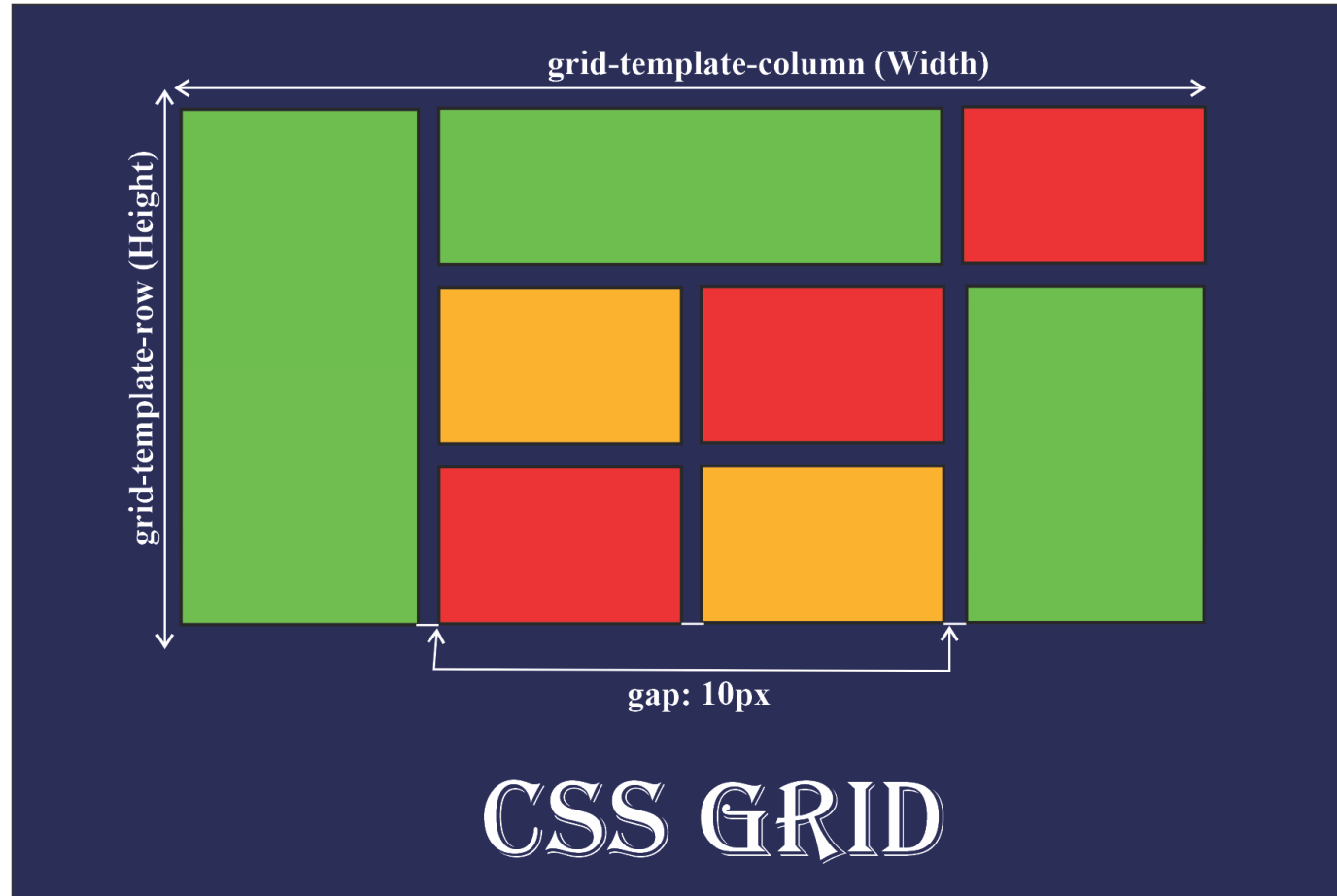


CSS GRID THEORY

Video to watch before session



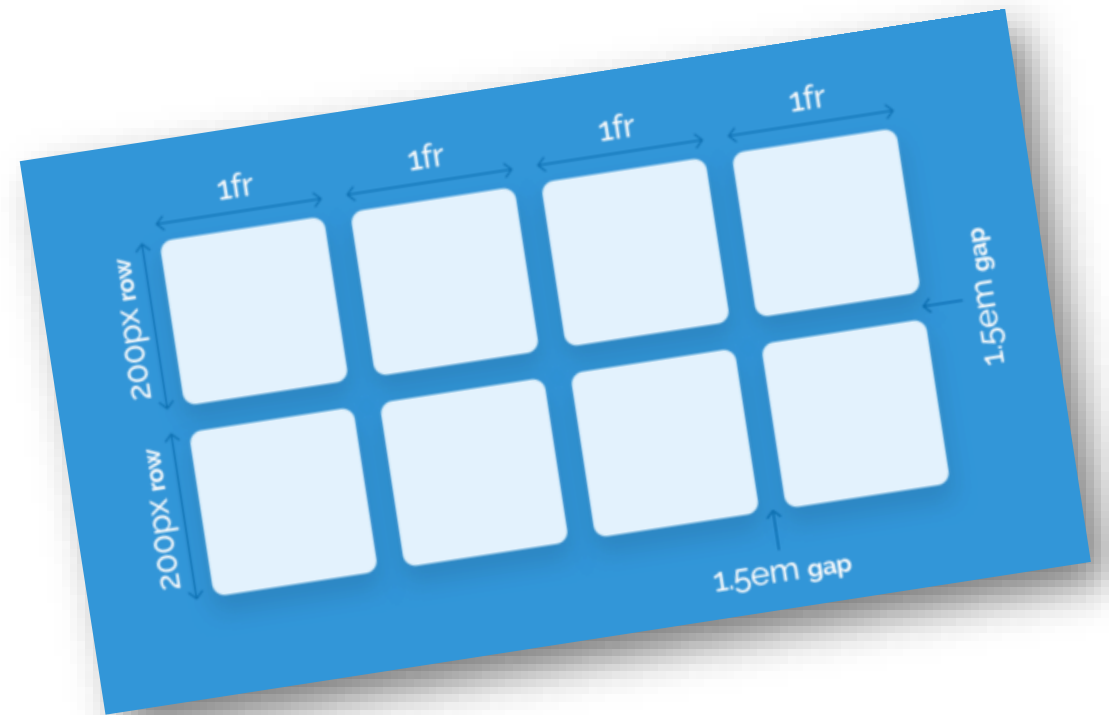
<https://www.youtube.com/watch?v=9zBsdzdE4sM>



OBJECTIVES FOR TODAY



- ✓ Grid Terminology
- ✓ Create the Grid Template
- ✓ Naming Grid Areas
- ✓ Properties of grid container:
 - `grid-template-columns`
 - `grid-template-areas`
 - `grid-area`
 - `gap`





05 MIN



How CSS Grids Work

- ✓ Set an element's display property to create **grid container**. Its children would become **grid items**.
- ✓ Set up the **columns** and **rows** for the grid.
- ✓ Assign each **grid item** to an area on the **grid**.



