API Collection

Numbers, Data, and Basic Values

Work with primitive values and other fundamental types used throughout Cocoa.

Topics

Numbers

`@frozen struct Int`

A signed integer value type.

`@frozen struct Double`

A double-precision, floating-point value type.

`struct Decimal`

A structure representing a base-10 number.

`class NumberFormatter`

A formatter that converts between numeric values and their textual representations.

Binary Data

`struct Data`

A byte buffer in memory.

`protocol DataProtocol`

A protocol that provides consistent data access to the bytes underlying contiguous and noncontiguous data buffers.

`protocol MutableDataProtocol`

A protocol that provides consistent data access to the bytes underlying contiguous and noncontiguous mutable data buffers.

`protocol ContiguousBytes`

A protocol that declares the type offers direct access to the underlying raw bytes in a contiguous manner.

URLs

`struct URL`

A value that identifies the location of a resource, such as an item on a remote server or the path to a local file.

`struct URLComponents`

A structure that parses URLs into and constructs URLs from their constituent parts.

`struct URLQueryItem`

A single name-value pair from the query portion of a URL.

Unique Identifiers

`struct UUID`

A universally unique value to identify types, interfaces, and other items.

Geometry

`@frozen struct CGFloat`

The basic type for floating-point scalar values in Core Graphics and related frameworks.

`typealias NSPoint`

A point in a Cartesian coordinate system.

`typealias NSSize`

A two-dimensional size.

`typealias NSRect`

A rectangle.

`struct AffineTransform`

A graphics coordinate transformation.

`struct` NSEdgeInsets

A description of the distance between the edges of two rectangles.

Ranges

`typealias` NSRange

A structure used to describe a portion of a series, such as characters in a string or objects in an array.

See Also

Fundamentals

- Strings and Text

  Create and process strings of Unicode characters, use regular expressions to find patterns, and perform natural language analysis of text.

- Collections

  Use arrays, dictionaries, sets, and specialized collections to store and iterate groups of objects or values.

- Dates and Times

  Compare dates and times, and perform calendar and time zone calculations.

- Units and Measurement

  Label numeric quantities with physical dimensions to allow locale-aware formatting and conversion between related units.

- Data Formatting

  Convert numbers, dates, measurements, and other values to and from locale-aware string representations.

- Filters and Sorting

  Use predicates, expressions, and sort descriptors to examine elements in collections and other services.