In element that is your container for the grid:

***Selector*** {

**display: grid;**

**grid-template-columns: 20% 20% 20% 20% 20%** ;(where you divide width of page into the number of columns you want and the % those columns should take up in the grid. In this example there will be 5 columns with equal widths:

**grid-template-rows:** **20% 20% 20% 20% 20%** ;( where you divide width of page into the number of columns you want and the % those columns should take up in the grid. In this example there will be 5 columns with equal widths:

}

|  |  |  |
| --- | --- | --- |
|  | Command | description |
|  | grid-column-start: # | Element will start in grid number specified |
|  | grid-column-end: # | Element will span from the grid noted in grid-column-start to the # noted in grid-column-end  Note:   * grid-column-start can be greater than grid-column-end. In this case the element will span from right to left * you can also use a negative number to specify how many grid lines the element should start or end from the right * instead of defining the start and end you can use span # to indicate how many columns you want the element to span. Can only be used with positive numbers   + grid-column-end: span 2 will span two grid lines to the right.   + grid-column-start: span 3 will extend your element 3 grid lines from the right) |
|  | grid-column: # / # | Is shorthand for grid-column-start and grid-column-end. The two numbers represent the start and end of the span. Can also use span # in place of the number. If you use only one number (without the slash) it assumes that number is both the start and the end. |
|  | grid-row-start: # | Works just like grid-column-start except it affects the y axis |
|  | grid-row-end: # | Works just like grid-column-end except it affects the y axis |
|  | grid-row: # / # | Works just like grid-column except it affects the y axis |
|  | grid-area: # / #/ #/ # | Short cut that allows you to span across an area by defining : grid-row-start/grid-column-start/ grid-row-end/grid-column-end |
|  | Order: # or -# | By default all grid items have an order of 0 – they show up in the order they are in the code. To change the default order, you can use order to move this to the specific grid you want. Negative number moves to the left. |
|  | Grid-template-columns: repeat (# columns, % width of columns) | Shorthand to specify the number and width columns. Instead of saying grid-template-columns: 20% 20% 20% 20% 20%; you can say grid-template-columns: repeat (5, 20%)  Can also use pixels and ems instead of % |
|  | grid-template-columns: fr# | Where the # is the fraction of space taken up by the column: A page where one element spans 1/6 of the width and the second element takes up the rest you would say grid-template-columns: 1fr 5fr; e.g 1/6 and 5/6 |
|  | grid-template: | Combines grid template rows and column commands using a shortcut. For example grid-template: 50% 50%/ 200px two rows 50% plus one column that is 50% wide |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |