

[AppKit](#) / [Views and Controls](#) / Collection View

API Collection

Collection View

Display one or more subviews in a highly configurable arrangement.

Topics

View

`class NSCollectionView`

An ordered collection of data items displayed in a customizable layout.

`protocol NSCollectionViewSectionHeaderView`

A protocol that defines a button to control the collapse of a collection view's section.

Components

`class NSCollectionLayoutItem`

The most basic component of a collection view's layout.

`class NSCollectionLayoutGroup`

A container for a set of items that lays out the items along a path.

`class NSCollectionLayoutSection`

A container that combines a set of groups into distinct visual groupings.

Size and spacing

`class NSCollectionLayoutDimension`

An individual dimension representing an item's width or height in a collection view.

```
class NSCollectionLayoutSize
```

The width and the height of an item in a collection view.

```
class NSCollectionLayoutSpacing
```

An object that defines the space between or around items in a collection view.

```
class NSCollectionLayoutEdgeSpacing
```

An object that defines the space around the edges of items in a collection view.

```
protocol NSCollectionLayoutContainer
```

A protocol used to provide information about the size and content insets of a layout's container.

Data

```
protocol NSCollectionViewDataSource
```

A set of methods that a data source object implements to provide the information and view objects that a collection view requires to present content.

```
protocol NSCollectionViewDelegate
```

A set of methods that you use to manage the behavior of a collection view.

```
class NSCollectionViewDiffableDataSource
```

The object you use to manage data and provide items for a collection view.

```
struct NSDiffableDataSourceSnapshot
```

A representation of the state of the data in a view at a specific point in time.

Items

```
class NSCollectionViewItem
```

The visual representation for a single data element in a collection view.

```
protocol NSCollectionViewElement
```

A set of methods that you use to manage the content in a collection view.

Layouts

```
{} Implementing modern collection views
```

Bring compositional layouts to your app and simplify updating your user interface with diffable data sources.

class NSCollectionViewFlowLayout

A layout that organizes items into a flexible and configurable arrangement.

protocol NSCollectionViewDelegateFlowLayout

A set of methods that a delegate implements to provide layout information to a flow layout object in a collection view.

class NSCollectionViewGridLayout

A layout that displays a single section of items in a row and column grid.

class NSCollectionViewTransitionLayout

An object that implements custom behaviors when changing from one layout to another in a collection view.

class NSCollectionViewLayoutAttributes

An object that contains layout-related attributes for an element in a collection view.

class NSCollectionViewLayout

An abstract base class that you subclass and use to generate layout information for a collection view.

class NSCollectionViewCompositionalLayout

A layout object that lets you combine items in highly adaptive and flexible visual arrangements.

class NSCollectionViewCompositionalLayoutConfiguration

An object that defines scroll direction, section spacing, and headers or footers for the layout.

typealias NSCollectionViewCompositionalLayoutSectionProvider

A closure that creates and returns each of the layout's sections.

enum NSCollectionLayoutSectionOrthogonalScrollingBehavior

The scrolling behavior of the layout's sections in relation to the main layout axis.

Appearance

class NSCollectionLayoutBoundarySupplementaryItem

An object used to add headers or footers to a collection view.

```
class NSCollectionLayoutSupplementaryItem
```

An object used to add an extra visual decoration to an item in a collection view.

```
class NSCollectionLayoutDecorationItem
```

An object used to add a background to a section of a collection view.

```
class NSCollectionLayoutAnchor
```

An object that defines how to attach a supplementary item to an item in a collection view.

Advanced layouts

```
class NSCollectionLayoutGroupCustomItem
```

An item used in a group with a custom layout arrangement.

```
typealias NSCollectionLayoutGroupCustomItemProvider
```

A closure that creates and returns each of the custom group's items.

Layout updates

```
protocol NSCollectionLayoutVisibleItem
```

An item that's currently visible within the bounds of a section.

```
typealias NSCollectionLayoutSectionVisibleItemsInvalidationHandler
```

A closure called before each layout cycle to allow modification of items in a section immediately before they're displayed.

Configuration

```
protocol NSCollectionLayoutEnvironment
```

A protocol used to provide information about the layout's container and environment traits, such as size classes and display scale factor.

Updates

```
class NSCollectionViewUpdateItem
```

A description of a single change to make to an item in a collection view.

```
class NSCollectionViewLayoutInvalidationContext
```

An object that identifies the portions of your layout that need to be updated.

```
class NSCollectionViewFlowLayoutInvalidationContext
```

An object that identifies the portions of a flow layout object that need to be updated.

See Also

Content views

☰ Browser View

Provide a column-based interface for viewing and navigating hierarchical information.

☰ Outline View

Display a list-based interface for hierarchical data, where each level of hierarchy is indented from the previous one.

☰ Table View

Display custom data in rows and columns.

```
class NSTextView
```

A view that draws text and handles user interactions with that text.