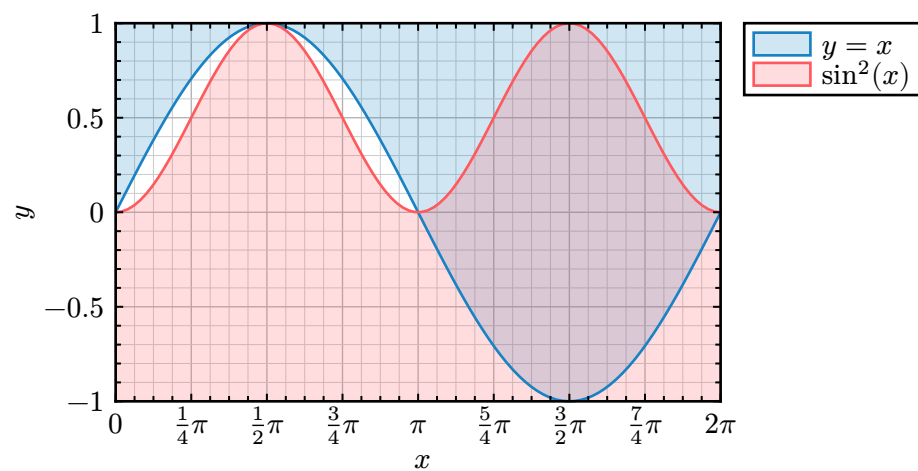


CeTZ Plot

Johannes Wolf
fenjalien

Version 0.1.0



1 Introduction	3
2 Usage	3
3 Plot	3
3.1 plot	3
3.1.1 Parameters	3
3.1.2 Options	5
3.2 Add	10
3.2.1 bar	10
4 Chart	11
4.1 Bar	11
4.1.1 clustered	11
4.1.2 stacked	12

1 Introduction

CeTZ-Plot is a package for making plots in Typst using CeTZ.

2 Usage

This is the minimal starting point:

```
#import "@preview/cetz:0.2.2"
#import "@preview/cetz-plot:0.1.0"
#cezt.canvas({
  cetz-plot.plot(...,{
    })
})
```

Note that plot functions are imported inside the scope of the canvas block. All following example code is expected to be inside a canvas block, with the cetz-plot module imported into the namespace.

3 Plot

3.1 plot

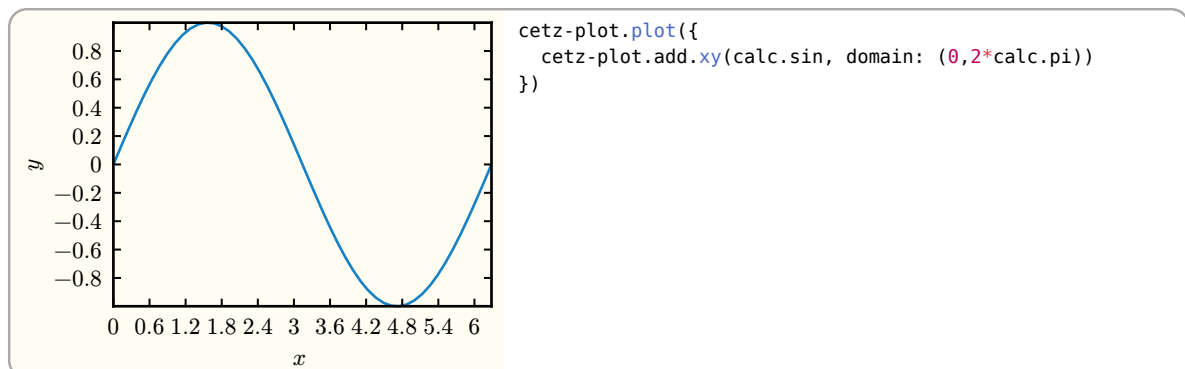
Create a plot environment. Data to be plotted is given by passing it to the `plot.add` or other plotting functions. The plot environment supports different axis styles to draw, see its parameter `axis-style`.

