

Dependency Manager

SEARCH*



* Type here to search by name, author and summary.

WHAT IS COCOAPODS

CocoaPods is a dependency manager for Swift and Objective-C Cocoa projects. It has over 62 thousand libraries and is used in over 3 million apps. CocoaPods can help you scale your projects elegantly.

INSTALL

GET STARTED

CREATE A POD

CocoaPods is built with Ruby and is installable with the default Ruby available on macOS. We recommend you use the default ruby.

COCOPODS

CocoaPods is a dependency manager for Swift and Objective-C Cocoa projects. It has over 62 thousand libraries and is used in over 3 million apps. CocoaPods can help you scale your projects elegantly.

SEARCH*



* Type here to search by name, version, author, keywords, summary, and dependencies.

[https://cocoapods.org/pods/\[libraryName\]](https://cocoapods.org/pods/[libraryName])

COCOAPODS Install

```
$ sudo gem install cocoapods
```

Edit Podfile

```
$ pod init
```

```
$ vi Podfile
```

```
target 'CocoaPodsExample' do
```

```
  use_frameworks!
```

```
  pod 'SnapKit' // 사용할 라이브러리 작성
```

```
end
```

Install Library

```
$ pod repo update
```

```
$ pod install
```

// pod install을 이전에 한 번도 수행한 적이 없을 경우 약 5 ~ 15분 소요

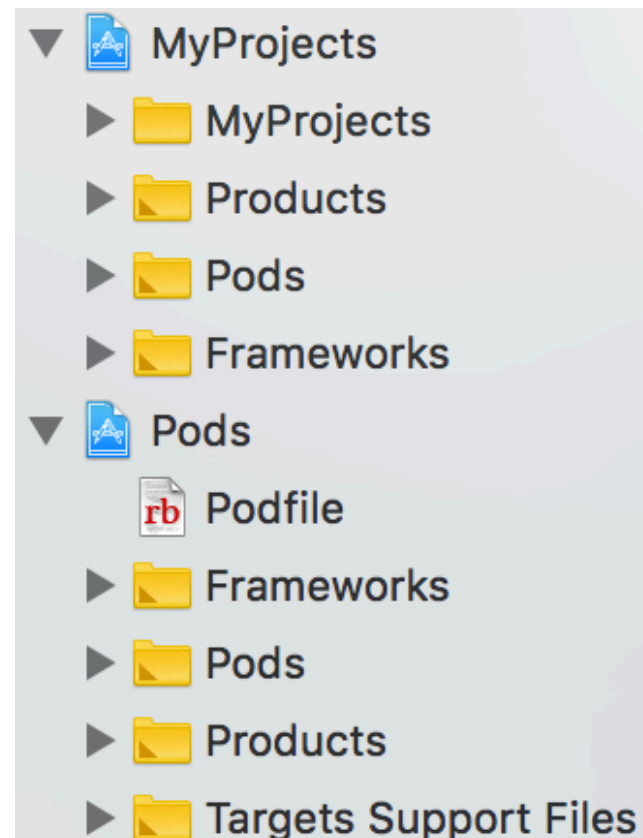
```
$ open [ProjectName].xcworkspace
```



MyProjects.xcode
proj




MyProjects.xcwor
kspace



COCOPODS

<https://github.com/CocoaPods/CocoaPods>

 CocoaPods / CocoaPods

 Used by ▾

8.7k

 Watch ▾

544

★ Star

12.3k

 Fork

2.2k

↔ Code

! Issues 178

🔗 Pull requests 21

▶ Actions

📁 Projects 0

📖 Wiki

🛡 Security

📊 Insights

The Cocoa Dependency Manager. <https://cocoapods.org/>

ios

dependency-manager

mac

objective-c

swift

cocoapods

package-manager

🕒 8,741 commits

🌿 54 branches

📦 0 packages

🏷 201 releases

👤 272 contributors

📄 View license

Branch: master ▾

New pull request

Create new file

Upload files

Find file

Clone or download ▾

Carthage

<https://github.com/Carthage/Carthage>

Carthage / Carthage

Watch

380

Star

13.5k

Fork

1.4k

Code

Issues148

Pull requests15

Actions

Projects0

Security

Insights

A simple, decentralized dependency manager for Cocoa

cocoa

carthage

swift

package-manager

4,207 commits

12 branches

0 packages

72 releases

164 contributors

View license

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download



Carthage

license MIT

release v0.23.0

Carthage is intended to be the simplest way to add frameworks to your Cocoa application.