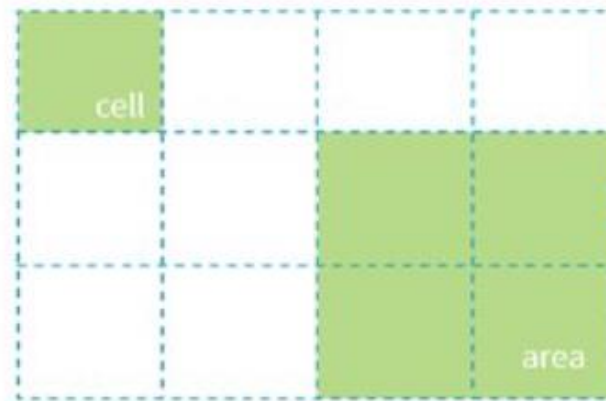
05 MIN



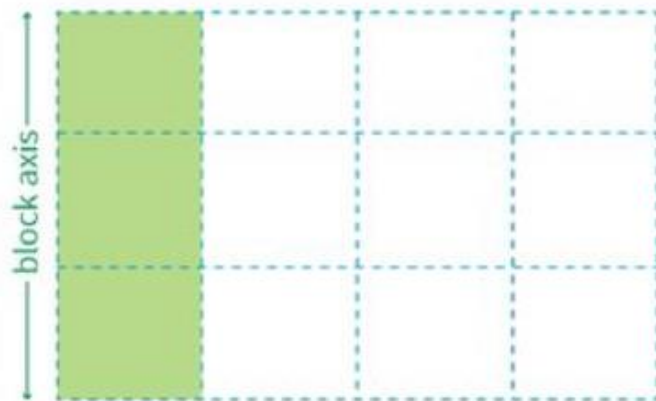
Grid Terminology



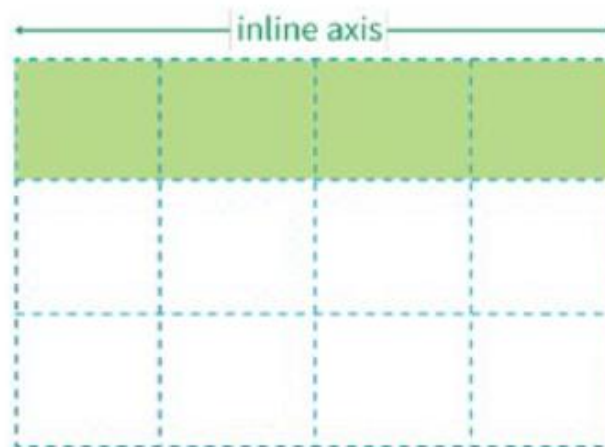
grid lines



grid cell and grid area



grid track (column)



grid track (row)



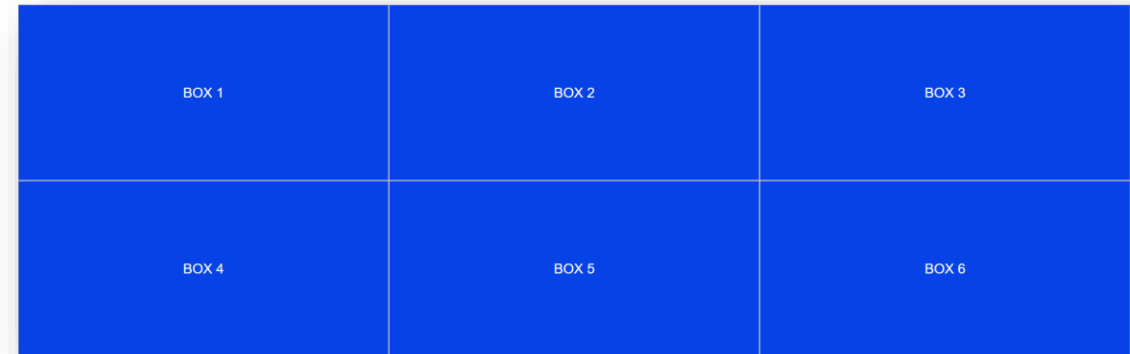
05 MIN



Creating a grid container

To make an element a **grid container**, set its **display** property to **grid**

All of its children automatically become grid items



```
<div class="container">
  <div class="box">Box 1</div>
  <div class="box">Box 2</div>
  <div class="box">Box 3</div>
  <div class="box">Box 4</div>
  <div class="box">Box 5</div>
  <div class="box">Box 6</div>
</div>
```

```
.container {
  display: grid;
  grid-template-columns: 1fr 1fr 1fr;
}
.box {
  height: 200px;
  border: 1px solid #ccc;
  background: #0642e6;
  color: #fff;
```

