

# Visualization: Area Chart

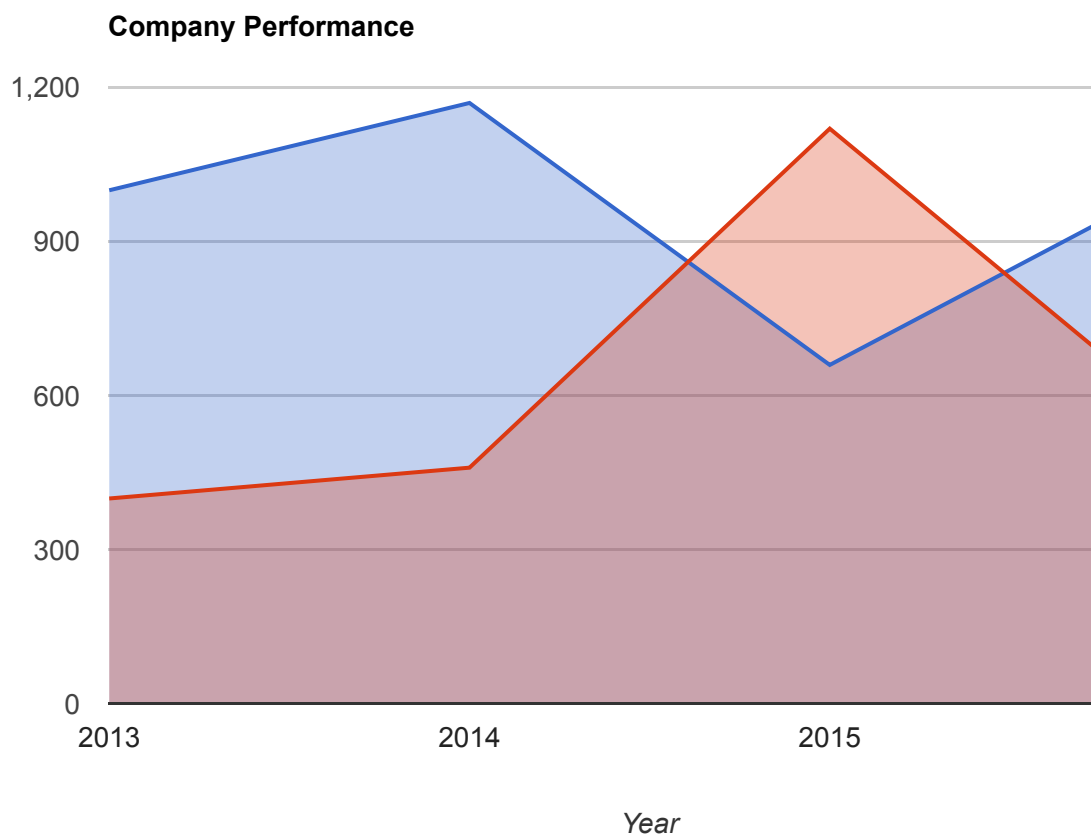
## Overview

An area chart that is rendered within the browser using [SVG](http://www.w3.org/Graphics/SVG/)

(<http://www.w3.org/Graphics/SVG/>) or [VML](http://en.wikipedia.org/wiki/Vector_Markup_Language) ([http://en.wikipedia.org/wiki/Vector\\_Markup\\_Language](http://en.wikipedia.org/wiki/Vector_Markup_Language)).

Displays tips when hovering over points.

## A Simple Example



[CODE IT YOURSELF ON JSFIDDLE](#)

```
<html>
<head>
  <script type="text/javascript" src="https://www.gstatic.com/charts/loader">
  <script type="text/javascript">
    google.charts.load('current', {'packages':['corechart']});
```

```

google.charts.setOnLoadCallback(drawChart);

function drawChart() {
  var data = google.visualization.arrayToDataTable([
    ['Year', 'Sales', 'Expenses'],
    ['2013', 1000, 400],
    ['2014', 1170, 460],
    ['2015', 660, 1120],
    ['2016', 1030, 540]
  ]);

  var options = {
    title: 'Company Performance',
    hAxis: {title: 'Year', titleTextStyle: {color: '#333'}},
    vAxis: {minValue: 0}
  };

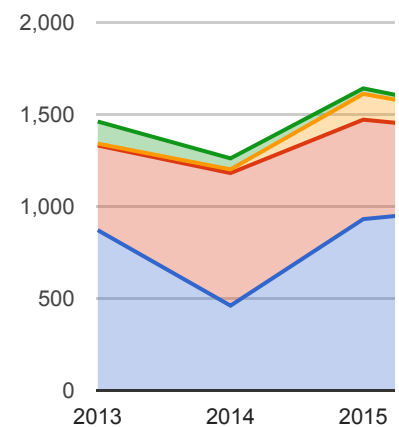
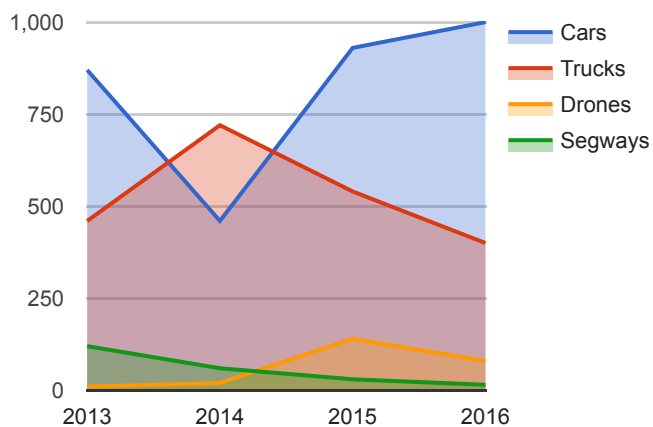
  var chart = new google.visualization.AreaChart(document.getElementById('chart_div'));
  chart.draw(data, options);
}
</script>
</head>
<body>
  <div id="chart_div" style="width: 100%; height: 500px;"></div>
</body>
</html>

```

## Stacking Areas

By default, the area chart draws the series on top of one another. You can stack them atop one another instead, so that the data values at each x-value are summed. In an area chart, the value for each series will always be stacked relative to the preceding series value. Stacking a mix of negative and positive values will cause the areas to overlap. It is important to note that the `interpolateNulls` option **does not** work with stacked area charts.

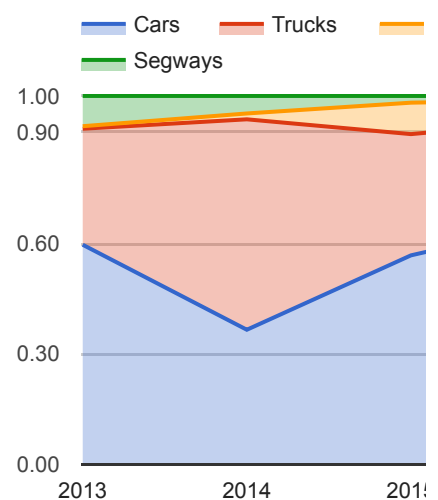
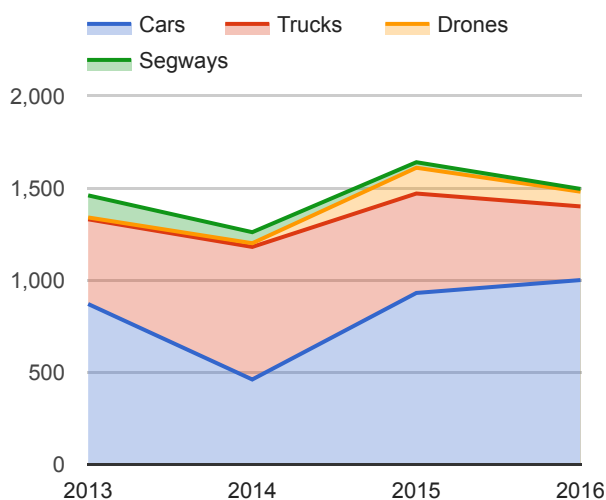
On the left, `isStacked` is set to false (the default), and on the right it's set to true:



Note that the order of the legend entries is different. In the second, stacked chart, the order is reversed, placing series 0 at the bottom, to better correspond with the stacking of the series elements, making the legend correspond to the data.

Stacked area charts also support 100% stacking, where the stacks of elements at each domain-value are rescaled such that they add up to 100%. The options for this are `isStacked: 'percent'`, which formats each value as a percentage of 100%, and `isStacked: 'relative'`, which formats each value as a fraction of 1. There is also an `isStacked: 'absolute'` option, which is functionally equivalent to `isStacked: true`.

Note in the 100% stacked chart on the right, the tick values are based on the relative 0-1 scale as fractions of 1.



