

Responsiveness

Frappe Charts are responsive, as they rerender all the data in the current available container width. To demonstrate, let's take the example of setting the `bar width` for bar charts.

In order to set the bar width, instead of defining it and the space between the bars independently, we simply define the **ratio of the space** between bars to the bar width. The chart then adjusts the actual size proportional to the chart container.

js

```
barOptions: {  
  spaceRatio: 0.2 // default: 1  
},
```

Try resizing the window to see the effect, with different ratio values.



spaceRatio: 0.2

Responsiveness

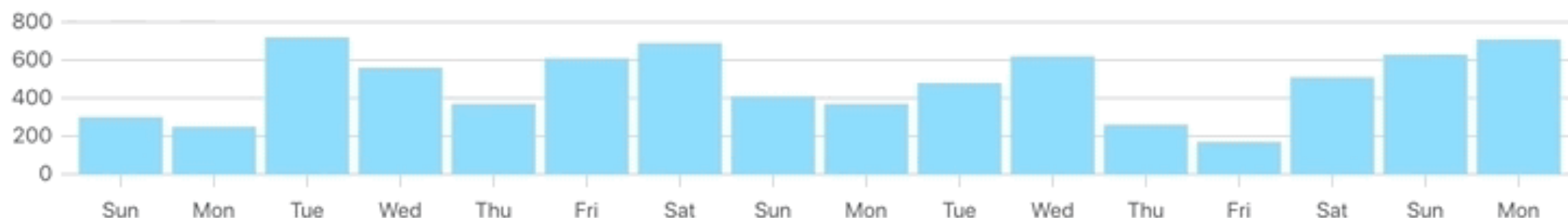
Frappe Charts are responsive, as they rerender all the data in the current available container width. To demonstrate, let's take the example of setting the `bar width` for bar charts.

In order to set the bar width, instead of defining it and the space between the bars independently, we simply define the **ratio of the space** between bars to the bar width. The chart then adjusts the actual size proportional to the chart container.

js

```
barOptions: {  
  spaceRatio: 0.2 // default: 1  
},
```

Try resizing the window to see the effect, with different ratio values.



spaceRatio: 0.2



Responsiveness

Frappe Charts are responsive, as they rerender all the data in the current available container width. To demonstrate, let's take the example of setting the `bar width` for bar charts.

In order to set the bar width, instead of defining it and the space between the bars independently, we simply define the **ratio of the space** between bars to the bar width. The chart then adjusts the actual size proportional to the chart container.

js

```
barOptions: {  
  spaceRatio: 0.2 // default: 1  
},
```

Try resizing the window to see the effect, with different ratio values.



spaceRatio: 0.2

Stacked Bar Chart

Unlike lines, bars have two ways to show multiple data point values: adjacent or stacked bars. Stacked bar charts are similar to area charts, being useful for comparisons of similar trends. The property [`stacked`]() in `barOptions` renders a stacked bar chart instead of the default adjacent bars:

```
```js
barOptions: {
 stacked: 1 // default 0, i.e. adjacent
}
```

<project-demo data="bar-composite-data" v-bind:config="{
  type: 'bar',
  height: 240,
  colors: ['blue', 'green', 'light-green'],
  barOptions: {
    spaceRatio: 0.4,
    stacked: 1
  },
}"
v-bind:options="[
  {
    name: 'barOptions',
    path: ['barOptions', 'stacked'],
    type: 'boolean',
    states: { 'Stacked': 1, 'Adjacent': 0 },
    activeState: 1
  }
]">
</project-demo>
```