* [Chart.js](http://docs.google.com/)
* [Getting Started](http://docs.google.com/getting-started/)
  + [Installation](http://docs.google.com/getting-started/installation.html)
  + [Integration](http://docs.google.com/getting-started/integration.html)
  + [Usage](http://docs.google.com/getting-started/usage.html)
* [General](http://docs.google.com/general/)
  + [Responsive](http://docs.google.com/general/responsive.html)
  + [Pixel Ratio](http://docs.google.com/general/device-pixel-ratio.html)
  + [Interactions](http://docs.google.com/general/interactions/)
    - [Events](http://docs.google.com/general/interactions/events.html)
    - [Modes](http://docs.google.com/general/interactions/modes.html)
  + [Options](http://docs.google.com/general/options.html)
  + [Colors](http://docs.google.com/general/colors.html)
  + [Fonts](http://docs.google.com/general/fonts.html)
* [Configuration](http://docs.google.com/configuration/)
  + [Animations](http://docs.google.com/configuration/animations.html)
  + [Layout](http://docs.google.com/configuration/layout.html)
  + [Legend](http://docs.google.com/configuration/legend.html)
  + [Title](http://docs.google.com/configuration/title.html)
  + [Tooltip](http://docs.google.com/configuration/tooltip.html)
  + [Elements](http://docs.google.com/configuration/elements.html)
* [Charts](http://docs.google.com/)
  + [Line](http://docs.google.com/line.html)
  + [Bar](http://docs.google.com/bar.html)
  + [Radar](http://docs.google.com/radar.html)
  + [Doughnut & Pie](http://docs.google.com/doughnut.html)
  + [Polar Area](http://docs.google.com/polar.html)
  + [Bubble](http://docs.google.com/bubble.html)
  + [Scatter](http://docs.google.com/scatter.html)
  + [Area](http://docs.google.com/area.html)
  + [Mixed](http://docs.google.com/mixed.html)
* [Axes](http://docs.google.com/axes/)
  + [Cartesian](http://docs.google.com/axes/cartesian/)
    - [Category](http://docs.google.com/axes/cartesian/category.html)
    - [Linear](http://docs.google.com/axes/cartesian/linear.html)
    - [Logarithmic](http://docs.google.com/axes/cartesian/logarithmic.html)
    - [Time](http://docs.google.com/axes/cartesian/time.html)
  + [Radial](http://docs.google.com/axes/radial/)
    - [Linear](http://docs.google.com/axes/radial/linear.html)
  + [Labelling](http://docs.google.com/axes/labelling.html)
  + [Styling](http://docs.google.com/axes/styling.html)
* [Developers](http://docs.google.com/developers/)
  + [Chart.js API](http://docs.google.com/developers/api.html)
  + [Updating Charts](http://docs.google.com/developers/updates.html)
  + [Plugins](http://docs.google.com/developers/plugins.html)
  + [New Charts](http://docs.google.com/developers/charts.html)
  + [New Axes](http://docs.google.com/developers/axes.html)
  + [Contributing](http://docs.google.com/developers/contributing.html)
* [Additional Notes](http://docs.google.com/notes/)
  + [Comparison Table](http://docs.google.com/notes/comparison.html)
  + [Popular Extensions](http://docs.google.com/notes/extensions.html)
  + [License](http://docs.google.com/notes/license.html)
* [Published with GitBook](https://www.gitbook.com)

[**Bar**](http://docs.google.com/)

Bar

A bar chart provides a way of showing data values represented as vertical bars. It is sometimes used to show trend data, and the comparison of multiple data sets side by side.

## Example Usage

var myBarChart = new Chart(ctx, {  
 type: 'bar',  
 data: data,  
 options: options  
});

## Dataset Properties

The bar chart allows a number of properties to be specified for each dataset. These are used to set display properties for a specific dataset. For example, the colour of the bars is generally set this way.

Some properties can be specified as an array. If these are set to an array value, the first value applies to the first bar, the second value to the second bar, and so on.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| label | String | The label for the dataset which appears in the legend and tooltips. |
| xAxisID | String | The ID of the x axis to plot this dataset on. If not specified, this defaults to the ID of the first found x axis |
| yAxisID | String | The ID of the y axis to plot this dataset on. If not specified, this defaults to the ID of the first found y axis. |
| backgroundColor | Color/Color[] | The fill color of the bar. See [Colors](http://docs.google.com/general/colors.html#colors) |
| borderColor | Color/Color[] | The color of the bar border. See [Colors](http://docs.google.com/general/colors.html#colors) |
| borderWidth | Number/Number[] | The stroke width of the bar in pixels. |
| borderSkipped | String | Which edge to skip drawing the border for. [more...](#gjdgxs) |
| hoverBackgroundColor | Color/Color[] | The fill colour of the bars when hovered. |
| hoverBorderColor | Color/Color[] | The stroke colour of the bars when hovered. |
| hoverBorderWidth | Number/Number[] | The stroke width of the bars when hovered. |

### borderSkipped

This setting is used to avoid drawing the bar stroke at the base of the fill. In general, this does not need to be changed except when creating chart types that derive from a bar chart.