The basic [workflow](#) looks something like this:

1. Create a [Cartfile](#) that lists the frameworks you'd like to use in your project.
2. [Run Carthage](#), which fetches and builds each framework you've listed.
3. Drag the built `.framework` binaries into your application's Xcode project.

Carthage builds your dependencies and provides you with binary frameworks, but you retain full control over your project structure and setup. Carthage does not automatically modify your project files or your build settings.

Install Carthage

```
$ brew update && brew install carthage
```

※ brew 가 없을 경우 <https://brew.sh> 에서 설치

```
$ brew upgrade carthage // 이미 설치되어 있을 때 업그레이드
```

Edit Cartfile

```
$ vi Cartfile
```

```
// 사용할 라이브러리 작성
```

```
github "Alamofire/Alamofire"
```

```
github "onevc/Kingfisher"
```

\$ carthage update

- 전체 업데이트

\$ carthage update --platform iOS



- iOS Platform 한정

\$ carthage update Alamofire

- 여러 가지 라이브러리 중 일부만 지정하여 업데이트 할 때

Link Binary With Libraries

▼ Link Binary With Libraries (2 items)

Name	Status
 Alamofire.framework	Required ⇅
 Kingfisher.framework	Required ⇅

+ —

Drag to reorder frameworks

The screenshot shows the Xcode interface with the 'Build Phases' tab selected. On the left, the 'PROJECT' and 'TARGETS' sections are visible, with 'MyProjects' listed under both. A context menu is open over the 'MyProjects' target, displaying a list of options to add new build phases. The options are: 'New Copy Files Phase', 'New Run Script Phase' (which is highlighted in blue), 'New Headers Phase', 'New Copy Bundle Resources Phase', 'New Compile Sources Phase', 'New Link Binary With Libraries Phase', and 'New Build Carbon Resources Phase'. Below this menu, the existing build phases for the target are visible: 'Copy Bundle Resources (3 items)' and 'Run Script'. Each phase has a plus icon on the left and a minus icon on the right.

Run Script

▼ Run Script



Shell

```
1 /usr/local/bin/carthage copy-frameworks
2
```

☒ Show environment variables in build log

☐ Run script only when installing

Input Files

+ -

Build Phases - Run Script - Shell

```
> /usr/local/bin/carthage copy-frameworks
```

Build Phases - Run Script - Input Files

```
> $(SRCROOT)/Carthage/Build/iOS/[Name].framework
```

OpenSources

[Dependency Manager]

CocoaPods	https://cocoapods.org
Carthage	https://github.com/Carthage/Carthage
Swift Package Manager	https://github.com/apple/swift-package-manager

[OpenSource Collection]

Awesome-Swift	https://github.com/matteocrippa/awesome-swift
Awesome-Swift	https://github.com/Wolg/awesome-swift
Awesome-iOS	https://github.com/vsouza/awesome-ios
Awesome-iOS-UI	https://github.com/cjwirth/awesome-ios-ui
Fantastic-iOS-Animation	https://github.com/onmyway133/fantastic-ios-animation
Awesome-iOS	http://awesome-ios.readthedocs.io/en/latest/#awesome-ios
Open Source iOS Apps	https://github.com/dkhamsing/open-source-ios-apps
Swift UI Libraries	https://goo.gl/WdvyLk
Cocoa Controls	https://www.cocoacontrols.com