STACKED

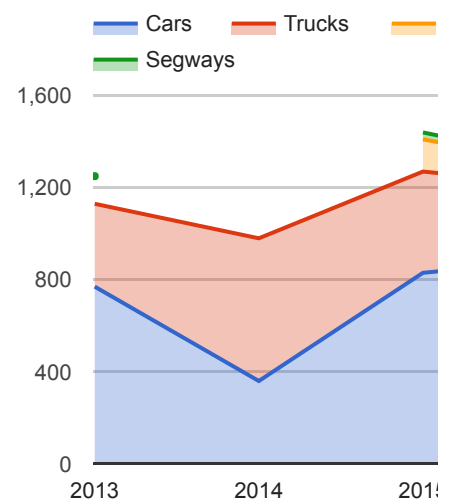
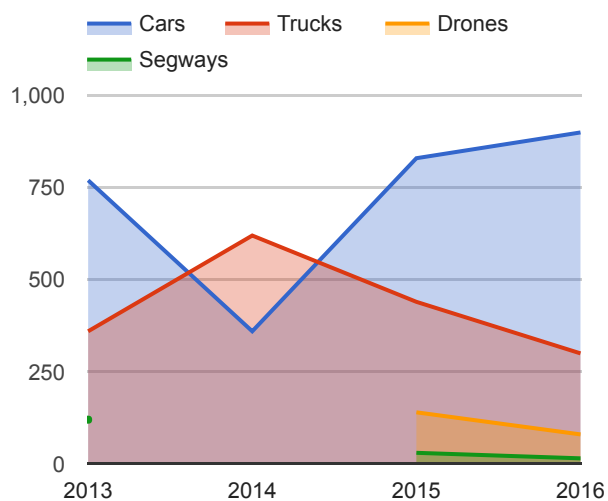
100% STACKED

```

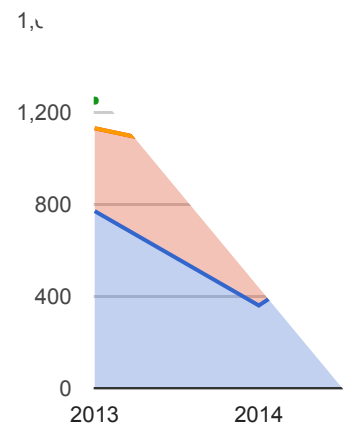
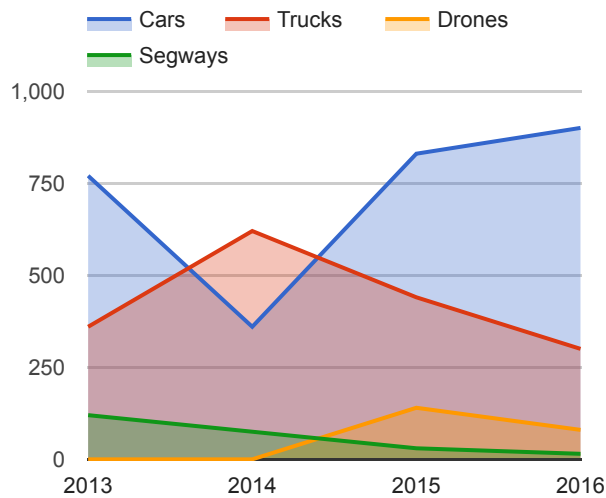
var options_fullStacked = {
  isStacked: 'relative',
  height: 300,
  legend: {position: 'top', maxLines: 3},
  vAxis: {
    minValue: 0,
    ticks: [0, .3, .6, .9, 1]
  }
};

```

Suppose one of your series has no data for some of your x-values. For instance, in the charts above, let's assume that drones aren't available until 2015, and Segways are not available in 2014. Note that we use null values where data is lacking, so the chart would look like this:



If those discontinuities don't appeal, you can substitute zeros for the nulls, and/or set the `interpolateNulls` option to `true`:



## Loading

The `google.charts.load` package name is `"corechart"`.

```
google.charts.load("current", {packages: ["corechart"]});
```

The visualization's class name is `google.visualization.AreaChart`.

```
var visualization = new google.visualization.AreaChart(container);
```

## Data Format

**Rows:** Each row in the table represents a set of data points with the same x-axis location.

**Columns:**

Column 0	
Purpose:	<ul style="list-style-type: none"> <li>X-axis group labels (<u>discrete</u> (<a href="https://developers.google.com/chart/">https://developers.google.com/chart/</a>))</li> <li>X-axis values (<u>continuous</u> (<a href="https://developers.google.com/chart/">https://developers.google.com/chart/</a>))</li> </ul>
Data Type:	<ul style="list-style-type: none"> <li>string (<u>discrete</u></li> </ul>

	<a href="https://developers.google.com/chart/interactive/docs/roles">https://developers.google.com/chart/interactive/docs/roles</a>
	<ul style="list-style-type: none"> <li>number, date, datetime, or timeofday (<a href="https://developers.google.com/chart/interactive/docs/roles">continuous</a>)</li> </ul>
Role:	domain
	<ul style="list-style-type: none"> <li><a href="https://developers.google.com/chart/interactive/docs/roles">annotation</a></li> <li><a href="https://developers.google.com/chart/interactive/docs/roles">annotationText</a></li> </ul>
<b>Optional <u>column roles</u></b> <a href="https://developers.google.com/chart/interactive/docs/roles">https://developers.google.com/chart/interactive/docs/roles</a>	

## Configuration Options

Name	
aggregationTarget	<p>How multiple data selections are rolled up into tooltips:</p> <ul style="list-style-type: none"> <li>'category': Group selected data by x-value.</li> <li>'series': Group selected data by series.</li> <li>'auto': Group selected data by x-value if all selections have the same x-value, otherwise.</li> <li>'none': Show only one tooltip per selection.</li> </ul> <p><b>aggregationTarget</b> will often be used in tandem with <b>selection</b>. e.g.:</p>

	<pre>var options = {   // Allow multiple   // simultaneous selections.   selectionMode: 'multiple',   // Trigger tooltips   // on selections.   tooltip: {trigger: 'selection'},   // Group selections   // by x-value.   aggregationTarget: 'category', };</pre> <p><b>Type:</b> string <b>Default:</b> 'auto'</p>
animation.duration	<p>The duration of the animation, in milliseconds. For details, see <a href="https://developers.google.com/chart/interactive/docs/animation">https://developers.google.com/chart/interactive/docs/animation</a></p> <p><b>Type:</b> number <b>Default:</b> 0</p>
animation.easing	<p>The easing function applied to the animation. The following options are available:</p> <ul style="list-style-type: none"> <li>• 'linear' - Constant speed.</li> <li>• 'in' - Ease in - Start slow and speed up.</li> <li>• 'out' - Ease out - Start fast and slow down.</li> <li>• 'inAndOut' - Ease in and out - Start slow, speed up, then slow down.</li> </ul> <p><b>Type:</b> string <b>Default:</b> 'linear'</p>
animation.startup	<p>Determines if the chart will animate on the initial draw. If <b>true</b>, the chart will animate to its final state.</p> <p><b>Type:</b> boolean <b>Default:</b> false</p>
annotations.boxStyle	<p>For charts that support <a href="https://developers.google.com/chart/interactive/docs/annotations">annotations</a> (<a href="https://developers.google.com/chart/interactive/docs/annotations">https://developers.google.com/chart/interactive/docs/annotations</a>), the <b>annotations.boxStyle</b> object controls the appearance of the annotation boxes.</p> <pre>var options = {   annotations: {     boxStyle: {       // Color of the box outline.       stroke: '#888',       // Thickness of the box outline.       strokeWidth: 1,</pre>

```

// x-radius of the corner curvature.
rx: 10,
// y-radius of the corner curvature.
ry: 10,
// Attributes for linear gradient fill.
gradient: {
  // Start color for gradient.
  color1: '#fbf6a7',
  // Finish color for gradient.
  color2: '#33b679',
  // Where on the boundary to start and
  // end the color1/color2 gradient,
  // relative to the upper left corner
  // of the boundary.
  x1: '0%', y1: '0%',
  x2: '100%', y2: '100%',
  // If true, the boundary for x1,
  // y1, x2, and y2 is the box. If
  // false, it's the entire chart.
  useObjectBoundingBoxUnits: true
}
}
}
};

```



This option is currently supported for area, bar, column, combo, and pie charts, and is also supported by the [Annotation Chart](https://developers.google.com/chart/interactive/docs/gallery) (<https://developers.google.com/chart/interactive/docs/gallery>).

**Type:** object

**Default:** null

annotations.datum

For charts that support [annotations](https://developers.google.com/chart/interactive/docs/gallery) (<https://developers.google.com/chart/interactive/docs/gallery>), the `annotations.datum` object lets you override Google Charts' default behavior for individual data elements (such as values displayed with each bar). You can specify the color with `annotations.datum.stem.color`, the stem length with `annotations.datum.stem.length`, and the style with `annotations.datum.stem.style`.

**Type:** object

**Default:** color is "black"; length is 12; style is "point".



annotations.domain	<p>For charts that support <a href="https://developers.google.com/chart/interactive/docs/annotations">annotations</a> (<a href="https://developers.google.com/chart/interactive/docs/annotations">https://developers.google.com/chart/interactive/docs/annotations</a>), the <code>annotations.domain</code> object lets you override Google Charts' domain (the major axis of the chart, such as the X axis on a typical line chart). You can override the domain color with <code>annotations.domain.stem.color</code>, the stem length with <code>annotations.domain.stem.length</code>, and the style with <code>annotations.domain.stem.style</code>.</p> <p><b>Type:</b> object</p> <p><b>Default:</b> color is "black"; length is 5; style is "point".</p>
annotations.highContrast	<p>For charts that support <a href="https://developers.google.com/chart/interactive/docs/annotations">annotations</a> (<a href="https://developers.google.com/chart/interactive/docs/annotations">https://developers.google.com/chart/interactive/docs/annotations</a>), the <code>annotations.highContrast</code> boolean lets you override Google Charts' default annotation color. By default, <code>annotations.highContrast</code> is true, which uses a color with good contrast: light colors on dark backgrounds, and dark colors on light backgrounds. Set <code>annotations.highContrast</code> to false and don't specify your own color, and you will use the default series color for the annotation:</p> <div data-bbox="761 734 1233 981"> </div> <p><b>Type:</b> boolean</p> <p><b>Default:</b> true</p>
annotations.stem	<p>For charts that support <a href="https://developers.google.com/chart/interactive/docs/annotations">annotations</a> (<a href="https://developers.google.com/chart/interactive/docs/annotations">https://developers.google.com/chart/interactive/docs/annotations</a>), the <code>annotations.stem</code> object lets you override Google Charts' default annotation stem color with <code>annotations.stem.color</code> and the stem length with <code>annotations.stem.length</code>. Note that the stem length option has no effect on annotations with <code>style</code> set to <code>'point'</code>. For line annotations, the stem length is always the same as the text, and the stem extends across the entire chart.</p> <p><b>Type:</b> object</p> <p><b>Default:</b> color is "black"; length is 5 for domain annotations and 10 for other annotations.</p>
annotations.style	<p>For charts that support <a href="https://developers.google.com/chart/interactive/docs/annotations">annotations</a> (<a href="https://developers.google.com/chart/interactive/docs/annotations">https://developers.google.com/chart/interactive/docs/annotations</a>), the <code>annotations.style</code> option lets you override Google Charts' default annotation style, which is either <code>'line'</code> or <code>'point'</code>.</p> <p><b>Type:</b> string</p> <p><b>Default:</b> 'point'</p>
annotations.textStyle	<p>For charts that support <a href="https://developers.google.com/chart/interactive/docs/annotations">annotations</a> (<a href="https://developers.google.com/chart/interactive/docs/annotations">https://developers.google.com/chart/interactive/docs/annotations</a>), the <code>annotations.textStyle</code> object controls the appearance of the annotation text.</p> <pre>var options = {   annotations: {     textStyle: {       fontName: 'Times-Roman',       fontSize: 18,       bold: true,</pre>

```

        italic: true,
        // The color of the text.
        color: '#871b47',
        // The color of the text outline.
        auraColor: '#d799ae',
        // The transparency of the text.
        opacity: 0.8
    }
}
};