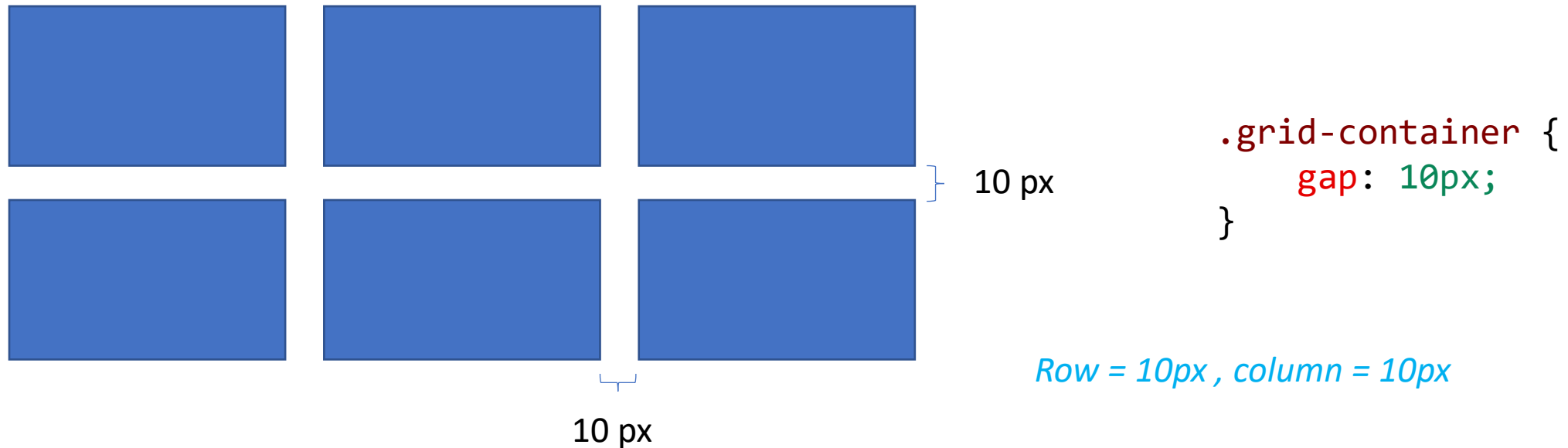


05 MIN



Grid Gap

To make the space in grid layout we can use **gap** or **margin**.



The **value** of the **gap** here is used in **both rows** and **columns**.
It means we set the **row-gap = 10px** and the **column-gap = 10px**.

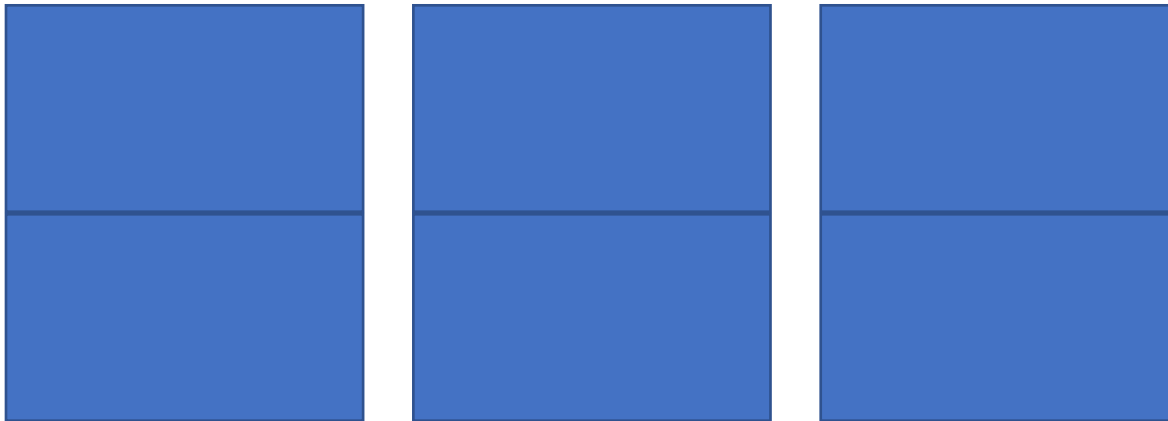


05 MIN



Grid Gap

To make the space in grid layout we can use **gap** or **margin**.



10 px

```
.grid-container {  
  gap: 0 10px;  
}
```

Row = 0px , column = 10px

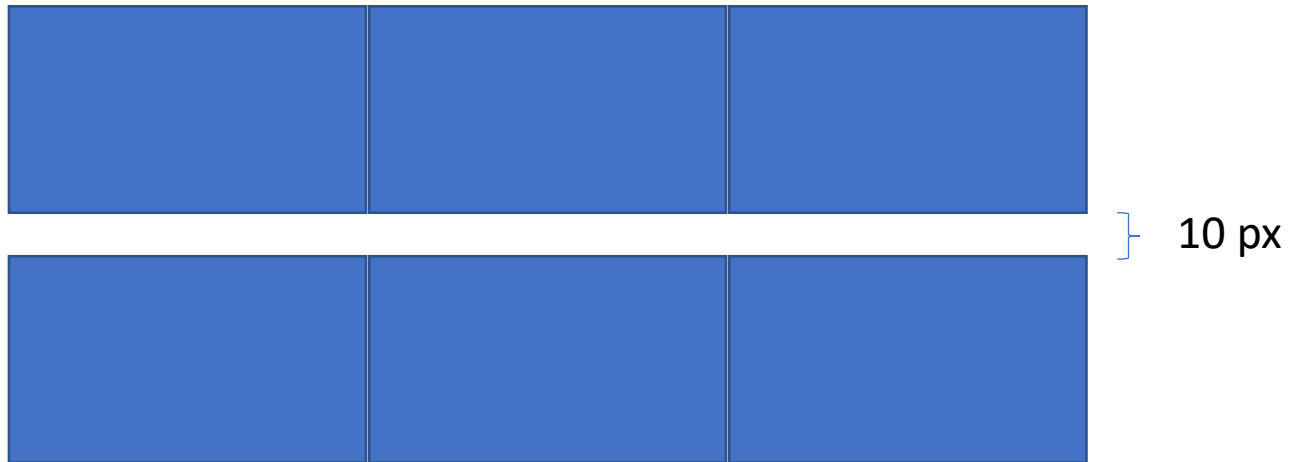


05 MIN



Grid Gap

What is the **value** of **gap** here?



```
.grid-container {  
  gap: ?? ??;  
}
```

Row = ??, column = ??

