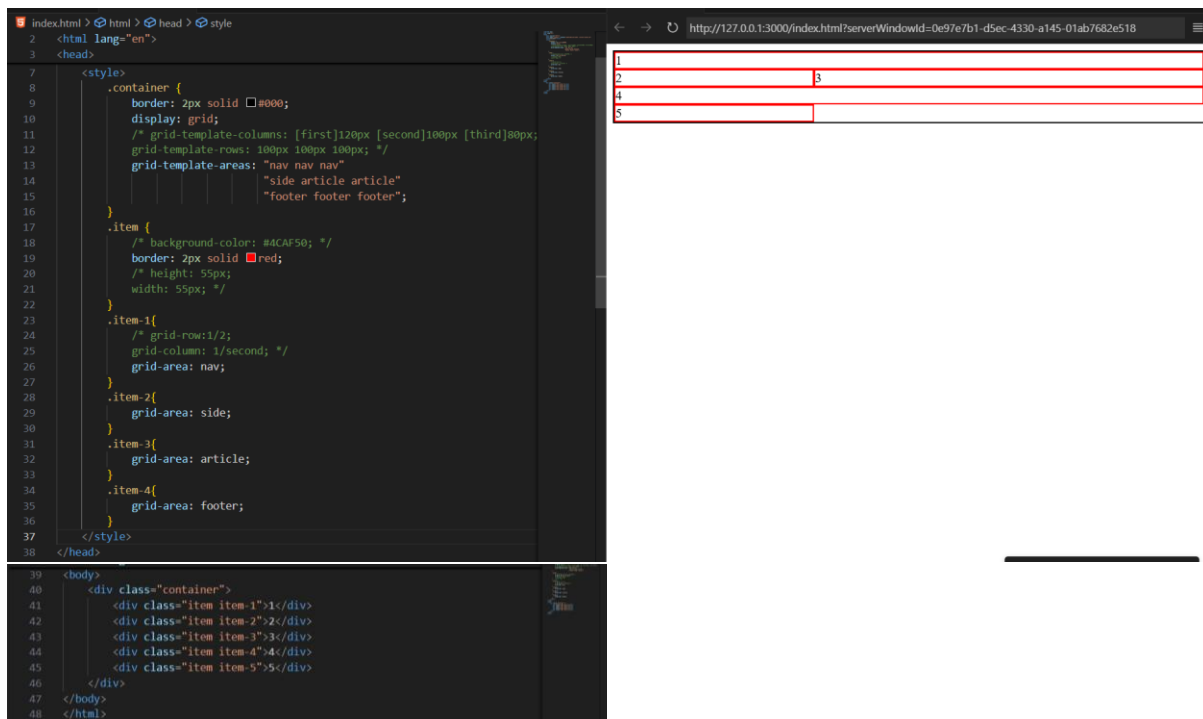
Now, in case we our expansion touches the columns where the boxes are already there then the boxes will get pushed below:



Now, grid template are:



Example:



What is fr?

- fr stands for fraction of available space in a CSS Grid container.
- It allows you to divide the space proportionally without using fixed pixels.

Think of it as:

“Take the available space and split it into fractions.”

How fr works?

Example:

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

Explanation:

- Total space = 1 + 2 + 1 = **4 parts**
- Column widths:
  - First column → 1/4 of available width
  - Second column → 2/4 (half)
  - Third column → 1/4 of available width

So the middle column is twice as wide as the side columns.

### Key Points About fr:

1. fr **ignores padding/margin**; it only splits **remaining space**.
2. Can mix fr with **fixed units** (px, %, etc.):

grid-template-columns: 200px 1fr 2fr;

- First column = 200px fixed
  - Remaining space = split into 1:2 fraction for the other two columns
3. Works **only in Grid**, not Flexbox. (Flexbox has flex-grow which is similar)

--The End--