```



This option is currently supported for area, bar, column, combo, and line charts. It is not supported by the [Annotation Chart](https://developers.google.com/chart/interactive/docs/gallery/AnnotationChart) (<https://developers.google.com/chart/interactive/docs/gallery/AnnotationChart>).

**Type:** object

**Default:** null

areaOpacity

The default opacity of the colored area under an area chart series is fully opaque. To specify opacity for an individual series, set the `areaOpacity` property.

**Type:** number, 0.0–1.0

**Default:** 0.3

axisTitlesPosition

Where to place the axis titles, compared to the chart area. Supported values are:

- in - Draw the axis titles inside the chart area.
- out - Draw the axis titles outside the chart area.
- none - Omit the axis titles.

**Type:** string

**Default:** 'out'

backgroundColor

The background color for the main area of the chart. Can be either a string (e.g., 'red' or '#00cc00'), or an object with the following properties:

**Type:** string or object

**Default:** 'white'

backgroundColor.stroke	<p>The color of the chart border, as an HTML color string.</p> <p><b>Type:</b> string <b>Default:</b> '#666'</p>
backgroundColor.strokeWidth	<p>The border width, in pixels.</p> <p><b>Type:</b> number <b>Default:</b> 0</p>
backgroundColor.fill	<p>The chart fill color, as an HTML color string.</p> <p><b>Type:</b> string <b>Default:</b> 'white'</p>
chartArea	<p>An object with members to configure the placement and size of drawn, excluding axis and legends). Two formats are supported. A simple number is a value in pixels; a number followed by % is a percentage. Example: <code>{left:20, top:0, width: '50%', height: '75%' }</code></p> <p><b>Type:</b> object <b>Default:</b> null</p>
chartArea.backgroundColor	<p>Chart area background color. When a string is used, it can be either an English color name or a hex string. When an object is used, the following properties are supported:</p> <ul style="list-style-type: none"> <li><b>stroke:</b> the color, provided as a hex string or English color name</li> <li><b>strokeWidth:</b> if provided, draws a border around the chart area (color of <b>stroke</b>).</li> </ul> <p><b>Type:</b> string or object <b>Default:</b> 'white'</p>
chartArea.left	<p>How far to draw the chart from the left border.</p> <p><b>Type:</b> number or string <b>Default:</b> auto</p>
chartArea.top	<p>How far to draw the chart from the top border.</p> <p><b>Type:</b> number or string <b>Default:</b> auto</p>
chartArea.width	<p>Chart area width.</p> <p><b>Type:</b> number or string <b>Default:</b> auto</p>
chartArea.height	<p>Chart area height.</p> <p><b>Type:</b> number or string <b>Default:</b> auto</p>

colors	<p>The colors to use for the chart elements. An array of strings, where each string represents a color, for example: <code>colors: [ 'red', '#004411' ]</code>.</p> <p><b>Type:</b> Array of strings  <b>Default:</b> default colors</p>
crosshair	<p>An object containing the <a href="https://developers.google.com/chart/interactive/docs/crosshair">crosshair</a> properties for the chart.</p> <p><b>Type:</b> object  <b>Default:</b> null</p>
crosshair.color	<p>The crosshair color, expressed as either a color name (e.g., "blue") or a hex code (e.g., "#0000FF").</p> <p><b>Type:</b> string  <b>Default:</b> default</p>
crosshair.focused	<p>An object containing the crosshair properties upon focus. Example: <code>crosshair: { focused: { color: '#3bc371', opacity: 0.5 } }</code></p> <p><b>Type:</b> object  <b>Default:</b> default</p>
crosshair.opacity	<p>The crosshair opacity, with 0.0 being fully transparent and 1.0 being fully opaque.</p> <p><b>Type:</b> number  <b>Default:</b> 1.0</p>
crosshair.orientation	<p>The crosshair orientation, which can be 'vertical' for vertical hair only, or 'both' for traditional crosshairs.</p> <p><b>Type:</b> string  <b>Default:</b> 'both'</p>
crosshair.selected	<p>An object containing the crosshair properties upon selection. Example: <code>crosshair: { selected: { color: '#3bc371', opacity: 0.5 } }</code></p> <p><b>Type:</b> object  <b>Default:</b> default</p>
crosshair.trigger	<p>When to display crosshairs: on 'focus', 'selection', or 'both'.</p> <p><b>Type:</b> string  <b>Default:</b> 'both'</p>
dataOpacity	<p>The transparency of data points, with 1.0 being completely opaque. For histogram, bar, and column charts, this refers to the visible data rectangles in the others. In charts where <i>selecting data</i> creates circles, this refers to the circles that appear upon hover or selection. This option has no effect on other charts. (To change the opacity of the crosshairs, see <a href="https://developers.google.com/chart/interactive/docs/crosshair">crosshair</a>.)</p>

	<p><b>Type:</b> number <b>Default:</b> 1.0</p>
enableInteractivity	<p>Whether the chart throws user-based events or reacts to user input. It will throw 'select' or other interaction-based events (but <i>will</i> throw <i>mouseover</i> or otherwise change depending on user input).</p> <p><b>Type:</b> boolean <b>Default:</b> true</p>
explorer	<p>The <b>explorer</b> option allows users to pan and zoom Google charts. By default, the explorer is enabled, enabling users to pan horizontally and vertically and in and out by scrolling.</p> <p>This feature is <b>experimental</b> and may change in future releases.</p> <p>★ <b>Note:</b> The explorer only works with continuous axes (such as numerical axes).</p> <p><b>Type:</b> object <b>Default:</b> null</p>
explorer.actions	<p>The Google Charts explorer supports three actions:</p> <ul style="list-style-type: none"> <li>• <b>dragToPan:</b> Drag to pan around the chart horizontally and vertically. To only pan horizontally, use <b>explorer: { axis: 'horizontal' }</b>.</li> <li>• <b>dragToZoom:</b> The explorer's default behavior is to zoom in and out. To only zoom horizontally, use <b>explorer: { actions: ['dragToZoom', 'rightClickToReset'], axis: 'horizontal' }</b>. A rectangular area zooms into that area. We recommend using <b>dragToZoom</b> whenever <b>dragToZoom</b> is used. See <b>explorer.maxZoomIn</b> and <b>explorer.zoomDelta</b> for zoom customizations.</li> <li>• <b>rightClickToReset:</b> Right clicking on the chart returns it to the original view.</li> </ul> <p><b>Type:</b> Array of strings <b>Default:</b> ['dragToPan', 'rightClickToReset']</p>
explorer.axis	<p>By default, users can pan both horizontally and vertically when they want to. To only pan horizontally, use <b>explorer: { axis: 'horizontal' }</b>. To only pan vertically, use <b>explorer: { axis: 'vertical' }</b>.</p> <p><b>Type:</b> string <b>Default:</b> both horizontal and vertical panning</p>
explorer.keepInBounds	<p>By default, users can pan all around, regardless of where the data is. To keep the chart within the original chart, use <b>explorer: { keepInBounds: true }</b>.</p> <p><b>Type:</b> boolean <b>Default:</b> false</p>
explorer.maxZoomIn	<p>The maximum that the explorer can zoom in. By default, users can zoom in to 25% of the original view. Setting <b>explorer: { maxZoomIn: 0.5 }</b> will allow them to see 50% of the original view.</p>

	<p>zoom in only far enough to see half of the original view.</p> <p><b>Type:</b> number <b>Default:</b> 0.25</p>
explorer.maxZoomOut	<p>The maximum that the explorer can zoom out. By default, users that the chart will take up only 1/4 of the available space. Setting would let users zoom out far enough that the chart would take u</p> <p><b>Type:</b> number <b>Default:</b> 4</p>
explorer.zoomDelta	<p>When users zoom in or out, <code>explorer.zoomDelta</code> determines the number, the smoother and slower the zoom.</p> <p><b>Type:</b> number <b>Default:</b> 1.5</p>
focusTarget	<p>The type of the entity that receives focus on mouse hover. Also mouse click, and which data table element is associated with ev</p> <ul style="list-style-type: none"> <li>• 'datum' - Focus on a single data point. Correlates to a cell in t</li> <li>• 'category' - Focus on a grouping of all data points along the r data table.</li> </ul> <p>In focusTarget 'category' the tooltip displays all the category val values of different series.</p> <p><b>Type:</b> string <b>Default:</b> 'datum'</p>
fontSize	<p>The default font size, in pixels, of all text in the chart. You can o chart elements.</p> <p><b>Type:</b> number <b>Default:</b> automatic</p>
fontName	<p>The default font face for all text in the chart. You can override th elements.</p> <p><b>Type:</b> string <b>Default:</b> 'Arial'</p>
forceIframe	<p>Draws the chart inside an inline frame. (Note that on IE8, this op in i-frames.)</p> <p><b>Type:</b> boolean <b>Default:</b> false</p>
hAxis	<p>An object with members to configure various horizontal axis ele object, you can use object literal notation, as shown here:</p>