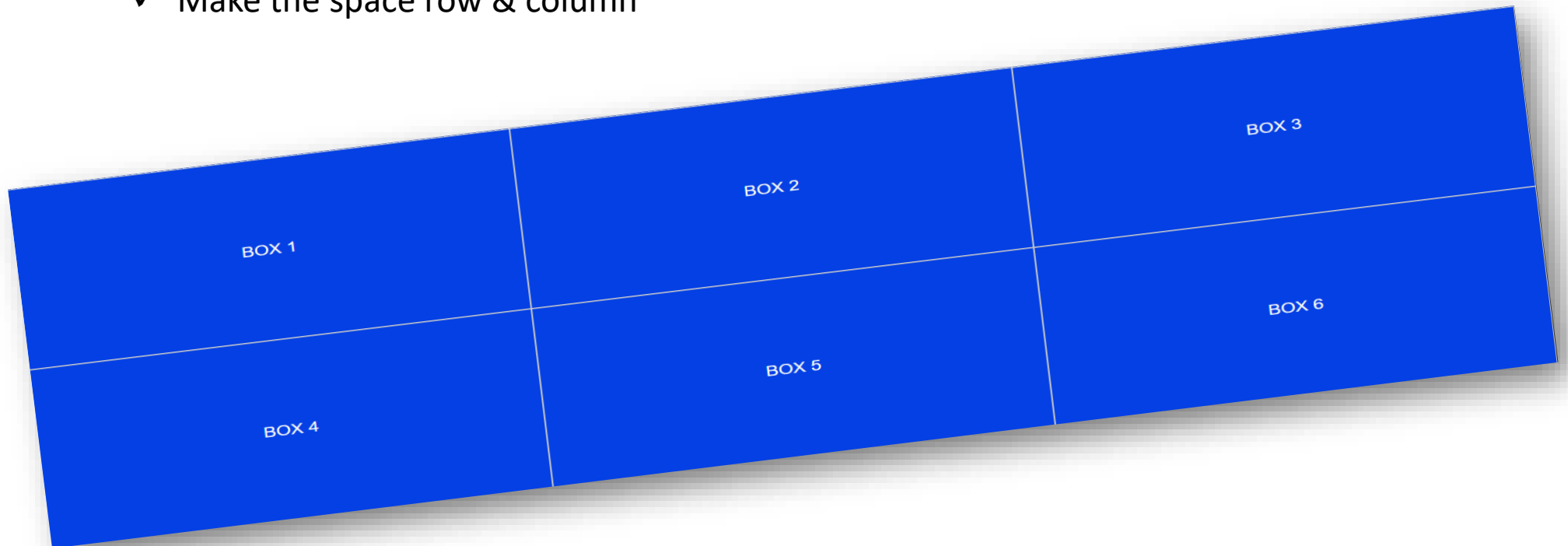


10 MIN

Activity 1

Define the layout like this picture with HTML that teacher provide
Using **gap** properties

- ✓ Make space only row
- ✓ Make space only column
- ✓ Make the space row & column





