 [Chart.js](http://docs.google.com/docs/3.9.1/)

[Home](http://docs.google.com/docs/3.9.1/)

[API](http://docs.google.com/docs/3.9.1/api/)

[Samples](http://docs.google.com/docs/3.9.1/samples/)

Ecosystem Ecosystem

* [Awesome (opens new window)](https://github.com/chartjs/awesome)
* [Slack (opens new window)](https://chartjs-slack.herokuapp.com/)
* [Stack Overflow (opens new window)](https://stackoverflow.com/questions/tagged/chart.js)

[GitHub (opens new window)](https://github.com/chartjs/Chart.js)

[Home](http://docs.google.com/docs/3.9.1/)

[API](http://docs.google.com/docs/3.9.1/api/)

[Samples](http://docs.google.com/docs/3.9.1/samples/)

Ecosystem Ecosystem

* [Awesome (opens new window)](https://github.com/chartjs/awesome)
* [Slack (opens new window)](https://chartjs-slack.herokuapp.com/)
* [Stack Overflow (opens new window)](https://stackoverflow.com/questions/tagged/chart.js)

[GitHub (opens new window)](https://github.com/chartjs/Chart.js)

* [Chart.js](http://docs.google.com/docs/3.9.1/)
* Getting Started
* General
* Configuration
  + [Configuration](http://docs.google.com/docs/3.9.1/configuration/)
  + [Animations](http://docs.google.com/docs/3.9.1/configuration/animations.html)
  + [Canvas background](http://docs.google.com/docs/3.9.1/configuration/canvas-background.html)
  + [Data Decimation](http://docs.google.com/docs/3.9.1/configuration/decimation.html)
  + [Device Pixel Ratio](http://docs.google.com/docs/3.9.1/configuration/device-pixel-ratio.html)
  + [Elements](http://docs.google.com/docs/3.9.1/configuration/elements.html)
  + [Interactions](http://docs.google.com/docs/3.9.1/configuration/interactions.html)
  + [Layout](http://docs.google.com/docs/3.9.1/configuration/layout.html)
  + [Legend](http://docs.google.com/docs/3.9.1/configuration/legend.html)
  + [Locale](http://docs.google.com/docs/3.9.1/configuration/locale.html)
  + [Responsive Charts](http://docs.google.com/docs/3.9.1/configuration/responsive.html)
  + [Subtitle](http://docs.google.com/docs/3.9.1/configuration/subtitle.html)
  + [Title](http://docs.google.com/docs/3.9.1/configuration/title.html)
  + [Tooltip](http://docs.google.com/docs/3.9.1/configuration/tooltip.html)
* Chart Types
* Axes
* Developers

[**#**](#gjdgxs) Animations

Chart.js animates charts out of the box. A number of options are provided to configure how the animation looks and how long it takes.

config setup

const config = { type: 'line', data: data, options: { animations: { tension: { duration: 1000, easing: 'linear', from: 1, to: 0, loop: true } }, scales: { y: { // defining min and max so hiding the dataset does not change scale range min: 0, max: 100 } } } };

const config = {  
 type: 'line',  
 data: data,  
 options: {  
 animations: {  
 tension: {  
 duration: 1000,  
 easing: 'linear',  
 from: 1,  
 to: 0,  
 loop: true  
 }  
 },  
 scales: {  
 y: { // defining min and max so hiding the dataset does not change scale range  
 min: 0,  
 max: 100  
 }  
 }  
 }  
};

const data = { labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July'], datasets: [{ label: 'Looping tension', data: [65, 59, 80, 81, 26, 55, 40], fill: false, borderColor: 'rgb(75, 192, 192)', }] };

const data = {  
 labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July'],  
 datasets: [{  
 label: 'Looping tension',  
 data: [65, 59, 80, 81, 26, 55, 40],  
 fill: false,  
 borderColor: 'rgb(75, 192, 192)',  
 }]  
};