Options are:

* 'bottom'
* 'left'
* 'top'
* 'right'

## Configuration Options

The bar chart defines the following configuration options. These options are merged with the global chart configuration options, Chart.defaults.global, to form the options passed to the chart.

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Default | Description |
| barPercentage | Number | 0.9 | Percent (0-1) of the available width each bar should be within the category width. 1.0 will take the whole category width and put the bars right next to each other. [more...](#30j0zll) |
| categoryPercentage | Number | 0.8 | Percent (0-1) of the available width each category should be within the sample width. [more...](#30j0zll) |
| barThickness | Number |  | Manually set width of each bar in pixels. If not set, the base sample widths are calculated automatically so that they take the full available widths without overlap. Then, the bars are sized using barPercentage and categoryPercentage. |
| maxBarThickness | Number |  | Set this to ensure that bars are not sized thicker than this. |
| gridLines.offsetGridLines | Boolean | true | If true, the bars for a particular data point fall between the grid lines. The grid line will move to the left by one half of the tick interval. If false, the grid line will go right down the middle of the bars. [more...](#1fob9te) |

### offsetGridLines

If true, the bars for a particular data point fall between the grid lines. The grid line will move to the left by one half of the tick interval, which is the space between the grid lines. If false, the grid line will go right down the middle of the bars. This is set to true for a bar chart while false for other charts by default.

This setting applies to the axis configuration. If axes are added to the chart, this setting will need to be set for each new axis.

options = {  
 scales: {  
 xAxes: [{  
 gridLines: {  
 offsetGridLines: true  
 }  
 }]  
 }  
}

## Default Options

It is common to want to apply a configuration setting to all created bar charts. The global bar chart settings are stored in Chart.defaults.bar. Changing the global options only affects charts created after the change. Existing charts are not changed.

## barPercentage vs categoryPercentage

The following shows the relationship between the bar percentage option and the category percentage option.

// categoryPercentage: 1.0  
// barPercentage: 1.0  
Bar: | 1.0 | 1.0 |  
Category: | 1.0 |  
Sample: |===========|  
  
// categoryPercentage: 1.0  
// barPercentage: 0.5  
Bar: |.5| |.5|  
Category: | 1.0 |  
Sample: |==============|  
  
// categoryPercentage: 0.5  
// barPercentage: 1.0  
Bar: |1.||1.|  
Category: | .5 |  
Sample: |==============|

## Data Structure

The data property of a dataset for a bar chart is specified as a an array of numbers. Each point in the data array corresponds to the label at the same index on the x axis.

data: [20, 10]

You can also specify the dataset as x/y coordinates when using the [time scale](http://docs.google.com/axes/cartesian/time.html#time-cartesian-axis).

data: [{x:'2016-12-25', y:20}, {x:'2016-12-26', y:10}]

# Stacked Bar Chart

Bar charts can be configured into stacked bar charts by changing the settings on the X and Y axes to enable stacking. Stacked bar charts can be used to show how one data series is made up of a number of smaller pieces.

var stackedBar = new Chart(ctx, {  
 type: 'bar',  
 data: data,  
 options: {  
 scales: {  
 xAxes: [{  
 stacked: true  
 }],  
 yAxes: [{  
 stacked: true  
 }]  
 }  
 }  
});

## Dataset Properties

The following dataset properties are specific to stacked bar charts.

|  |  |  |
| --- | --- | --- |
| Name | Type | Description |
| stack | String | The ID of the group to which this dataset belongs to (when stacked, each group will be a separate stack) |

# Horizontal Bar Chart

A horizontal bar chart is a variation on a vertical bar chart. It is sometimes used to show trend data, and the comparison of multiple data sets side by side.

## Example

var myBarChart = new Chart(ctx, {  
 type: 'horizontalBar',  
 data: data,  
 options: options  
});

## Config Options

The configuration options for the horizontal bar chart are the same as for the [bar chart](#3znysh7). However, any options specified on the x axis in a bar chart, are applied to the y axis in a horizontal bar chart.

The default horizontal bar configuration is specified in Chart.defaults.horizontalBar.

results matching ""

No results matching ""