05 MIN

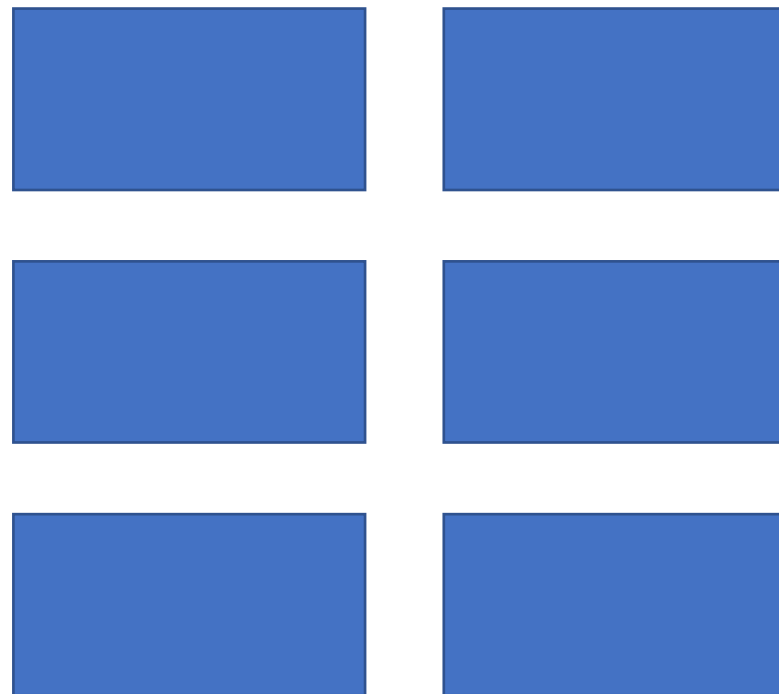


Grid Template Column

```
.grid-container{  
  display: grid;  
  grid-template-columns: 50% 50%;  
}
```

```
.grid-container{  
  display: grid;  
  grid-template-columns: auto auto;  
}
```

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



CSS Grid Layout introduces a new **unit of length** called fr, which is short for “**fraction**”. We can use any unit in CSS like **px**, **em**, and **rem** but we suggest: **fr** is easy!



05 MIN

Grid Template Column

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



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



OR

```
.grid-container{  
  display: grid;  
  grid-template-columns: repeat(4, 1fr);  
}
```





05 MIN



Grid Template Column

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





20 MIN



Activity 2

Define the layout like this picture with HTML that teacher provide
Using ***grid-template-columns***