3.1.1 Parameters

```
plot(
  body: body,
  size: array,
  axis-style: axis-style-module,
  name: string none,
  plot-style: style function,
  mark-style: style function,
  legend: none auto coordinate,
  legend-anchor: auto string,
  legend-style: style,
  ..options: any
)
```

body: body

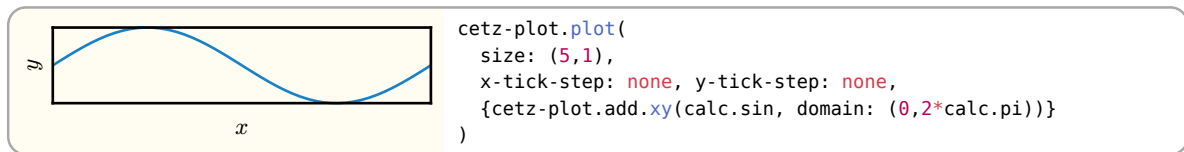
Calls of `plot.add` or `plot.add-*` commands. Note that normal drawing commands like `line` or `rect` are not allowed inside the plots body, instead wrap them in `plot.annotate`, which lets you select the axes used for drawing.



size: array

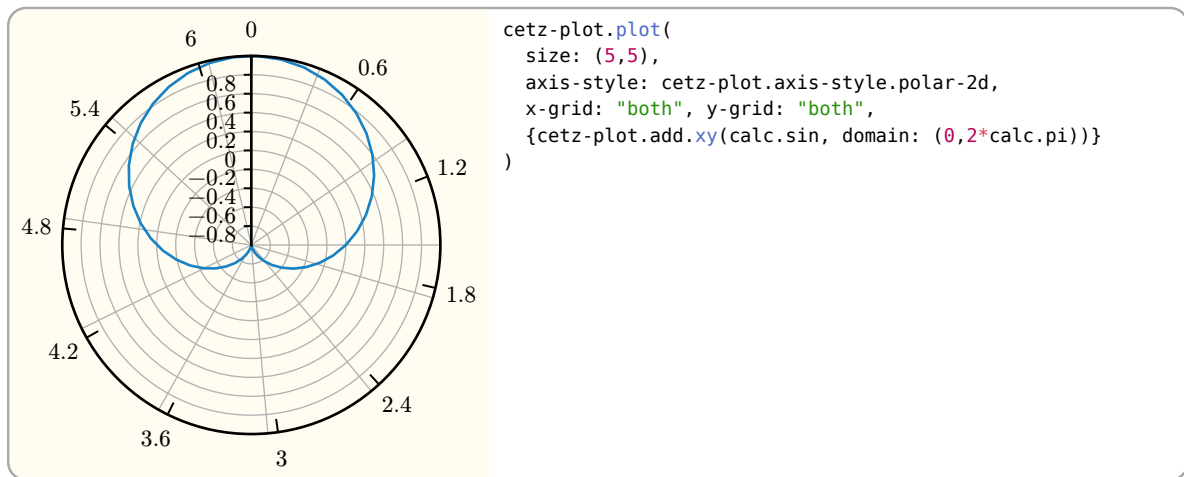
Default: (5, 5 * 3/4)

Plot size tuple of (<width>, <height>) in canvas units. This is the plots inner plotting size without axes and labels. this value, as it doesn't include axis labels, ticks, or the legend.

**axis-style:** axis-style-module

Default: axis-style.orthorect-2d

TODO: Make this link to the axis-style section

**name:** string or none

Default: none

The plots element name to be used when referring to anchors

plot-style: style or function

Default: default-plot-style

Styling to use for drawing plot graphs. This style gets inherited by all plots and supports palette functions. The following style keys are supported:

stroke: none or stroke

Default: 1pt

Stroke style to use for stroking the graph.

fill: none or paint

Default: none

Paint to use for filled graphs. Note that not all graphs may support filling and that you may have to enable filling per graph, see `plot.add(fill: ..)`.

mark-style: style or function

Default: default-mark-style

Styling to use for drawing plot marks. This style gets inherited by all plots and supports palette functions. The following style keys are supported:

stroke: none or stroke

Default: 1pt

Stroke style to use for stroking the mark.

fill: none or paint

Default: none

Paint to use for filling marks.

legend: none or auto or coordinate

Default: auto

The position the legend will be drawn at. See plot-legends for information about legends. If set to <auto>, the legend's "default-placement" styling will be used. If set to a <coordinate>, it will be taken as relative to the plot's origin.

legend-anchor: auto or string

Default: auto

Anchor of the legend group to use as its origin. If set to auto and legend is one of the predefined legend anchors, the opposite anchor to legend gets used.

legend-style: style

Default: (:)

Style key-value overwrites for the legend style with style root legend.

..options: any

Axis options, see *options* below.

To draw elements inside a plot, using the plot's coordinate system, use the `plot.annotate(. .)` function.

