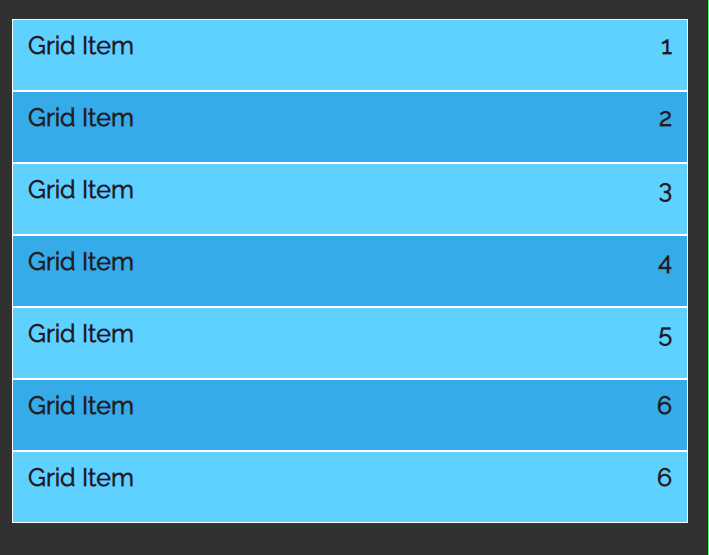
CSS GRID

{

display: grid

}



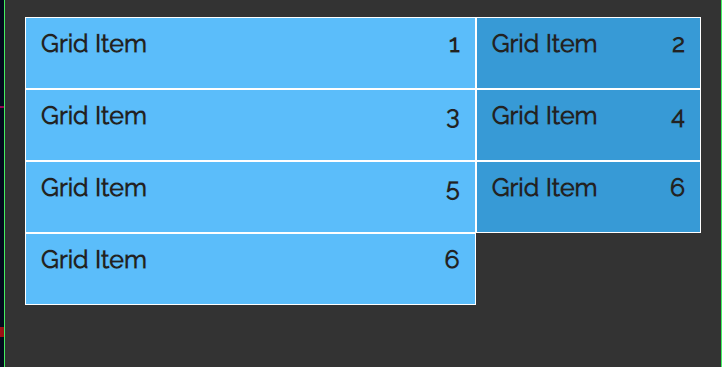
*/\* 200 px columns on left and 100px column on right \*/*

grid-template-columns: 200px 100px;



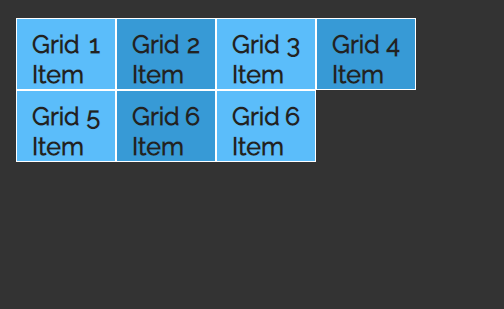
*/\* to flexibility, 2fr means twice the size of column and 1fr \*/*

grid-template-columns: 2fr 1fr;



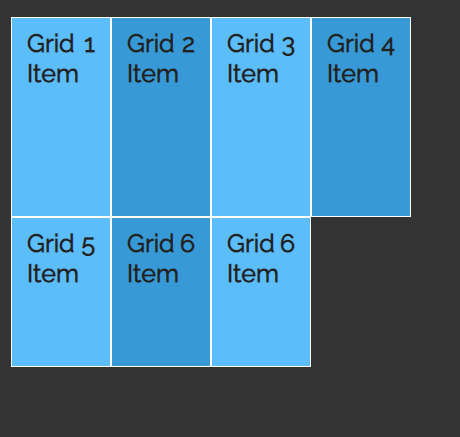
*/\* repeating columns size 4 times of 100px \*/*

grid-template-columns: repeat(4, 100px);



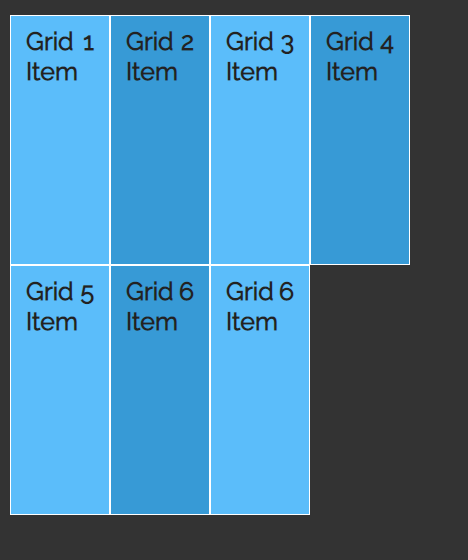
*/\* the first rows will be in size 200 px and 2nd row will be in 150px \*/*

grid-template-rows: 200px 150px ;



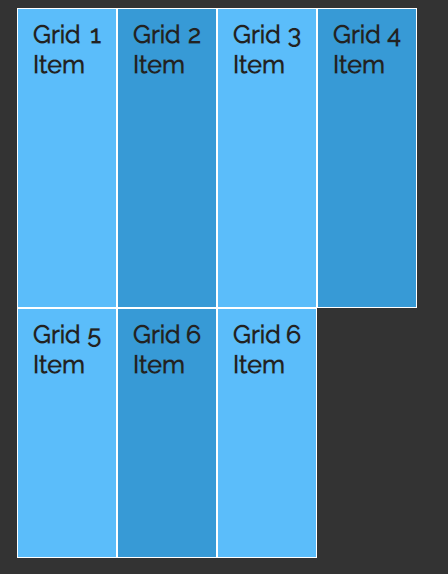
*/\* grid auto row- it will automatically resize the rows to the given values\*/*

grid-auto-rows: 250px;



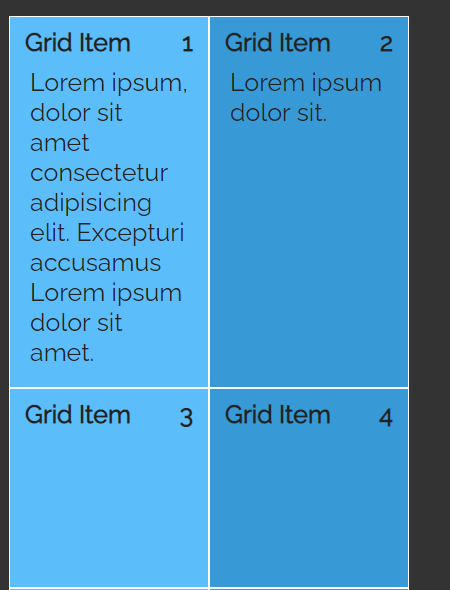
*/\* If we give the size of first column as 300px, the other rows are still 250px (as mentioned in grid-auto-row) \*/*

grid-template-rows: 300px;



*/\* to size the rows according to the content \*/*

*/\* It indicates the minimun height of the row will be 150px and maximum height will be auto, to the content display \*/*



*/\* Instead using row-start and row-end , we can use span , similarly ro column-start and column-end \*/*

.grid-item-4{

grid-row: span 2;

}

.grid-item-5{

grid-row: span 2;

}

**MDN link to learn CSS Grid :**

[*Basic Concepts of grid layout - CSS: Cascading Style Sheets | MDN*](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/Basic_Concepts_of_Grid_Layout)