EX 1

1fr

1fr

EX 2

2fr

1fr

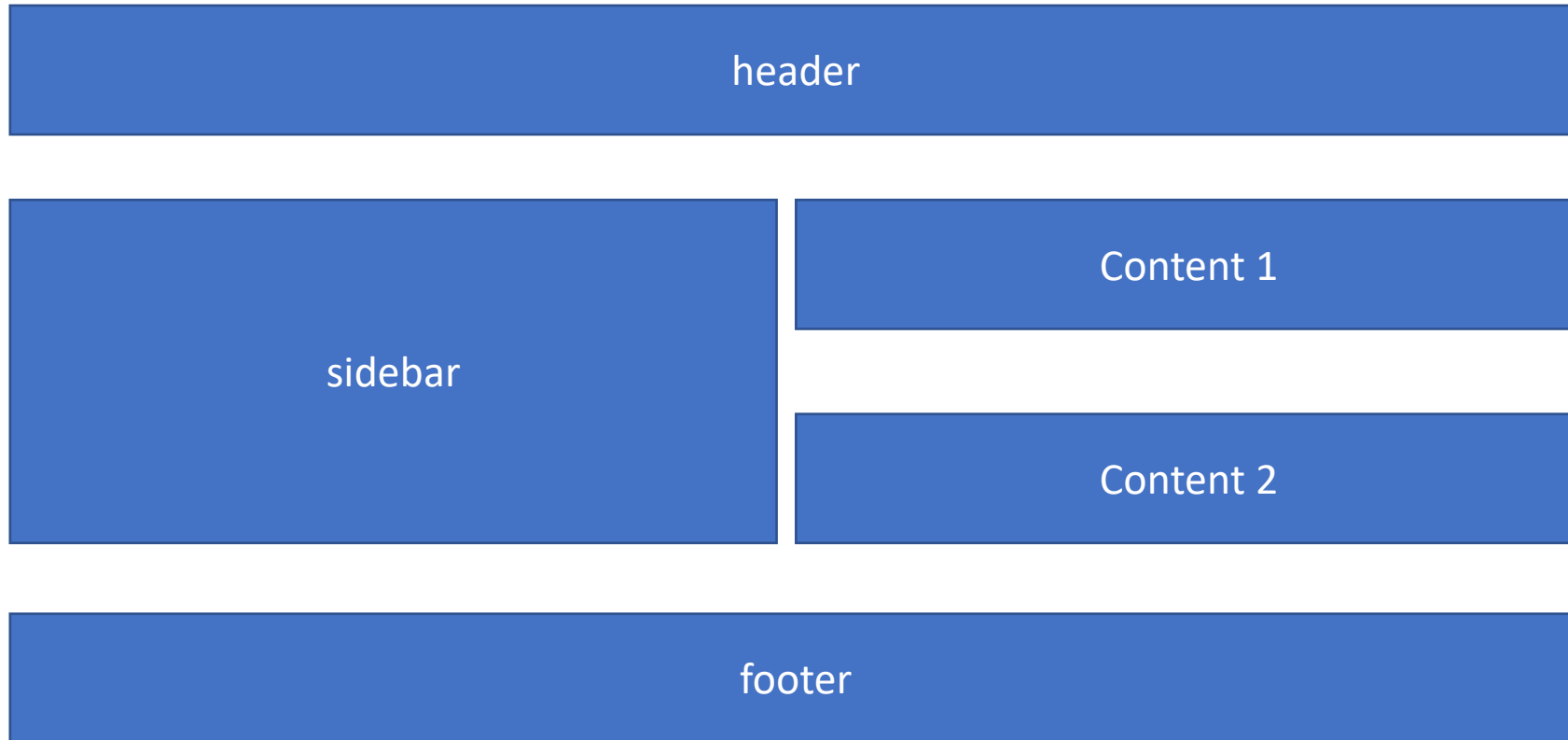
2fr



05 MIN



Grid Template Area



Grid area is the easy ways in **CSS Grid layout** to define the **column** of **layout faster**.



05 MIN

Example

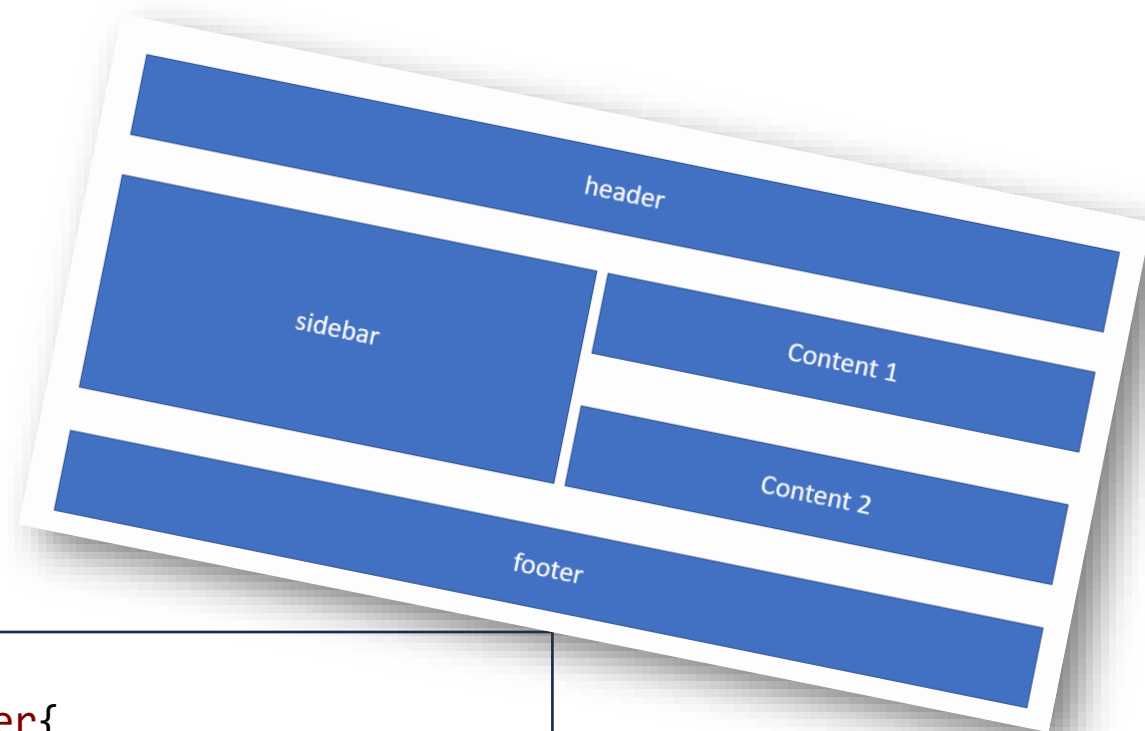
<HTML>

```
<div class="grid-container">
  <header class="header">Header</header>
  <aside class="sidebar">SideBar</aside>
  <div class="content1">Content 1</div>
  <div class="content2">Content 2</div>
  <footer class="footer">Footer</footer>
</div>
```

