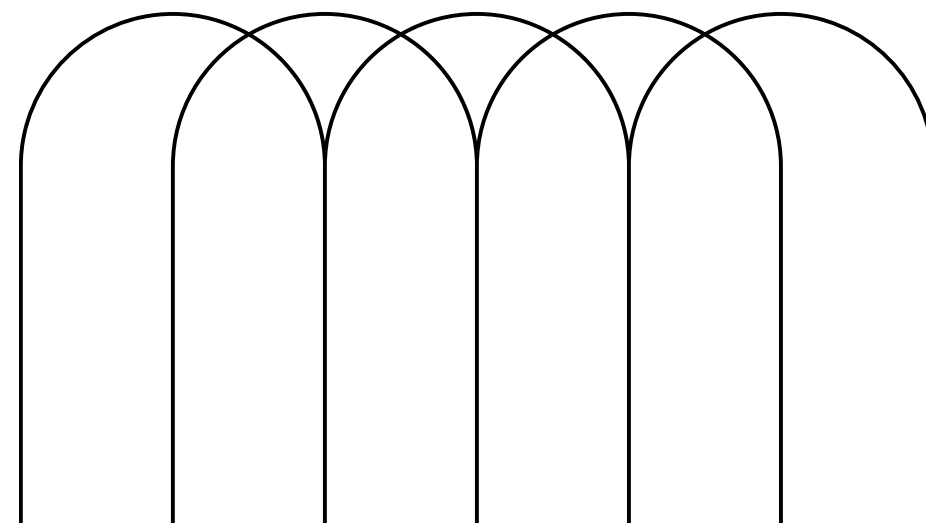


# CSS Grid



# CSS Grid

CSS Grid allows web developers to create complex and responsive designs with ease. Unlike floats and flexbox, CSS Grid enables precise control over both rows and columns, making it a two-dimensional system.

**1. Grid Container** : A grid container is the parent element that holds grid items.

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

**2. Grid tracks (Row & Columns)** : Grid tracks are the rows and columns that define the grid structure. You can specify the number of rows and columns using grid-template-rows and grid-template-columns.

```
.container {  
  display: grid;  
  grid-template-columns: 100px 200px 100px;  
  grid-template-rows: 50px 100px;  
}
```

**3. Grid Gaps** : Grid gaps (also called gutters) define spacing between rows and columns.

```
.container {  
  display: grid; grid-gap: 20px;  
}
```

# CSS Grid

**4. Fractional Units** : The fr unit represents a fraction of the available space in the grid container.

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

**5. Grid Item Positioning** : Grid items can be positioned using grid-column and grid-row properties.

```
.grid-item {  
  grid-column: 1 / 3;  
  grid-row: 1 / 2;  
}
```

**6. justify & Align** : These properties control alignment inside the grid container and items.

```
.grid-item {  
  display: grid;  
  justify-items: center;  
  align-items: center;  
}
```

# CSS Grid

**7. Minmax() Function :** The minmax(min, max) function allows dynamic sizing of grid tracks.

```
.container {  
  display: grid;  
  grid-template-columns: repeat(3, minmax(300px, 1fr));  
}
```

**5. Auto-Fill and Auto-Fit :** Both of these creates as many columns as possible but auto-fit collapses empty columns

```
.container {  
  display: grid;  
  grid-template-columns: repeat(auto-fit, minmax(250px, 1fr));  
}
```

