API Collection

Object Runtime

Get low-level support for basic Objective-C features, Cocoa design patterns, and Swift integration.

Topics

Object Basics

`class` NSObject

The root class of most Objective-C class hierarchies, from which subclasses inherit a basic interface to the runtime system and the ability to behave as Objective-C objects.

`protocol` NSObjectProtocol

The group of methods that are fundamental to all Objective-C objects.

NSKeyValueCoding

A mechanism by which you can access the properties of an object indirectly by name or key.

Copying

`protocol` NSCopying

A protocol that objects adopt to provide functional copies of themselves.

`protocol` NSMutableCopying

A protocol that mutable objects adopt to provide functional copies of themselves.

Value Wrappers and Transformations

`class` NSNumber

An object wrapper for primitive scalar numeric values.

`class` NSValue

A simple container for a single C or Objective-C data item.

`class` ValueTransformer

An abstract class used to transform values from one representation to another.

Swift Support

SupportedLanguage(swift)

`protocol` ReferenceConvertible

A decoration applied to types that are backed by a Foundation reference type.

Classes Bridged to Swift Standard Library Value Types

Use bridged reference types when you need reference semantics or Foundation-specific behavior.

Remote Objects

`class` NSProxy

An abstract superclass defining an API for objects that act as stand-ins for other objects or for objects that don't exist yet.

Memory Management

Memory Management Functions

Perform low-level memory management tasks.

Objective-C Runtime

Objective-C Runtime Utilities

Interact with the Objective-C runtime.

Versions and API Availability

Foundation Framework Version Numbers

Recognize the constants for comparing the current running version of Foundation against known OS version numbers.

Legacy

SupportedLanguage(swift)

- Distributed Objects Support

  Enable communication among objects in different processes, both locally and on remote systems.

- Objective-C Garbage Collection

  Interface with the legacy garbage collection system.

See Also

Low-Level Utilities

- XPC

  Manage secure interprocess communication.

- Processes and Threads

  Manage your app's interaction with the host operating system and other processes, and implement low-level concurrency features.

- Streams, Sockets, and Ports

  Use low-level Unix features to manage input and output among files, processes, and the network.