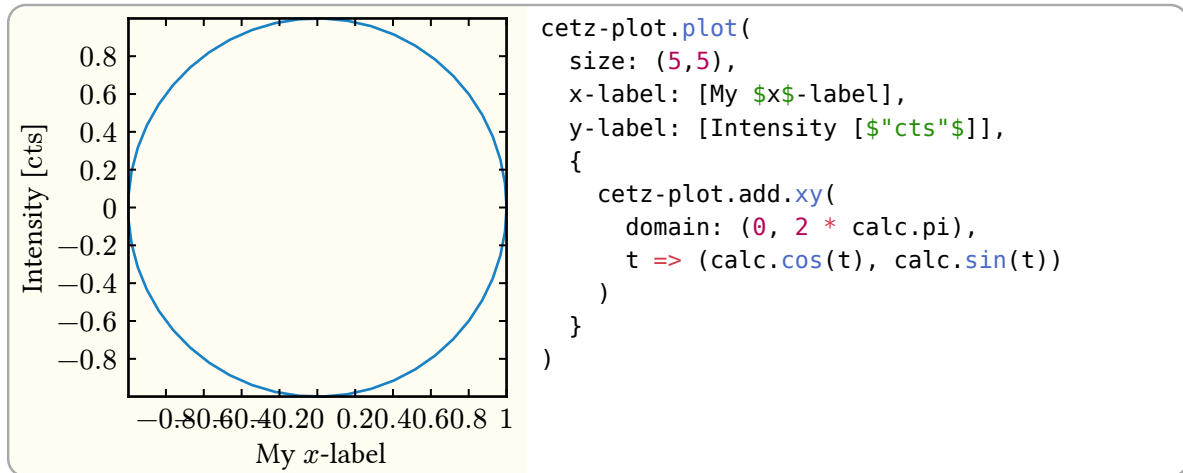
3.1.2 Options

You can use the following options to customize each axis of the plot. You must pass them as named arguments prefixed by the axis name followed by a dash (-) they should target. Example: `x-min: 0`, `y-ticks: (..)` or `x2-label: [..]`.

label: none or content

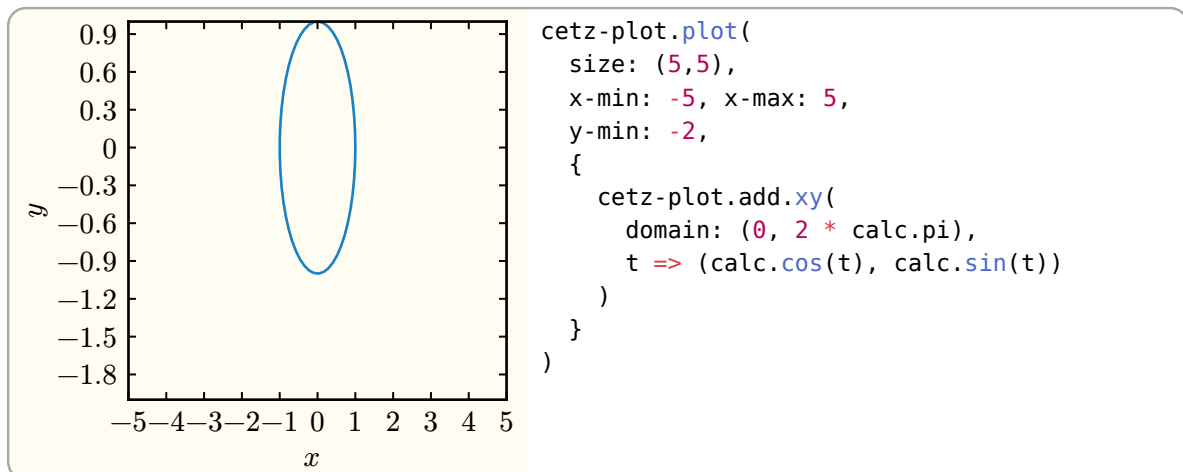
Default: none

The axis' label. If and where the label is drawn depends on the axis-style.