config setup

const config = { type: 'line', data: data, options: { transitions: { show: { animations: { x: { from: 0 }, y: { from: 0 } } }, hide: { animations: { x: { to: 0 }, y: { to: 0 } } } } } };

const config = {  
 type: 'line',  
 data: data,  
 options: {  
 transitions: {  
 show: {  
 animations: {  
 x: {  
 from: 0  
 },  
 y: {  
 from: 0  
 }  
 }  
 },  
 hide: {  
 animations: {  
 x: {  
 to: 0  
 },  
 y: {  
 to: 0  
 }  
 }  
 }  
 }  
 }  
};

const data = { labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July'], datasets: [{ label: 'Try hiding me', data: [65, 59, 80, 81, 26, 55, 40], fill: false, borderColor: 'rgb(75, 192, 192)', }] };

const data = {  
 labels: ['January', 'February', 'March', 'April', 'May', 'June', 'July'],  
 datasets: [{  
 label: 'Try hiding me',  
 data: [65, 59, 80, 81, 26, 55, 40],  
 fill: false,  
 borderColor: 'rgb(75, 192, 192)',  
 }]  
};

## [**#**](#30j0zll) Animation configuration

Animation configuration consists of 3 keys.

| Name | Type | Details |
| --- | --- | --- |
| animation | object | [animation](#1fob9te) |
| animations | object | [animations](#gjdgxs) |
| transitions | object | [transitions](#3znysh7) |

These keys can be configured in following paths:

* `` - chart options
* datasets[type] - dataset type options
* overrides[type] - chart type options

These paths are valid under defaults for global configuration and options for instance configuration.

## [**#**](#1fob9te) animation

The default configuration is defined here: [core.animations.js](https://github.com/chartjs/Chart.js/blob/master/src/core/core.animations.js)

Namespace: options.animation

| Name | Type | Default | Description |
| --- | --- | --- | --- |
| duration | number | 1000 | The number of milliseconds an animation takes. |
| easing | string | 'easeOutQuart' | Easing function to use. [more...](#2et92p0) |
| delay | number | undefined | Delay before starting the animations. |
| loop | boolean | undefined | If set to true, the animations loop endlessly. |

These defaults can be overridden in options.animation or dataset.animation and tooltip.animation. These keys are also [Scriptable](http://docs.google.com/docs/3.9.1/general/options.html#scriptable-options).

## [**#**](#tyjcwt) animations

Animations options configures which element properties are animated and how. In addition to the main [animation configuration](#30j0zll), the following options are available:

Namespace: options.animations[animation]

| Name | Type | Default | Description |
| --- | --- | --- | --- |
| properties | string[] | key | The property names this configuration applies to. Defaults to the key name of this object. |
| type | string | typeof property | Type of property, determines the interpolator used. Possible values: 'number', 'color' and 'boolean'. Only really needed for 'color', because typeof does not get that right. |
| from | number|Color|boolean | undefined | Start value for the animation. Current value is used when undefined |
| to | number|Color|boolean | undefined | End value for the animation. Updated value is used when undefined |
| fn | <T>(from: T, to: T, factor: number) => T; | undefined | Optional custom interpolator, instead of using a predefined interpolator from type |

### [**#**](#3dy6vkm) Default animations

| Name | Option | Value |
| --- | --- | --- |
| numbers | properties | ['x', 'y', 'borderWidth', 'radius', 'tension'] |
| numbers | type | 'number' |
| colors | properties | ['color', 'borderColor', 'backgroundColor'] |
| colors | type | 'color' |

Note

These default animations are overridden by most of the dataset controllers.

## [**#**](#3znysh7) transitions

The core transitions are 'active', 'hide', 'reset', 'resize', 'show'. A custom transition can be used by passing a custom mode to [update](http://docs.google.com/docs/3.9.1/developers/api.html#updatemode). Transition extends the main [animation configuration](#30j0zll) and [animations configuration](#1t3h5sf).