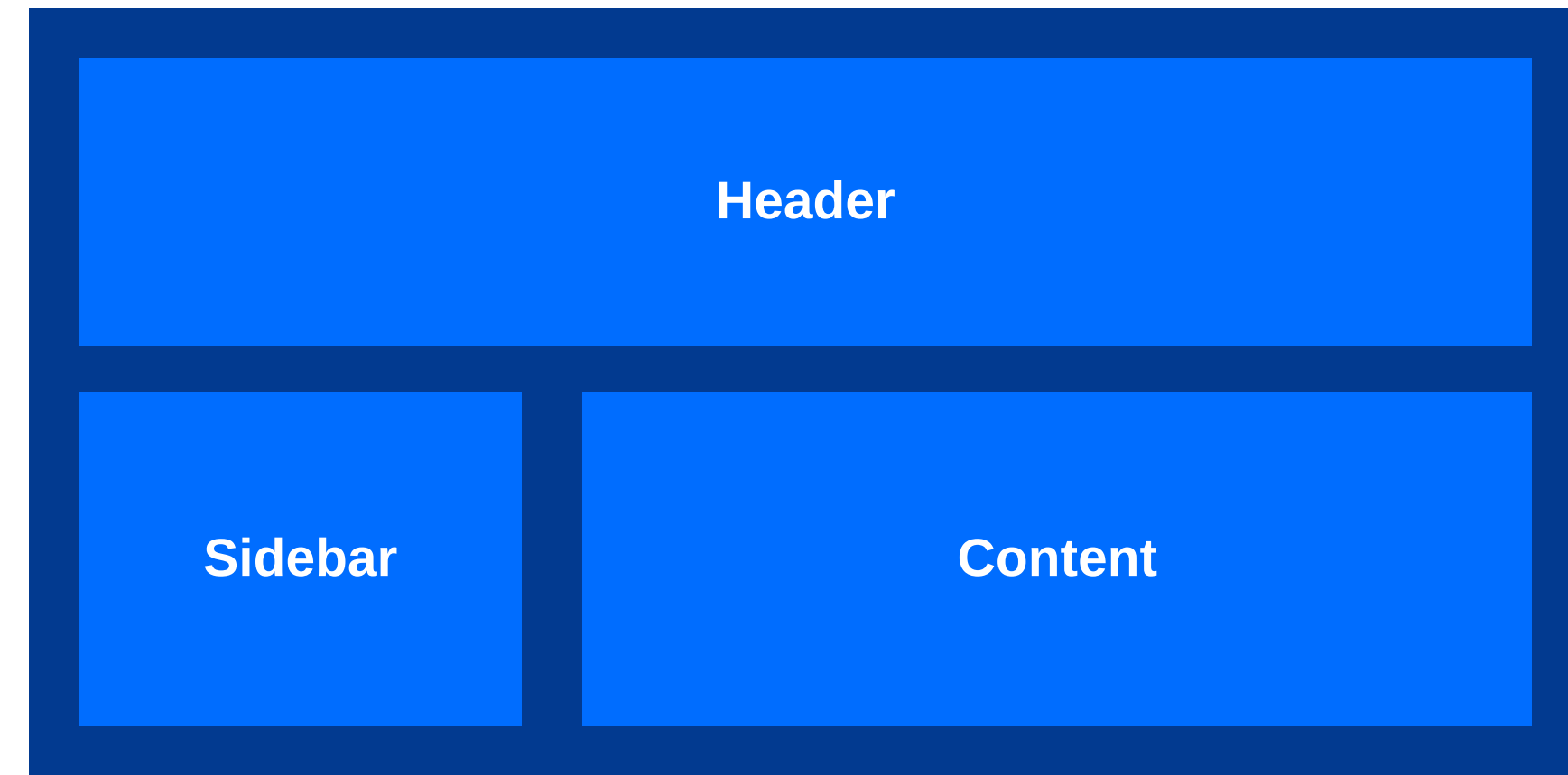
**6. justify & Align :** These properties control alignment inside the grid container and items.

```
.grid-item {  
  display: grid;  
  justify-items: center;  
  align-items: center;  
}
```

# CSS Grid

**7. Grid Area:** Grid areas are named sections that span multiple grid cells.

```
.container {  
  display: grid;  
  grid-template-columns: repeat(3, 1fr);  
  grid-template-rows: repeat(2, 100px);  
  grid-template-areas:  
    "header header header"  
    "sidebar content content";  
  grid-gap: 1rem;  
}  
  
.item1 { grid-area: header; }  
.item2 { grid-area: sidebar; }  
.item3 { grid-area: content; }
```





# THANK YOU

PHONE NUMBER

**(+91) 778 899 2897**

WEBSITE

**[www.indixpert.com](http://www.indixpert.com)**