**min:** auto or float

Default: auto

Axis lower domain value. If this is set greater than than max, the axis' direction is swapped

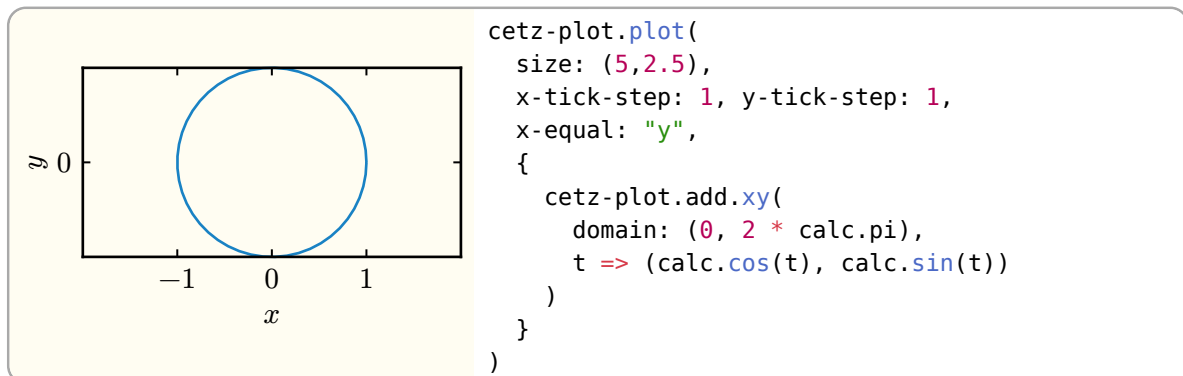
**max:** auto or float

Default: auto

Axis upper domain value. If this is set to a lower value than min, the axis' direction is swapped

equal: stringDefault: **none**

Set the axis domain to keep a fixed aspect ratio by multiplying the other axis domain by the plots aspect ratio, depending on the other axis orientation (see `horizontal`). This can be useful to force one axis to grow or shrink with another one. You can only “lock” two axes of different orientations.

**horizontal:** bool

Default: "axis name dependant"

If true, the axis is considered an axis that gets drawn horizontally, vertically otherwise. The default value depends on the axis name on axis creation. Axes which name start with x have this set to true, all others have it set to false. Each plot has to use one horizontal and one vertical axis for plotting, a combination of two y-axes will panic: ("y", "y2").

tick-step: none or auto or floatDefault: **auto**

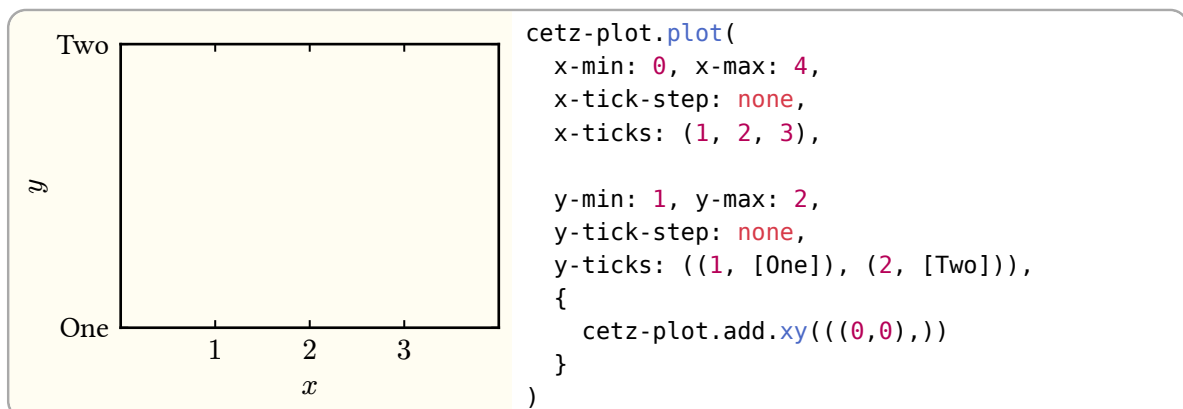
The increment between tick marks on the axis. If set to auto, an increment is determined. When set to none, incrementing tick marks are disabled.

minor-tick-step: none or floatDefault: **none**

Like tick-step, but for minor tick marks. In contrast to ticks, minor ticks do not have labels.

ticks: none or arrayDefault: **none**

A List of custom tick marks to additionally draw along the axis. They can be passed as an array of <float> values or an array of (<float>, <content>) tuples for setting custom tick mark labels per mark.