### [**#**](#4d34og8) Default transitions

Namespace: options.transitions[mode]

| Mode | Option | Value | Description |
| --- | --- | --- | --- |
| 'active' | animation.duration | 400 | Override default duration to 400ms for hover animations |
| 'resize' | animation.duration | 0 | Override default duration to 0ms (= no animation) for resize |
| 'show' | animations.colors | { type: 'color', properties: ['borderColor', 'backgroundColor'], from: 'transparent' } | Colors are faded in from transparent when dataset is shown using legend / [api](http://docs.google.com/docs/3.9.1/developers/api.html#showdatasetIndex). |
| 'show' | animations.visible | { type: 'boolean', duration: 0 } | Dataset visibility is immediately changed to true so the color transition from transparent is visible. |
| 'hide' | animations.colors | { type: 'color', properties: ['borderColor', 'backgroundColor'], to: 'transparent' } | Colors are faded to transparent when dataset id hidden using legend / [api](http://docs.google.com/docs/3.9.1/developers/api.html#hidedatasetIndex). |
| 'hide' | animations.visible | { type: 'boolean', easing: 'easeInExpo' } | Visibility is changed to false at a very late phase of animation |

## [**#**](#2s8eyo1) Disabling animation

To disable an animation configuration, the animation node must be set to false, with the exception for animation modes which can be disabled by setting the duration to 0.

chart.options.animation = false; // disables all animations  
chart.options.animations.colors = false; // disables animation defined by the collection of 'colors' properties  
chart.options.animations.x = false; // disables animation defined by the 'x' property  
chart.options.transitions.active.animation.duration = 0; // disables the animation for 'active' mode

## [**#**](#2et92p0) Easing

Available options are:

* 'linear'
* 'easeInQuad'
* 'easeOutQuad'
* 'easeInOutQuad'
* 'easeInCubic'
* 'easeOutCubic'
* 'easeInOutCubic'
* 'easeInQuart'
* 'easeOutQuart'
* 'easeInOutQuart'
* 'easeInQuint'
* 'easeOutQuint'
* 'easeInOutQuint'
* 'easeInSine'
* 'easeOutSine'
* 'easeInOutSine'
* 'easeInExpo'
* 'easeOutExpo'
* 'easeInOutExpo'
* 'easeInCirc'
* 'easeOutCirc'
* 'easeInOutCirc'
* 'easeInElastic'
* 'easeOutElastic'
* 'easeInOutElastic'
* 'easeInBack'
* 'easeOutBack'
* 'easeInOutBack'
* 'easeInBounce'
* 'easeOutBounce'
* 'easeInOutBounce'

See [Robert Penner's easing equations (opens new window)](http://robertpenner.com/easing/).

## [**#**](#17dp8vu) Animation Callbacks

The animation configuration provides callbacks which are useful for synchronizing an external draw to the chart animation. The callbacks can be set only at main [animation configuration](#30j0zll).

Namespace: options.animation

| Name | Type | Default | Description |
| --- | --- | --- | --- |
| onProgress | function | null | Callback called on each step of an animation. |
| onComplete | function | null | Callback called when all animations are completed. |

The callback is passed the following object:

{  
 // Chart object  
 chart: Chart,  
 // Number of animations still in progress  
 currentStep: number,  
 // `true` for the initial animation of the chart  
 initial: boolean,  
 // Total number of animations at the start of current animation  
 numSteps: number,  
}

The following example fills a progress bar during the chart animation.

const chart = new Chart(ctx, {  
 type: 'line',  
 data: data,  
 options: {  
 animation: {  
 onProgress: function(animation) {  
 progress.value = animation.currentStep / animation.numSteps;  
 }  
 }  
 }  
});

Another example usage of these callbacks can be found [in this progress bar sample.](http://docs.google.com/docs/3.9.1/samples/advanced/progress-bar.html) which displays a progress bar showing how far along the animation is.

Last Updated: 8/3/2022, 12:46:38 PM

←  [Configuration](http://docs.google.com/docs/3.9.1/configuration/)   [Canvas background](http://docs.google.com/docs/3.9.1/configuration/canvas-background.html)  →