	<p>For date axis labels, this is a subset of the date formatting <a href="http://icu-project.org/apiref/icu4c/classSimpleDateFormat.html">ICU</a> (http://icu-project.org/apiref/icu4c/classSimpleDateFormat.html). <code>{format: 'MMM d, y'}</code> will display the value "Jul 1, 2011" for the date July 1, 2011.</p> <p>The actual formatting applied to the label is derived from the locale. For more details, see <a href="https://developers.google.com/chart/interactive/docs/library#loading-charts-with-a-specific-locale">loading charts with a specific locale</a> (https://developers.google.com/chart/interactive/docs/library#loading-charts-with-a-specific-locale).</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-axis">continuous</a> (https://developers.google.com/chart/interactive/docs/customizing-axis).</p> <p><b>Type:</b> string <b>Default:</b> auto</p>
hAxis.gridlines	<p>An object with members to configure the gridlines on the horizontal axis. If you use object literal notation, as shown here:</p> <pre>{color: '#333', count: 4}</pre> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-axis">continuous</a> (https://developers.google.com/chart/interactive/docs/customizing-axis).</p> <p><b>Type:</b> object <b>Default:</b> null</p>
hAxis.gridlines.color	<p>The color of the horizontal gridlines inside the chart area. Specified as a hex color string.</p> <p><b>Type:</b> string <b>Default:</b> '#CCC'</p>
hAxis.gridlines.count	<p>The number of horizontal gridlines inside the chart area. Minimum value is 1. If not specified, the chart will automatically compute the number of gridlines.</p> <p><b>Type:</b> number <b>Default:</b> 5</p>
hAxis.gridlines.units	<p>Overrides the default format for various aspects of date/datetime labels on the horizontal axis. Allows formatting for years, months, days, and milliseconds.</p> <p>General format is:</p> <pre>gridlines: {   units: {     years: {format: [/format strings here/]},     months: {format: [/format strings here/]},     days: {format: [/format strings here/]},     hours: {format: [/format strings here/]}   } }</pre>



	<pre> minutes: {format: [/*format strings here*/] seconds: {format: [/*format strings here*/] milliseconds: {format: [/*format strings he } } </pre> <p>Additional information can be found in <a href="https://developers.google.com/chart/interactive/docs/datesa">Dates and Times</a> (<a href="https://developers.google.com/chart/interactive/docs/datesa">https://developers.google.com/chart/interactive/docs/datesa</a>)</p> <p><b>Type:</b> object <b>Default:</b> null</p>
hAxis.minorGridlines	<p>An object with members to configure the minor gridlines on the hAxis.gridlines option.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/custom">continuous</a> (<a href="https://developers.google.com/chart/interactive/docs/custom">https://developers.google.com/chart/interactive/docs/custom</a>)</p> <p><b>Type:</b> object <b>Default:</b> null</p>
hAxis.minorGridlines.color	<p>The color of the horizontal minor gridlines inside the chart area.</p> <p><b>Type:</b> string <b>Default:</b> A blend of the gridline and background colors</p>
hAxis.minorGridlines.count	<p>The number of horizontal minor gridlines between two regular g</p> <p><b>Type:</b> number <b>Default:</b> 0</p>
hAxis.minorGridlines.units	<p>Overrides the default format for various aspects of date/datetin with chart computed minorGridlines. Allows formatting for year seconds, and milliseconds.</p> <p>General format is:</p> <pre> gridlines: {   units: {     years: {format: [/*format strings here*/]},     months: {format: [/*format strings here*/]},     days: {format: [/*format strings here*/]},     hours: {format: [/*format strings here*/]},     minutes: {format: [/*format strings here*/]},     seconds: {format: [/*format strings here*/]},     milliseconds: {format: [/*format strings he   } } </pre>

	<p>Additional information can be found in <a href="https://developers.google.com/chart/interactive/docs/datesa">Dates and Times</a> (<a href="https://developers.google.com/chart/interactive/docs/datesa">https://developers.google.com/chart/interactive/docs/datesa</a>)</p> <p><b>Type:</b> object <b>Default:</b> null</p>
hAxis.logScale	<p><b>hAxis</b> property that makes the horizontal axis a logarithmic scale. Set to true for yes.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/custom">continuous</a> (<a href="https://developers.google.com/chart/interactive/docs/custom">https://developers.google.com/chart/interactive/docs/custom</a>)</p> <p><b>Type:</b> boolean <b>Default:</b> false</p>
hAxis.scaleType	<p><b>hAxis</b> property that makes the horizontal axis a logarithmic scale.</p> <ul style="list-style-type: none"> <li>• null - No logarithmic scaling is performed.</li> <li>• 'log' - Logarithmic scaling. Negative and zero values are not supported. Setting <b>hAxis: { logscale: true }</b>.</li> <li>• 'mirrorLog' - Logarithmic scaling in which negative and zero values are supported. A negative number is the negative of the log of the absolute value of the number on a linear scale.</li> </ul> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/custom">continuous</a> (<a href="https://developers.google.com/chart/interactive/docs/custom">https://developers.google.com/chart/interactive/docs/custom</a>)</p> <p><b>Type:</b> string <b>Default:</b> null</p>
hAxis.textPosition	<p>Position of the horizontal axis text, relative to the chart area. Supported values are 'in', 'out', and 'none'.</p> <p><b>Type:</b> string <b>Default:</b> 'out'</p>
hAxis.textStyle	<p>An object that specifies the horizontal axis text style. The object has the following properties:</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is in pixels.</p> <p><b>Type:</b> object</p>

	<p>Default: {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: &lt;global-font-size&gt;}</p>
hAxis.ticks	<p>Replaces the automatically generated X-axis ticks with the specified ticks. Each tick should be either a valid tick value (such as a number, date, date object, or string), or an object, it should have a v property for the tick value, and an f property for the label string to be displayed as the label.</p> <p>Examples:</p> <ul style="list-style-type: none"> <li>• hAxis: { ticks: [5,10,15,20] }</li> <li>• hAxis: { ticks: [{v:32, f:'thirty two'}, {v:33, f:'thirty three'}] }</li> <li>• hAxis: { ticks: [new Date(2014,3,15), new Date(2014,3,16)] }</li> <li>• hAxis: { ticks: [16, {v:32, f:'thirty two'}, {v:33, f:'thirty three'}] }</li> </ul> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#continuous-axis">continuous</a> chart (<a href="https://developers.google.com/chart/interactive/docs/customizing-charts#continuous-axis">https://developers.google.com/chart/interactive/docs/customizing-charts#continuous-axis</a>)</p> <p>Type: Array of elements Default: auto</p>
hAxis.title	<p>hAxis property that specifies the title of the horizontal axis.</p> <p>Type: string Default: null</p>
hAxis.titleTextStyle	<p>An object that specifies the horizontal axis title text style. The object has the following properties:</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is in pixels.</p> <p>Type: object Default: {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: &lt;global-font-size&gt;}</p>
hAxis.allowContainerBoundaryTextCutoff	<p>If false, will hide outermost labels rather than allow them to be cropped. If true, will allow label cropping.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete-axis">discrete</a> chart (<a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete-axis">https://developers.google.com/chart/interactive/docs/customizing-charts#discrete-axis</a>)</p>

	<p><b>Type:</b> boolean <b>Default:</b> false</p>
hAxis.slantedText	<p>If true, draw the horizontal axis text at an angle, to help fit more horizontal axis text upright. Default behavior is to slant text if it is too long. This option is available only when the <code>hAxis.textPosition</code> is <code>auto</code>.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete">discrete</a> chart.</p> <p><b>Type:</b> boolean <b>Default:</b> automatic</p>
hAxis.slantedTextAngle	<p>The angle of the horizontal axis text, if it's drawn slanted. Ignore this option if the chart is in auto mode, and the chart decided to draw the text horizontal.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete">discrete</a> chart.</p> <p><b>Type:</b> number, 1–90 <b>Default:</b> 30</p>
hAxis.maxAlternation	<p>Maximum number of levels of horizontal axis text. If axis text labels are too long, the chart might shift neighboring labels up or down in order to fit labels on the chart. The maximum number of levels to use; the server can use fewer levels, if needed.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete">discrete</a> chart.</p> <p><b>Type:</b> number <b>Default:</b> 2</p>
hAxis.maxTextLines	<p>Maximum number of lines allowed for the text labels. Labels can wrap, and the number of lines is, by default, limited by the height of the chart.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete">discrete</a> chart.</p> <p><b>Type:</b> number <b>Default:</b> auto</p>
hAxis.minTextSpacing	<p>Minimum horizontal spacing, in pixels, allowed between two adjacent labels. If labels are spaced too densely, or they are too long, the spacing can drop below the minimum. If the spacing is too small, the label-unclutter measures will be applied (e.g. truncating the label).</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#discrete">discrete</a> chart.</p> <p><b>Type:</b> number <b>Default:</b> The value of <code>hAxis.textStyle.fontSize</code></p>
hAxis.showTextEvery	<p>How many horizontal axis labels to show, where 1 means show all labels.</p>