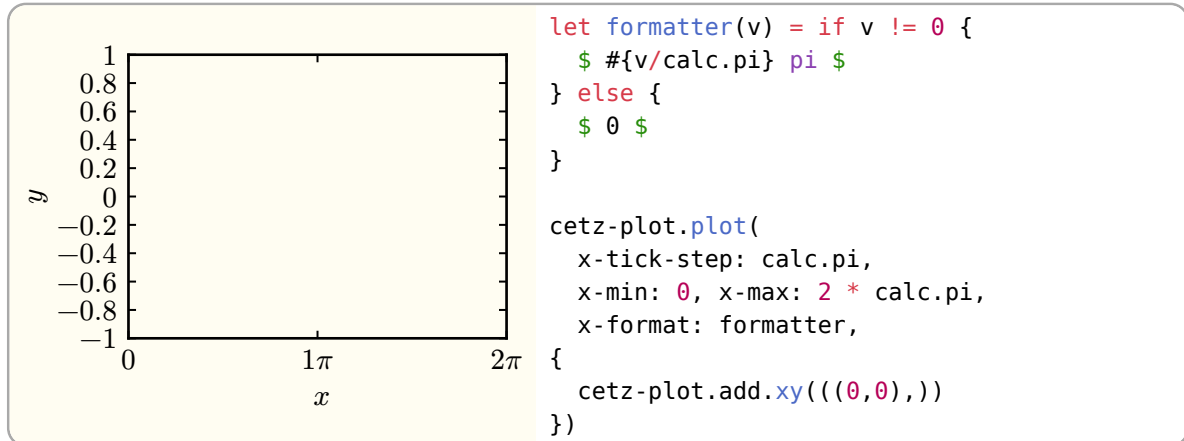
Examples: (1, 2, 3) or ((1, [One]), (2, [Two]), (3, [Three]))

format: `none` or `string` or `function`Default: `"float"`

How to format the tick label: You can give a function that takes a `<float>` and return `<content>` to use as the tick label. You can also give one of the predefined options:

float Floating point formatting rounded to two digits after the point (see decimals)

sci Scientific formatting with $\times 10^n$ used as exponent syntax

**decimals:** `int`Default: `2`

Number of decimals digits to display for tick labels, if the format is set to `"float"`.

unit: `none` or `content`Default: `none`

Suffix to append to all tick labels.

mode: `none` or `string`Default: `none`

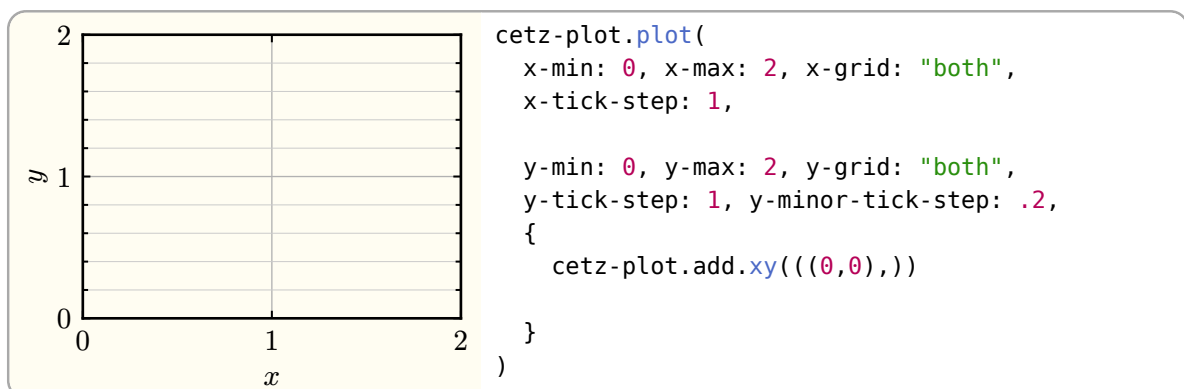
The scaling function of the axis. Takes `lin` (default) for linear scaling, and `log` for logarithmic scaling.

base: `none` or `number`Default: `none`

The base to be used when labeling axis ticks in logarithmic scaling

grid: `bool` or `string`Default: `false`

If true or `"major"`, show grid lines for all major ticks. If set to `"minor"`, show grid lines for minor ticks only. The value `"both"` enables grid lines for both, major- and minor ticks.



break: bool

Default: false

If true, add a “sawtooth” at the start or end of the axis line, depending on the axis bounds. If the axis min. value is > 0 , a sawtooth is added to the start of the axes, if the axis max. value is < 0 , a sawtooth is added to its end.