<CSS>

```
.grid-container{
  display: grid;
  grid-template-columns: repeat(2,1fr);
  grid-template-areas:
    'header header'
    'sidebar content1'
    'sidebar content2'
    'footer footer';
}
```

```
.header{
  grid-area: header;
}
.sidebar{
  grid-area: sidebar;
}
.content1{
  grid-area: content1;
}
```

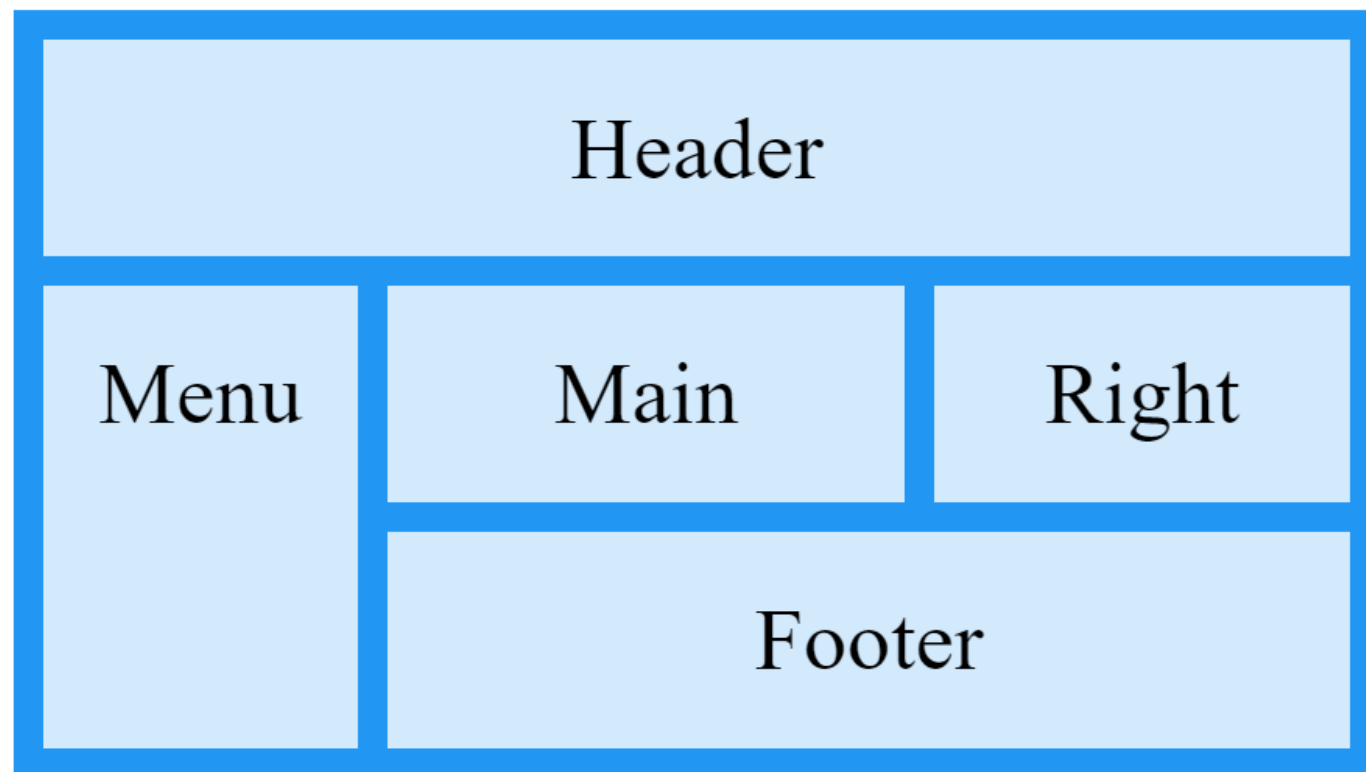




20 MIN

Activity 3

Define the layout like this picture with HTML that teacher provide
Using ***grid-template-areas***





GRID GARDEN

◀ Level 1 of 28 ▶

Welcome to Grid Garden, where you write CSS code to grow your carrot garden! Water only the areas that have carrots by using the `grid-column-start` property.

For example, `grid-column-start: 3;` will water the area starting at the 3rd vertical grid line, which is another way of saying the 3rd vertical border from the left in the grid.

```
1 #garden {  
2   display: grid;  
3   grid-template-columns: 20% 20% 20% 20% 20%;  
4   grid-template-rows: 20% 20% 20% 20% 20%;  
5 }  
6  
7 #water {  
8     
9 }  
10  
11  
12  
13  
14
```

Next



<https://cssgridgarden.com/>