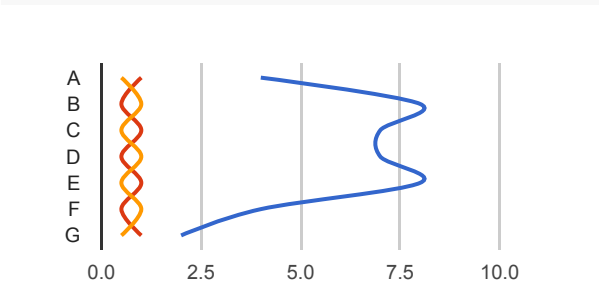
	<p>label, and so on. Default is to try to show as many labels as possible.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-axis-labels">discrete</a> chart.</p> <p><b>Type:</b> number <b>Default:</b> automatic</p>
<code>hAxis.maxValue</code>	<p>Moves the max value of the horizontal axis to the specified value. Ignored if this is set to a value smaller than the maximum x-value. <code>hAxis.viewWindow.max</code> overrides this property.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-axis-labels">continuous</a> chart.</p> <p><b>Type:</b> number <b>Default:</b> automatic</p>
<code>hAxis.minValue</code>	<p>Moves the min value of the horizontal axis to the specified value. Ignored if this is set to a value greater than the minimum x-value. <code>hAxis.viewWindow.min</code> overrides this property.</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-axis-labels">continuous</a> chart.</p> <p><b>Type:</b> number <b>Default:</b> automatic</p>
<code>hAxis.viewWindowMode</code>	<p>Specifies how to scale the horizontal axis to render the values when string values are supported:</p> <ul style="list-style-type: none"> <li>'pretty' - Scale the horizontal values so that the maximum and minimum are a bit inside the left and right of the chart area. This will cause <code>hAxis.viewWindow.max</code> to be ignored.</li> <li>'maximized' - Scale the horizontal values so that the maximum and minimum are at the left and right of the chart area. This will cause <code>hAxis.viewWindow.min</code> and <code>hAxis.viewWindow.max</code> to be ignored.</li> <li>'explicit' - A deprecated option for specifying the left and right values. (Deprecated because it's redundant with <code>hAxis.viewWindow.min</code> and <code>hAxis.viewWindow.max</code>.) Data values outside these values will be truncated. <code>hAxis.viewWindow</code> object describing the maximum and minimum values.</li> </ul> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-axis-labels">continuous</a> chart.</p> <p><b>Type:</b> string <b>Default:</b> Equivalent to 'pretty', but <code>hAxis.viewWindow.min</code> and <code>hAxis.viewWindow.max</code> take precedence if used.</p>
<code>hAxis.viewWindow</code>	<p>Specifies the cropping range of the horizontal axis.</p>

	<p><b>Type:</b> object</p> <p><b>Default:</b> null</p>
hAxis.viewWindow.max	<ul style="list-style-type: none"> <li>For a <a href="https://developers.google.com/chart/interactive/docs/customizing-view-window">continuous</a> (https://developers.google.com/chart/interactive/docs/customizing-view-window) The maximum horizontal data value to render.</li> <li>For a <a href="https://developers.google.com/chart/interactive/docs/customizing-view-window">discrete</a> (https://developers.google.com/chart/interactive/docs/customizing-view-window) The zero-based row index where the cropping window ends. <code>index &gt; max</code> will be cropped out. In conjunction with <code>vAxis.viewWindowMode</code> range <code>[min, max)</code> that denotes the element indices to display. In other words, <code>min &lt;= index &lt; max</code> will be displayed.</li> </ul> <p>Ignored when <code>hAxis.viewWindowMode</code> is 'pretty' or 'maximized'.</p> <p><b>Type:</b> number</p> <p><b>Default:</b> auto</p>
hAxis.viewWindow.min	<ul style="list-style-type: none"> <li>For a <a href="https://developers.google.com/chart/interactive/docs/customizing-view-window">continuous</a> (https://developers.google.com/chart/interactive/docs/customizing-view-window) The minimum horizontal data value to render.</li> <li>For a <a href="https://developers.google.com/chart/interactive/docs/customizing-view-window">discrete</a> (https://developers.google.com/chart/interactive/docs/customizing-view-window) The zero-based row index where the cropping window begins. <code>index &lt; min</code> will be cropped out. In conjunction with <code>vAxis.viewWindowMode</code> range <code>[min, max)</code> that denotes the element indices to display. In other words, <code>min &lt;= index &lt; max</code> will be displayed.</li> </ul> <p>Ignored when <code>hAxis.viewWindowMode</code> is 'pretty' or 'maximized'.</p> <p><b>Type:</b> number</p> <p><b>Default:</b> auto</p>
height	<p>Height of the chart, in pixels.</p> <p><b>Type:</b> number</p> <p><b>Default:</b> height of the containing element</p>
interpolateNulls	<p>Whether to guess the value of missing points. If true, it will guess the value based on neighboring points. If false, it will leave a break in the line at the missing point.</p> <p>This is <b>not</b> supported by <a href="https://developers.google.com/chart/interactive/docs/gallery">Area</a> (https://developers.google.com/chart/interactive/docs/gallery). The <code>isStacked</code> property must be set to <code>true</code> or <code>'percent'</code> or <code>'relative'</code> or <code>'absolute'</code>.</p> <p><b>Type:</b> boolean</p> <p><b>Default:</b> false</p>

isStacked	<p>If set to true, stacks the elements for all series at each domain value. (https://developers.google.com/chart/interactive/docs/gallery, (https://developers.google.com/chart/interactive/docs/gallery, (https://developers.google.com/chart/interactive/docs/gallery, Charts reverses the order of legend items to better correspond to the order of the data series (E.g. series 0 will be the bottom-most legend item). This <b>does not</b> apply to 100% stacking. (https://developers.google.com/chart/interactive/docs/gallery, 100% stacking only supports data values of type <b>number</b>, and not <b>string</b>).</p> <p>The <b>isStacked</b> option also supports 100% stacking, where the values are rescaled to add up to 100%.</p> <p>The options for <b>isStacked</b> are:</p> <ul style="list-style-type: none"> <li>• <b>false</b> – elements will not stack. This is the default option.</li> <li>• <b>true</b> – stacks elements for all series at each domain value.</li> <li>• <b>'percent'</b> – stacks elements for all series at each domain value, add up to 100%, with each element's value calculated as a percentage of the total.</li> <li>• <b>'relative'</b> – stacks elements for all series at each domain value, they add up to 1, with each element's value calculated as a fraction of the total.</li> <li>• <b>'absolute'</b> – functions the same as <b>isStacked: true</b>.</li> </ul> <p>For 100% stacking, the calculated value for each element will appear on the target axis.</p> <p>The target axis will default to tick values based on the relative values for <b>'relative'</b>, and 0-100% for <b>'percent'</b> (<b>Note:</b> when using the <b>'percent'</b> values are displayed as percentages, however the actual values are not, because the percentage axis ticks are the result of applying a formula to the actual values. When using <b>isStacked: 'percent'</b>, be sure to specify 0-1 scale values). You can customize the gridlines/tick values with the <b>hAxis/vAxis</b> options.</p> <p>100% stacking only supports data values of type <b>number</b>, and not <b>string</b>.</p> <p><b>Type:</b> boolean/string <b>Default:</b> false</p>
legend	<p>An object with members to configure various aspects of the legend. You can use object literal notation, as shown here:</p> <pre>{position: 'top', textStyle: {color: 'blue', fontStyle: 'italic', size: 14}}</pre> <p><b>Type:</b> object <b>Default:</b> null</p>
legend.alignment	<p>Alignment of the legend. Can be one of the following:</p> <ul style="list-style-type: none"> <li>• <b>'start'</b> - Aligned to the start of the area allocated for the legend.</li> <li>• <b>'center'</b> - Centered in the area allocated for the legend.</li> </ul>


	<ul style="list-style-type: none"> <li>'end' - Aligned to the end of the area allocated for the legend</li> </ul> <p>Start, center, and end are relative to the style -- vertical or horizontal. For a 'right' legend, 'start' and 'end' are at the top and bottom, respectively. For a 'left' legend, 'start' and 'end' would be at the left and right of the area, respectively.</p> <p>The default value depends on the legend's position. For 'bottom' legends default to 'start'.</p> <p><b>Type:</b> string  <b>Default:</b> automatic</p>
legend.maxLines	<p>Maximum number of lines in the legend. Set this to a number greater than or equal to 1. Note: The exact logic used to determine the actual number of lines is not guaranteed.</p> <p>This option currently works only when legend.position is 'top'.</p> <p><b>Type:</b> number  <b>Default:</b> 1</p>
legend.position	<p>Position of the legend. Can be one of the following:</p> <ul style="list-style-type: none"> <li>'bottom' - Below the chart.</li> <li>'left' - To the left of the chart, provided the left axis has no secondary y-axis. If you want to place the legend on the left, use the option <b>targetAxisIndex</b>.</li> <li>'in' - Inside the chart, by the top left corner.</li> <li>'none' - No legend is displayed.</li> <li>'right' - To the right of the chart. Incompatible with the <b>vAxes</b> option.</li> <li>'top' - Above the chart.</li> </ul> <p><b>Type:</b> string  <b>Default:</b> 'right'</p>
legend.textStyle	<p>An object that specifies the legend text style. The object has the following properties:</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is the size of the font in pixels.</p> <p><b>Type:</b> object  <b>Default:</b> {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: &lt;global-font-size&gt;}</p>