3.2 Add

3.2.1 bar

Parameters

```
bar(  
    data,  
    x-key,  
    y-key,  
    y-offset-key,  
    bar-width,  
    label,  
    style,  
    axes  
)
```

data:

x-key: Default: **0**

y-key: Default: **1**

y-offset-key: Default: **none**

bar-width: Default: **0.5**

label: Default: **none**

style: Default: **(:)**

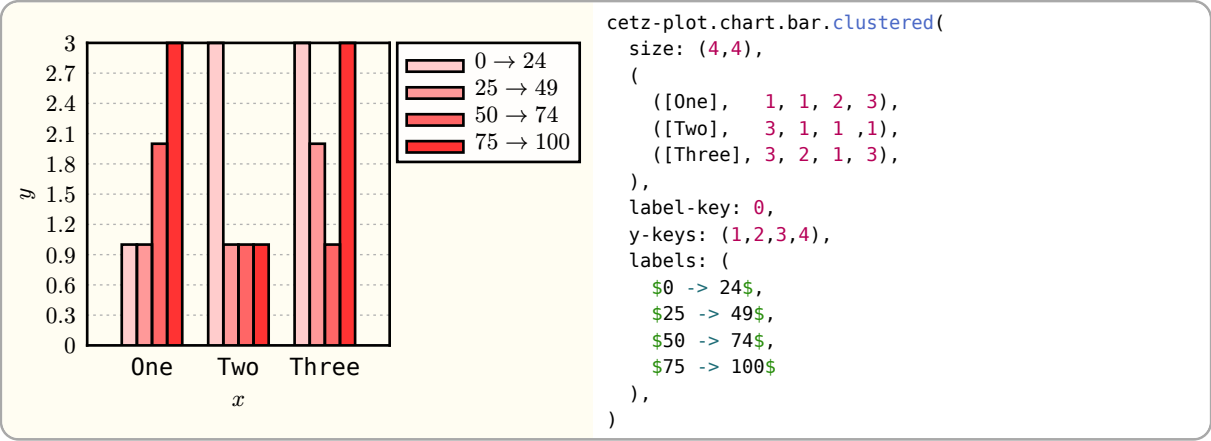
axes: Default: **("x", "y")**

4 Chart

4.1 Bar

4.1.1 clustered

Render a clustered bar chart



Parameters

```
clustered(  
  data,  
  labels,  
  label-key,  
  y-keys,  
  y-error-keys,  
  bar-width,  
  bar-spacing,  
  ..plot-args  
)  
data:
```

labels: Default: ()

label-key: Default: 0

y-keys: Default: (1,)

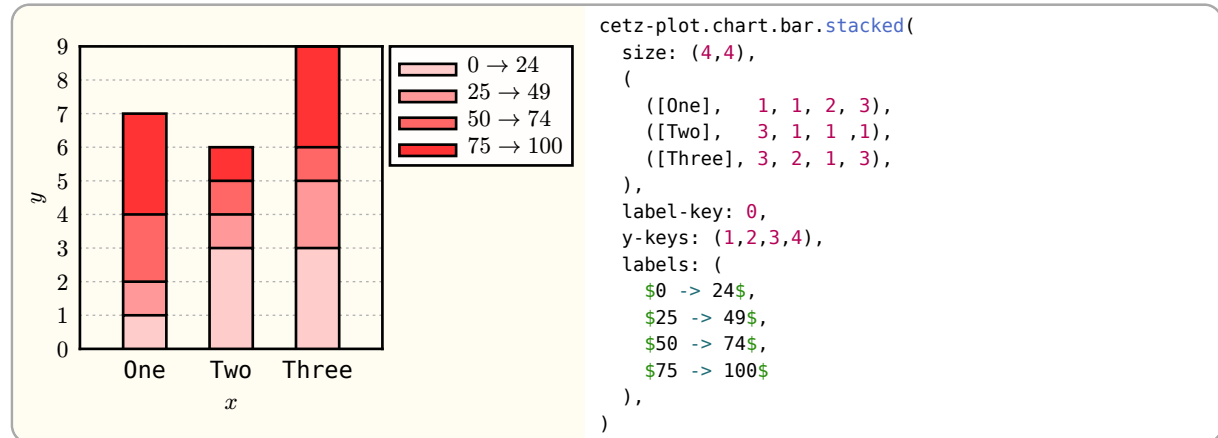
y-error-keys: Default: none

bar-width: Default: 0.7

bar-spacing: Default: 0

..plot-args:**4.1.2 stacked**

Render a stacked bar chart

**Parameters**

```
stacked(
  data,
  labels,
  label-key,
  y-keys,
  y-error-keys,
  bar-width,
  ..plot-args: variadic
)
```

data:**labels:** Default: `()`**label-key:** Default: `0`**y-keys:** Default: `(1,)`**y-error-keys:** Default: `none`**bar-width:** Default: `0.5`**..plot-args:** `variadic`Additional plotting parameters and axis options to be passed to `plot()`