

Framework

Swift Charts

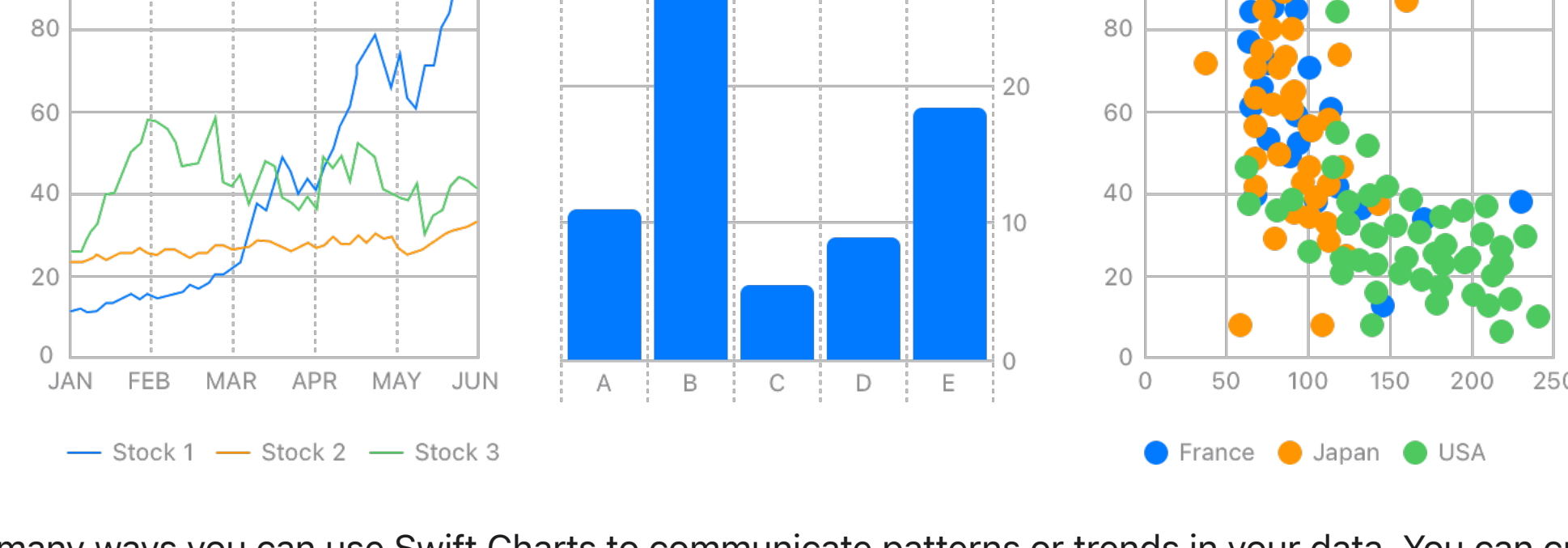
Construct and customize charts on every Apple platform.

iOS 16.0+ | iPadOS 16.0+ | Mac Catalyst 16.0+ | macOS 13.0+ | tvOS 16.0+ | visionOS 1.0+ | watchOS 9.0+



Overview

Swift Charts is a powerful and concise SwiftUI framework you can use to transform your data into informative visualizations. With Swift Charts, you can build effective and customizable charts with minimal code. This framework provides marks, scales, axes, and legends as building blocks that you can combine to develop a broad range of data-driven charts.



There are many ways you can use Swift Charts to communicate patterns or trends in your data. You can create a variety of charts including line charts, bar charts, and scatter plots as shown above. When you create a chart using this framework, it automatically generates scales and axes that fit your data.

Swift Charts supports localization and accessibility features. You can also override default behavior to customize your charts by using chart modifiers. For example, you can create a dynamic experience by adding animations to your charts.

Topics

Essentials

[Swift Charts updates](#)
Learn about important changes to Swift Charts.

Charts

[Creating a chart using Swift Charts](#)
Make a chart by combining chart building blocks in SwiftUI.

[Visualizing your app's data](#)
Build complex and interactive charts using Swift Charts.

[Chart](#)
A SwiftUI view that displays a chart.

[ChartContent](#)
A type that represents the content that you draw on a chart.

[ChartContentBuilder](#)
A result builder that you use to compose the contents of a chart.

[Plot](#)
A mechanism for grouping chart contents into a single entity.

3D charts

[Chart3D](#)
A SwiftUI view that displays interactive 3D charts and visualizations.

[Chart3DContent](#)
A type that represents the three-dimensional content that you draw on a chart.