lineDashStyle	<p>The on-and-off pattern for dashed lines. For instance, <code>[4, 4]</code> will repeat a 4-length dash, and <code>[5, 1, 3]</code> will repeat a 5-length dash, a 1-length gap, a 1-length dash, and a 3-length gap. See <a href="https://developers.google.com/chart/interactive/docs/lines#dash">Dashed Lines</a> (<a href="https://developers.google.com/chart/interactive/docs/lines#dash">https://developers.google.com/chart/interactive/docs/lines#dash</a>)</p> <p><b>Type:</b> Array of numbers  <b>Default:</b> null</p>
lineWidth	<p>Data line width in pixels. Use zero to hide all lines and show only individual series using the <b>series</b> property.</p> <p><b>Type:</b> number  <b>Default:</b> 2</p>
orientation	<p>The orientation of the chart. When set to <b>'vertical'</b>, rotates (for instance) a column chart becomes a bar chart, and an area chart becomes a line chart.</p>  <p><b>Type:</b> string  <b>Default:</b> 'horizontal'</p>
pointShape	<p>The shape of individual data elements: 'circle', 'triangle', 'square', etc. See <a href="https://developers.google.com/chart/interactive/docs/points">points documentation</a> (<a href="https://developers.google.com/chart/interactive/docs/points">https://developers.google.com/chart/interactive/docs/points</a>)</p> <p><b>Type:</b> string  <b>Default:</b> 'circle'</p>
pointSize	<p>Diameter of displayed points in pixels. Use zero to hide all point series using the <b>series</b> property. If you're using a <a href="https://developers.google.com/chart/interactive/docs/gallery#trendline">trendline</a> (<a href="https://developers.google.com/chart/interactive/docs/gallery#trendline">https://developers.google.com/chart/interactive/docs/gallery#trendline</a>), this option will affect the width of the trendline unless you override it with the <b>trendlines.n.pointsVisible</b> option.</p> <p><b>Type:</b> number  <b>Default:</b> 0</p>
pointsVisible	<p>Determines whether points will be displayed. Set to <b>false</b> to hide points for individual series using the <b>series</b> property. If you're using a <a href="https://developers.google.com/chart/interactive/docs/gallery#trendline">trendline</a> (<a href="https://developers.google.com/chart/interactive/docs/gallery#trendline">https://developers.google.com/chart/interactive/docs/gallery#trendline</a>), this option will affect the visibility of the points on all trendlines unless you override it with the <b>trendlines.n.pointsVisible</b> option.</p> <p>This can also be overridden using the <a href="https://developers.google.com/chart/interactive/docs/roles#style">style role</a> (<a href="https://developers.google.com/chart/interactive/docs/roles#style">https://developers.google.com/chart/interactive/docs/roles#style</a>) with <code>{visible: true}</code>.</p>

	<p><b>Type:</b> boolean</p> <p><b>Default:</b> true</p>
reverseCategories	<p>If set to true, will draw series from right to left. The default is to</p> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/custom">discrete</a> (<a href="https://developers.google.com/chart/interactive/docs/custom">https://developers.google.com/chart/interactive/docs/custom</a>)</p> <p><b>Type:</b> boolean</p> <p><b>Default:</b> false</p>
selectionMode	<p>When <b>selectionMode</b> is 'multiple', users may select multiple</p> <p><b>Type:</b> string</p> <p><b>Default:</b> 'single'</p>
series	<p>An array of objects, each describing the format of the corresponding values for a series, specify an empty object {}. If a series or a value be used. Each object supports the following properties:</p> <ul style="list-style-type: none"> <li>• <b>annotations</b> - An object to be applied to annotations for this series. For instance, the <b>textStyle</b> for the series:</li> </ul> <pre> series: {   0: {     annotations: {       textStyle: {fontSize: 12, color: 'red'}     }   } } </pre> <p>See the various <b>annotations</b> options for a more complete</p> <ul style="list-style-type: none"> <li>• <b>areaOpacity</b> - Overrides the global <b>areaOpacity</b> for this series.</li> <li>• <b>color</b> - The color to use for this series. Specify a valid HTML color.</li> <li>• <b>labelInLegend</b> - The description of the series to appear in the legend.</li> <li>• <b>lineDashStyle</b> - Overrides the global <b>lineDashStyle</b> value for this series.</li> <li>• <b>lineWidth</b> - Overrides the global <b>lineWidth</b> value for this series.</li> <li>• <b>pointShape</b> - Overrides the global <b>pointShape</b> value for this series.</li> <li>• <b>pointSize</b> - Overrides the global <b>pointSize</b> value for this series.</li> <li>• <b>pointsVisible</b> - Overrides the global <b>pointsVisible</b> value for this series.</li> <li>• <b>targetAxisIndex</b> - Which axis to assign this series to, where 0 is the horizontal axis and 1 is the vertical axis. Default value is 0; set to 1 to define a chart with series plotted against different axes. At least one series must be allocated to each axis to have different scale for different axes.</li> </ul>

	<ul style="list-style-type: none"> <li>• <b>visibleInLegend</b> - A boolean value, where true means that the series should be visible in the legend and false means that it should not. Default is true.</li> </ul> <p>You can specify either an array of objects, each of which applies to a series, or an object with nested objects. For example, you can specify an object where each child has a numeric key indicating the series index. In the following example, the following two declarations are identical, and the first declares the series from the legend, and the fourth as red and absent from the legend.</p> <pre>series: [   {color: 'black', visibleInLegend: false}, {},   {color: 'red', visibleInLegend: false} ] series: {   0:{color: 'black', visibleInLegend: false},   3:{color: 'red', visibleInLegend: false} }</pre> <p><b>Type:</b> Array of objects, or object with nested objects  <b>Default:</b> {}</p>
theme	<p>A theme is a set of predefined option values that work together to create a specific visual effect. Currently only one theme is available:</p> <ul style="list-style-type: none"> <li>• 'maximized' - Maximizes the area of the chart, and draws the chart area. Sets the following options:</li> </ul> <pre>chartArea: {width: '100%', height: '100%'}, legend: {position: 'in'}, titlePosition: 'in', axisTitlesPosition: 'in', hAxis: {textPosition: 'in'}, vAxis: {textPosition: 'in'}</pre> <p><b>Type:</b> string  <b>Default:</b> null</p>
title	<p>Text to display above the chart.</p> <p><b>Type:</b> string  <b>Default:</b> no title</p>
titlePosition	<p>Where to place the chart title, compared to the chart area. Supported values are:</p> <ul style="list-style-type: none"> <li>• in - Draw the title inside the chart area.</li> <li>• out - Draw the title outside the chart area.</li> <li>• none - Omit the title.</li> </ul> <p><b>Type:</b> string  <b>Default:</b> 'out'</p>

titleTextStyle	<p>An object that specifies the title text style. The object has this format:</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is an integer.</p> <p><b>Type:</b> object  <b>Default:</b> {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: 14, bold: false, italic: false}</p>
tooltip	<p>An object with members to configure various tooltip elements. The object can use object literal notation, as shown here:</p> <pre>{textStyle: {color: '#FF0000'}, showColorCode: true}</pre> <p><b>Type:</b> object  <b>Default:</b> null</p>
tooltip.ignoreBounds	<p>If set to <b>true</b>, allows the drawing of tooltips to flow outside of the chart area.</p> <p><b>Note:</b> This only applies to HTML tooltips. If this is enabled with HTML tooltips, the chart bounds will be cropped. See <a href="https://developers.google.com/chart/interactive/docs/customizing_tooltips">Customizing Tooltip Content</a> (https://developers.google.com/chart/interactive/docs/customizing_tooltips).</p> <p><b>Type:</b> boolean  <b>Default:</b> false</p>
tooltip.isHtml	<p>If set to true, use HTML-rendered (rather than SVG-rendered) tooltips. See <a href="https://developers.google.com/chart/interactive/docs/customizing_tooltips">Customizing Tooltip Content</a> (https://developers.google.com/chart/interactive/docs/customizing_tooltips).</p> <div>  <p><b>Note:</b> customization of the HTML tooltip content via the <a href="https://developers.google.com/chart/interactive/docs/roles#tooltip">tooltip</a> role (https://developers.google.com/chart/interactive/docs/roles#tooltip) in a <a href="https://developers.google.com/chart/interactive/docs/customizing_bubble_chart">Bubble Chart</a> (https://developers.google.com/chart/interactive/docs/customizing_bubble_chart).</p> </div> <p><b>Type:</b> boolean  <b>Default:</b> false</p>
tooltip.showColorCode	<p>If true, show colored squares next to the series information in the tooltip. If <b>focusTarget</b> is set to 'category', otherwise the default is false.</p> <p><b>Type:</b> boolean  <b>Default:</b> automatic</p>

tooltip.textStyle	<p>An object that specifies the tooltip text style. The object has this</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is the size of the font in pixels.</p> <p><b>Type:</b> object  <b>Default:</b> {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: &lt;global-font-size&gt;}</p>
tooltip.trigger	<p>The user interaction that causes the tooltip to be displayed:</p> <ul style="list-style-type: none"> <li>• 'focus' - The tooltip will be displayed when the user hovers over the data point.</li> <li>• 'none' - The tooltip will not be displayed.</li> <li>• 'selection' - The tooltip will be displayed when the user selects the data point.</li> </ul> <p><b>Type:</b> string  <b>Default:</b> 'focus'</p>
vAxes	<p>Specifies properties for individual vertical axes, if the chart has a <b>vAxis</b> object, and can contain all the properties supported by the <b>vAxis</b> object to override any global settings for the same property.</p> <p>To specify a chart with multiple vertical axes, first define a new <b>series.targetAxisIndex</b>, then configure the axis using <b>vAxes</b>. For example, to move series 2 to the right axis and specifies a custom title and text style:</p> <pre>{   series: {     2: {       targetAxisIndex: 1     }   },   vAxes: {     1: {       title: 'Losses',       textStyle: {color: 'red'}     }   } }</pre>

