

Class

NSRunLoop

The `NSRunLoop` class declares the programmatic interface to objects that manage input sources. An `NSRunLoop` object processes input for sources such as mouse and keyboard events from the window system, `NSPort` objects, and `NSConnection` objects. An `NSRunLoop` object also processes `NSTimer` events.

Language

Swift | Objective-C

SDKs

iOS 2.0+

macOS 10.0+

tvOS 9.0+

watchOS 2.0+

On This Page

Overview ☺

Symbols ☺

Relationships ☺

Overview

Your application cannot either create or explicitly manage `NSRunLoop` objects. Each `NSThread` object—including the application’s main thread—has an `NSRunLoop` object automatically created for it as needed. If you need to access the current thread’s run loop, you do so with the class method `currentRunLoop`.

Note that from the perspective of `NSRunLoop`, `NSTimer` objects are not "input"—they are a special type, and one of the things that means is that they do not cause the run loop to return when they fire.

Warning

The `NSRunLoop` class is generally not considered to be thread-safe and its methods should only be called within the context of the current thread. You should never try to call the methods of an `NSRunLoop` object running in a different thread, as doing so might cause unexpected results.

Symbols

Accessing Run Loops and Modes

<code>currentRunLoop</code>	Returns the run loop for the current thread.
<code>currentMode</code>	The receiver's current input mode.
<code>– limitDateForMode:</code>	Performs one pass through the run loop in the specified mode and returns the date at which the next timer is scheduled to fire.
<code>– ...</code>	

API Reference

- `getCFRunLoop`
Returns the receiver's underlying CFRunLoop object.

Managing Timers

- `addTimer:forMode:`
Registers a given timer with a given input mode.

Managing Ports

- `addPort:forMode:`
Adds a port as an input source to the specified mode of the run loop.
- `removePort:forMode:`
Removes a port from the specified input mode of the run loop.

Configuring as Server Process

- ~~`configureAsServer`~~
Deprecated. Does nothing.

Deprecated

Running a Loop

- `run`
Puts the receiver into a permanent loop, during which time it processes data from all attached input sources.
- `runMode:beforeDate:`
Runs the loop once, blocking for input in the specified mode until a given date.
- `runUntilDate:`
Runs the loop until the specified date, during which time it processes data from all attached input sources.
- `acceptInputForMode:beforeDate:`
Runs the loop once or until the specified date, accepting input only for the specified mode.

Scheduling and Canceling Messages

- `performSelector:target:argument:order:modes:`
Schedules the sending of a message on the receiver.
- `cancelPerformSelector:target:argument:`
Cancels the sending of a previously scheduled message.
- `cancelPerformSelectorsWithTarget:`
Cancels all outstanding ordered performs scheduled with a given target.

Constants

- Run Loop Modes
NSRunLoop defines the following run loop mode.

Instance Methods

- `performBlock:`

– `performInModes:block:`

Relationships

Inherits From `NSObject`