[Chart3DContentBuilder](#)
A result builder that you use to compose the three-dimensional contents of a chart.

[SurfacePlot](#)
Chart content that represents a mathematical function of two variables using a 3D surface.

Marks

[AreaMark](#)
Chart content that represents data using the area of one or more regions.

[LineMark](#)
Chart content that represents data using a sequence of connected line segments.

[PointMark](#)
Chart content that represents data using points.

[RectangleMark](#)
Chart content that represents data using rectangles.

[RuleMark](#)
Chart content that represents data using a single horizontal or vertical rule.

[BarMark](#)
Chart content that represents data using bars.

[SectorMark](#)
A sector of a pie or donut chart, which shows how individual categories make up a meaningful total.

Vectorized plots

[Creating a data visualization dashboard with Swift Charts](#)
Visualize an entire data collection efficiently by instantiating a single vectorized plot in Swift Charts.

[AreaPlot](#)
Chart content that represents a function or a collection of data using the area of one or more regions.

[LinePlot](#)
Chart content that represents a function or a collection of data using a sequence of connected line segments.

[PointPlot](#)
Chart content that represents a collection of data using points.

[RectanglePlot](#)
Chart content that represents a collection of data using rectangles.

[RulePlot](#)
Chart content that represents a collection of data using a single horizontal or vertical rule.

[BarPlot](#)
Chart content that represents a collection of data using bars.

[SectorPlot](#)
Chart content that represents a collection of data using a sector of a pie or donut chart, which shows how individual categories make up a meaningful total.

[VectorizedChartContent](#)
A generic type that represents content conveyed via a chart.

Mark configuration

[MarkStackingMethod](#)
The ways in which you can stack marks in a chart.

[MarkDimension](#)
An individual dimension representing a mark's width or height.

[InterpolationMethod](#)
The ways in which line or area marks interpolate their data.

[BasicChartSymbolShape](#)
A basic chart symbol shape.

[ChartSymbolShape](#)
A type that can act as a shape for the marks that you add to a chart.

[AnyChartSymbolShape](#)
A type-erased plotting shape.

Labeled data

[PlottableValue](#)
Labeled data that you plot in a chart using marks.

[Plottable](#)
A type that can serve as data to plot in a chart.

Scales

[ScaleRange](#)
A type that you can use to configure the range of a chart.

[PositionScaleRange](#)
A type that configures the x-axis and y-axis values.

[PlotDimensionScaleRange](#)
A range that represents the plot area's width or height.

[ScaleDomain](#)
A type that you can use to configure the domain of a chart.

[AutomaticScaleDomain](#)
A domain that the chart infers from its data.

[ScaleType](#)
The ways you can scale the domain or range of a plot.

Axes

[Customizing axes in Swift Charts](#)
Improve the clarity of your chart by configuring the appearance of its axes.

[ChartAxisContent](#)
A view that represents a chart's axis.

[AxisContent](#)
A type that represents the elements you use to build a chart's axes.

[AxisMarks](#)
A group of visual marks that a chart draws to indicate the composition of a chart's axes.

[AnyAxisContent](#)
A type-erased element of a chart's axis.

[AxisContentBuilder](#)
A result builder that constructs axis content.

Axis marks

[AxisMark](#)
A type that serves as the basic building block for the elements of an axis.

[AxisTick](#)
A mark that a chart draws on an axis to indicate a reference point along that axis.

[AxisGridLine](#)
A line that a chart draws across its plot area to indicate a reference point along a particular axis.

[AxisValueLabel](#)
A label that describes the value for an axis mark.

[AxisValue](#)
A value for an axis mark.

[AnyAxisMark](#)
A type-erased axis mark.

[AxisMarkBuilder](#)
A result builder that constructs axis marks and overrides default marks.

Annotations

[AnnotationContext](#)
Information about an item that you add an annotation to.

[AnnotationPosition](#)
The position of an annotation.

[AnnotationOverflowResolution](#)

Data bins

[NumberBins](#)
A collection of bins for a chart that plots data against numbers.

[DateBins](#)
A collection of bins for a chart that plots data against dates.

[ChartBinRange](#)
The range of data that a single bin of a chart represents.

Chart management

[ChartPlotContent](#)
A view that represents a chart's plot area.

[ChartProxy](#)
A proxy that you use to access the scales and plot area of a chart.

Scrolling

[ChartScrollTargetBehavior](#)
A type that configures the scroll behavior of charts.

[ChartScrollTargetBehaviorContext](#)
Contextual information that you can use to determine how to best adjust how charts scroll.

Structures

[Chart3DRenderingStyle](#)