	<p>This property can be either an object or an array: the object is a numeric label that specifies the axis that it defines--this is the function of objects, one per axis. For example, the following array-style notation is shown above:</p> <pre>vAxes: [   {}, // Nothing specified for axis 0   {     title:'Losses',     textStyle: {color: 'red'} // Axis 1   } ]</pre> <p><b>Type:</b> Array of object, or object with child objects  <b>Default:</b> null</p>
vAxis	<p>An object with members to configure various vertical axis elements; you can use object literal notation, as shown here:</p> <pre>{title: 'Hello', titleTextStyle: {color: '#FF0000'}}</pre> <p><b>Type:</b> object  <b>Default:</b> null</p>
vAxis.baseline	<p><b>vAxis</b> property that specifies the baseline for the vertical axis. If the grid line or smaller than the lowest grid line, it will be rounded to the nearest grid line.</p> <p><b>Type:</b> number  <b>Default:</b> automatic</p>
vAxis.baselineColor	<p>Specifies the color of the baseline for the vertical axis. Can be a string or a number. If a string, it can be 'red' or '#00cc00'.</p> <p><b>Type:</b> number  <b>Default:</b> 'black'</p>
vAxis.direction	<p>The direction in which the values along the vertical axis grow. Specify 1 for increasing values and -1 for decreasing values.</p> <p><b>Type:</b> 1 or -1  <b>Default:</b> 1</p>
vAxis.format	<p>A format string for numeric axis labels. This is a subset of the ICU format string (http://icu-project.org/apiref/icu4c/classDecimalFormat.html#format). For example, {format: '#,###%' } will display values "1,000%", "750%", and "0%". It can also supply any of the following:</p> <ul style="list-style-type: none"> <li><b>{format: 'none' }</b>: displays numbers with no formatting</li> </ul>

	<ul style="list-style-type: none"> <li>• <code>{format: 'decimal'}</code>: displays numbers with thousands</li> <li>• <code>{format: 'scientific'}</code>: displays numbers in scientific</li> <li>• <code>{format: 'currency'}</code>: displays numbers in the local cu</li> <li>• <code>{format: 'percent'}</code>: displays numbers as percentages</li> <li>• <code>{format: 'short'}</code>: displays abbreviated numbers (e.g.,</li> <li>• <code>{format: 'long'}</code>: displays numbers as full words (e.g.,</li> </ul> <p>The actual formatting applied to the label is derived from the lo more details, see <a href="https://developers.google.com/chart/interactive/docs/library_">loading charts with a specific locale</a> (<a href="https://developers.google.com/chart/interactive/docs/library_">https://developers.google.com/chart/interactive/docs/library_</a> .</p> <p><b>Type:</b> string <b>Default:</b> auto</p>
<code>vAxis.gridlines</code>	<p>An object with members to configure the gridlines on the vertical object, you can use object literal notation, as shown here:</p> <pre>{color: '#333', count: 4}</pre> <p><b>Type:</b> object <b>Default:</b> null</p>
<code>vAxis.gridlines.color</code>	<p>The color of the vertical gridlines inside the chart area. Specify</p> <p><b>Type:</b> string <b>Default:</b> '#CCC'</p>
<code>vAxis.gridlines.count</code>	<p>The number of vertical gridlines inside the chart area. Minimum compute the number of gridlines.</p> <p><b>Type:</b> number <b>Default:</b> 5</p>
<code>vAxis.gridlines.units</code>	<p>Overrides the default format for various aspects of date/datetin with chart computed gridlines. Allows formatting for years, mor milliseconds.</p> <p>General format is:</p> <pre>gridlines: {   units: {     years: {format: [/*format strings here*/]},     months: {format: [/*format strings here*/]}     days: {format: [/*format strings here*/]}     hours: {format: [/*format strings here*/]}     minutes: {format: [/*format strings here*/]}     seconds: {format: [/*format strings here*/]}</pre>

	<pre>         milliseconds: {format: [/format strings here*/]},       }     } </pre> <p>Additional information can be found in <a href="https://developers.google.com/chart/interactive/docs/datesandtimes">Dates and Times</a> (<a href="https://developers.google.com/chart/interactive/docs/datesandtimes">https://developers.google.com/chart/interactive/docs/datesandtimes</a>)</p> <p><b>Type:</b> object <b>Default:</b> null</p>
vAxis.minorGridlines	<p>An object with members to configure the minor gridlines on the vAxis.gridlines option.</p> <p><b>Type:</b> object <b>Default:</b> null</p>
vAxis.minorGridlines.color	<p>The color of the vertical minor gridlines inside the chart area. Specified as a CSS color string.</p> <p><b>Type:</b> string <b>Default:</b> A blend of the gridline and background colors</p>
vAxis.minorGridlines.count	<p>The number of vertical minor gridlines between two regular gridlines.</p> <p><b>Type:</b> number <b>Default:</b> 0</p>
vAxis.minorGridlines.units	<p>Overrides the default format for various aspects of date/datetime with chart computed minorGridlines. Allows formatting for year, month, seconds, and milliseconds.</p> <p>General format is:</p> <pre> gridlines: {   units: {     years: {format: [/format strings here*/]},     months: {format: [/format strings here*/]},     days: {format: [/format strings here*/]},     hours: {format: [/format strings here*/]},     minutes: {format: [/format strings here*/]},     seconds: {format: [/format strings here*/]},     milliseconds: {format: [/format strings here*/]},   } } </pre> <p>Additional information can be found in <a href="https://developers.google.com/chart/interactive/docs/datesandtimes">Dates and Times</a> (<a href="https://developers.google.com/chart/interactive/docs/datesandtimes">https://developers.google.com/chart/interactive/docs/datesandtimes</a>)</p> <p><b>Type:</b> object</p>



	<p><b>Default:</b> null</p>
vAxis.logScale	<p>If true, makes the vertical axis a logarithmic scale. Note: All values must be positive.</p> <p><b>Type:</b> boolean <b>Default:</b> false</p>
vAxis.scaleType	<p>vAxis property that makes the vertical axis a logarithmic scale</p> <ul style="list-style-type: none"> <li>• null - No logarithmic scaling is performed.</li> <li>• 'log' - Logarithmic scaling. Negative and zero values are not supported. To use logarithmic scaling, set <code>vAxis: { logscale: true }</code>.</li> <li>• 'mirrorLog' - Logarithmic scaling in which negative and zero values are not supported. A negative number is the negative of the log of the absolute value of the number on a linear scale.</li> </ul> <p>This option is only supported for a <a href="https://developers.google.com/chart/interactive/docs/customizing-charts#continuous">continuous</a> chart.</p> <p><b>Type:</b> string <b>Default:</b> null</p>
vAxis.textPosition	<p>Position of the vertical axis text, relative to the chart area. Supported values are 'in' and 'out'.</p> <p><b>Type:</b> string <b>Default:</b> 'out'</p>
vAxis.textStyle	<p>An object that specifies the vertical axis text style. The object has the following properties:</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is in pixels.</p> <p><b>Type:</b> object <b>Default:</b> {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: &lt;global-font-size&gt;}</p>
vAxis.ticks	<p>Replaces the automatically generated Y-axis ticks with the specified ticks. Each tick should be either a valid tick value (such as a number, date, date range, or time range) or an object. If it is an object, it should have a v property for the tick value, and an l property for the label string to be displayed as the label.</p> <p>Examples:</p>

	<ul style="list-style-type: none"> <li>• <code>vAxis: { ticks: [5,10,15,20] }</code></li> <li>• <code>vAxis: { ticks: [{v:32, f:'thirty two'}, {v:32, f:'thirty two'}]</code></li> <li>• <code>vAxis: { ticks: [new Date(2014,3,15), new Date(2014,3,15)] }</code></li> <li>• <code>vAxis: { ticks: [16, {v:32, f:'thirty two'}, {v:32, f:'thirty two'}]</code></li> </ul> <p><b>Type:</b> Array of elements  <b>Default:</b> auto</p>
<code>vAxis.title</code>	<p><code>vAxis</code> property that specifies a title for the vertical axis.</p> <p><b>Type:</b> string  <b>Default:</b> no title</p>
<code>vAxis.titleTextStyle</code>	<p>An object that specifies the vertical axis title text style. The object has the following properties:</p> <pre>{ color: &lt;string&gt;,   fontName: &lt;string&gt;,   fontSize: &lt;number&gt;,   bold: &lt;boolean&gt;,   italic: &lt;boolean&gt; }</pre> <p>The <b>color</b> can be any HTML color string, for example: 'red' or 'blue'. The <b>fontSize</b> is a number.</p> <p><b>Type:</b> object  <b>Default:</b> {color: 'black', fontName: &lt;global-font-name&gt;, fontSize: &lt;global-font-size&gt;}</p>
<code>vAxis.maxValue</code>	<p>Moves the max value of the vertical axis to the specified value; this property is ignored if this is set to a value smaller than the maximum y-value of the chart. <code>vAxis.viewWindow.max</code> overrides this property.</p> <p><b>Type:</b> number  <b>Default:</b> automatic</p>
<code>vAxis.minValue</code>	<p>Moves the min value of the vertical axis to the specified value; this property is ignored if this is set to a value greater than the minimum y-value of the chart. <code>vAxis.viewWindow.min</code> overrides this property.</p> <p><b>Type:</b> number  <b>Default:</b> null</p>
<code>vAxis.viewWindowMode</code>	<p>Specifies how to scale the vertical axis to render the values with the specified range. The following modes are supported:</p> <ul style="list-style-type: none"> <li>• 'pretty' - Scale the vertical values so that the maximum and minimum values are inside the top and bottom of the chart area. This will cause values outside the range to be truncated.</li> </ul>

	<p><code>vaxis.viewWindow.max</code> to be ignored.</p> <ul style="list-style-type: none"> <li>'maximized' - Scale the vertical values so that the maximum and bottom of the chart area. This will cause <code>vaxis.viewWindow.max</code> to be ignored.</li> <li>'explicit' - A deprecated option for specifying the top and bottom (Deprecated because it's redundant with <code>vaxis.viewWindow.max</code>). Data values outside these values are ignored. <code>vAxis.viewWindow</code> object describing the maximum and minimum values.</li> </ul> <p><b>Type:</b> string  <b>Default:</b> Equivalent to 'pretty', but <code>vaxis.viewWindow.min</code> and <code>vaxis.viewWindow.max</code> have precedence if used.</p>
<code>vAxis.viewWindow</code>	<p>Specifies the cropping range of the vertical axis.</p> <p><b>Type:</b> object  <b>Default:</b> null</p>
<code>vAxis.viewWindow.max</code>	<p>The maximum vertical data value to render.</p> <p>Ignored when <code>vAxis.viewWindowMode</code> is 'pretty' or 'maximized'.</p> <p><b>Type:</b> number  <b>Default:</b> auto</p>
<code>vAxis.viewWindow.min</code>	<p>The minimum horizontal data value to render.</p> <p>Ignored when <code>vAxis.viewWindowMode</code> is 'pretty' or 'maximized'.</p> <p><b>Type:</b> number  <b>Default:</b> auto</p>
<code>width</code>	<p>Width of the chart, in pixels.</p> <p><b>Type:</b> number  <b>Default:</b> width of the containing element</p>

## Methods

Method	
<code>draw(data, options)</code>	<p>Draws the chart. The chart accepts further method calls only after the <code>chart.draw</code> (<code>#Events</code>) event is fired. <a href="https://developers.google.com/chart/interactive/docs/reference#vis_events">Extended description</a> (<a href="https://developers.google.com/chart/interactive/docs/reference#vis_events">https://developers.google.com/chart/interactive/docs/reference#vis_events</a>)</p> <p><b>Return Type:</b> none</p>

<b>getAction(actionID)</b>	<p>Returns the tooltip action object with the requested <b>actionID</b>.</p> <p><b>Return Type:</b> object</p>
<b>getBoundingBox(id)</b>	<p>Returns an object containing the left, top, width, and height of chart element <b>id</b>. The format for <b>id</b> isn't yet documented (they're the return values of <a href="https://developers.google.com/chart/interactive/docs/events">events</a>), but here are some examples:</p> <pre>var cli = chart.getChartLayoutInterface();</pre> <p><b>Height of the chart area</b></p> <pre>cli.getBoundingBox('chartarea').height</pre> <p><b>Width of the third bar in the first series of a bar or column chart</b></p> <pre>cli.getBoundingBox('bar#0#2').width</pre> <p><b>Bounding box of the fifth wedge of a pie chart</b></p> <pre>cli.getBoundingBox('slice#4')</pre> <p><b>Bounding box of the chart data of a vertical (e.g., column) chart</b></p> <pre>cli.getBoundingBox('vAxis#0#gridline')</pre> <p><b>Bounding box of the chart data of a horizontal (e.g., bar) chart</b></p> <pre>cli.getBoundingBox('hAxis#0#gridline')</pre> <p>Values are relative to the container of the chart. Call this <i>after</i> the chart is rendered.</p> <p><b>Return Type:</b> object</p>
<b>getChartAreaBoundingBox()</b>	<p>Returns an object containing the left, top, width, and height of the chart area (i.e., excluding labels and legend):</p> <pre>var cli = chart.getChartLayoutInterface();</pre> <pre>cli.getChartAreaBoundingBox().left</pre> <pre>cli.getChartAreaBoundingBox().top</pre> <pre>cli.getChartAreaBoundingBox().height</pre> <pre>cli.getChartAreaBoundingBox().width</pre>

	<p>Values are relative to the container of the chart. Call this <i>after</i> the chart is drawn.</p> <p><b>Return Type:</b> object</p>
<b>getChartLayoutInterface()</b>	<p>Returns an object containing information about the onscreen placement of the chart and its elements.</p> <p>The following methods can be called on the returned object:</p> <ul style="list-style-type: none"> <li>• <b>getBoundingBox</b></li> <li>• <b>getChartAreaBoundingBox</b></li> <li>• <b>getHAxisValue</b></li> <li>• <b>getVAxisValue</b></li> <li>• <b>getXLocation</b></li> <li>• <b>getYLocation</b></li> </ul> <p>Call this <i>after</i> the chart is drawn.</p> <p><b>Return Type:</b> object</p>
<b>getHAxisValue(position, optional_axis_index)</b>	<p>Returns the logical horizontal value at <b>position</b>, which is an offset from the container's left edge. Can be negative.</p> <p>Example: <code>chart.getChartLayoutInterface().getHAxisValue(position)</code></p> <p>Call this <i>after</i> the chart is drawn.</p> <p><b>Return Type:</b> number</p>
<b>getImageURI()</b>	<p>Returns the chart serialized as an image URI.</p> <p>Call this <i>after</i> the chart is drawn.</p> <p>See <a href="https://developers.google.com/chart/interactive/docs/printing">Printing PNG Charts</a> (<a href="https://developers.google.com/chart/interactive/docs/printing">https://developers.google.com/chart/interactive/docs/printing</a>).</p> <p><b>Return Type:</b> string</p>
<b>getSelection()</b>	<p>Returns an array of the selected chart entities. Selectable entities are points, annotations, legend entries and categories. A point or annotation corresponds to a cell in the data table, a legend entry to a column (row index is null), and a category to a row (column index is null). For this chart, only one entity can be selected at a given moment. <a href="https://developers.google.com/chart/interactive/docs/reference#visualselection">Extended description</a> (<a href="https://developers.google.com/chart/interactive/docs/reference#visualselection">https://developers.google.com/chart/interactive/docs/reference#visualselection</a>).</p> <p><b>Return Type:</b> Array of selection elements</p>
<b>getVAxisValue(position,</b>	<p>Returns the logical vertical value at <b>position</b>, which is an offset from</p>

<code>optional_axis_index)</code>	<p>container's top edge. Can be negative.</p> <p>Example: <code>chart.getChartLayoutInterface().getVAxisValue</code></p> <p>Call this <i>after</i> the chart is drawn.</p> <p><b>Return Type:</b> number</p>
<code>getLocation(position, optional_axis_index)</code>	<p>Returns the screen x-coordinate of <b>position</b> relative to the chart's co</p> <p>Example: <code>chart.getChartLayoutInterface().getLocation(</code></p> <p>Call this <i>after</i> the chart is drawn.</p> <p><b>Return Type:</b> number</p>
<code>getYLocation(position, optional_axis_index)</code>	<p>Returns the screen y-coordinate of <b>position</b> relative to the chart's co</p> <p>Example: <code>chart.getChartLayoutInterface().getYLocation(</code></p> <p>Call this <i>after</i> the chart is drawn.</p> <p><b>Return Type:</b> number</p>
<code>removeAction(actionID)</code>	<p>Removes the tooltip action with the requested <b>actionID</b> from the cha</p> <p><b>Return Type:</b> none</p>
<code>setAction(action)</code>	<p>Sets a tooltip action to be executed when the user clicks on the action</p> <p>The <b>setAction</b> method takes an object as its action parameter. This should specify 3 properties: <b>id</b>— the ID of the action being set, <b>text</b> — should appear in the tooltip for the action, and <b>action</b> — the function to be run when a user clicks on the action text.</p> <p>Any and all tooltip actions should be set prior to calling the chart's <b>dra</b> method. <a href="https://developers.google.com/chart/interactive/docs/reference#vis">Extended description</a> (https://developers.google.com/chart/interactive/docs/reference#vis</p> <p><b>Return Type:</b> none</p>
<code>setSelection()</code>	<p>Selects the specified chart entities. Cancels any previous selection. Se entities are points, annotations, legend entries and categories. A point annotation corresponds to a cell in the data table, a legend entry to a c index is null), and a category to a row (column index is null). For this ch entity can be selected at a time. <a href="https://developers.google.com/chart/interactive/docs/reference#vis">Extended description</a> (https://developers.google.com/chart/interactive/docs/reference#vis</p> <p><b>Return Type:</b> none</p>
<code>clearChart()</code>	<p>Clears the chart, and releases all of its allocated resources.</p>

**Return Type:** none

## Events

For more information on how to use these events, see [Basic Interactivity](https://developers.google.com/chart/interactive/docs/basic_interactivity) ([https://developers.google.com/chart/interactive/docs/basic\\_interactivity](https://developers.google.com/chart/interactive/docs/basic_interactivity)), [Handling Events](https://developers.google.com/chart/interactive/docs/events) (<https://developers.google.com/chart/interactive/docs/events>), and [Firing Events](https://developers.google.com/chart/interactive/docs/dev/events) (<https://developers.google.com/chart/interactive/docs/dev/events>).

Name	
<b>animationfinish</b>	Fired when transition animation is complete.  <b>Properties:</b> none
<b>click</b>	Fired when the user clicks inside the chart. Can be used to identify when the title, data elements, legend entries, axes, gridlines, or labels are clicked.  <b>Properties:</b> targetID
<b>error</b>	Fired when an error occurs when attempting to render the chart.  <b>Properties:</b> id, message
<b>onmouseover</b>	Fired when the user mouses over a visual entity. Passes back the row and column indices of the corresponding data table element.  <b>Properties:</b> row, column
<b>onmouseout</b>	Fired when the user mouses away from a visual entity. Passes back the row and column indices of the corresponding data table element.  <b>Properties:</b> row, column
<b>ready</b>	The chart is ready for external method calls. If you want to interact with the chart, and call methods after you draw it, you should set up a listener for this event <i>before</i> you call the <b>draw</b> method, and call them only after the event was fired.  <b>Properties:</b> none
<b>select</b>	Fired when the user clicks a visual entity. To learn what has been selected, call <a href="#">getSelection()</a> (#Methods).  <b>Properties:</b> none

# Data Policy

---

All code and data are processed and rendered in the browser. No data is sent to any server.

---

*Except as otherwise noted, the content of this page is licensed under the [Creative Commons Attribution 3.0 License](http://creativecommons.org/licenses/by/3.0/) (<http://creativecommons.org/licenses/by/3.0/>), and code samples are licensed under the [Apache 2.0 License](http://www.apache.org/licenses/LICENSE-2.0) (<http://www.apache.org/licenses/LICENSE-2.0>). For details, see our [Site Policies](https://developers.google.com/terms/site-policies) (<https://developers.google.com/terms/site-policies>). Java is a registered trademark of Oracle and/or its affiliates.*

上次更新日期: 二月 23, 2017