

极客学院
jikexueyuan.com

使用 CocoaPods 管理 iOS 库

使用 CocoaPods 管理 iOS 库 — 课程概要

- iOS 库的介绍
- CocoaPods 基础知识
- podspec 和私有 pods

使用 CocoaPods 管理 iOS 库

iOS 库的介绍

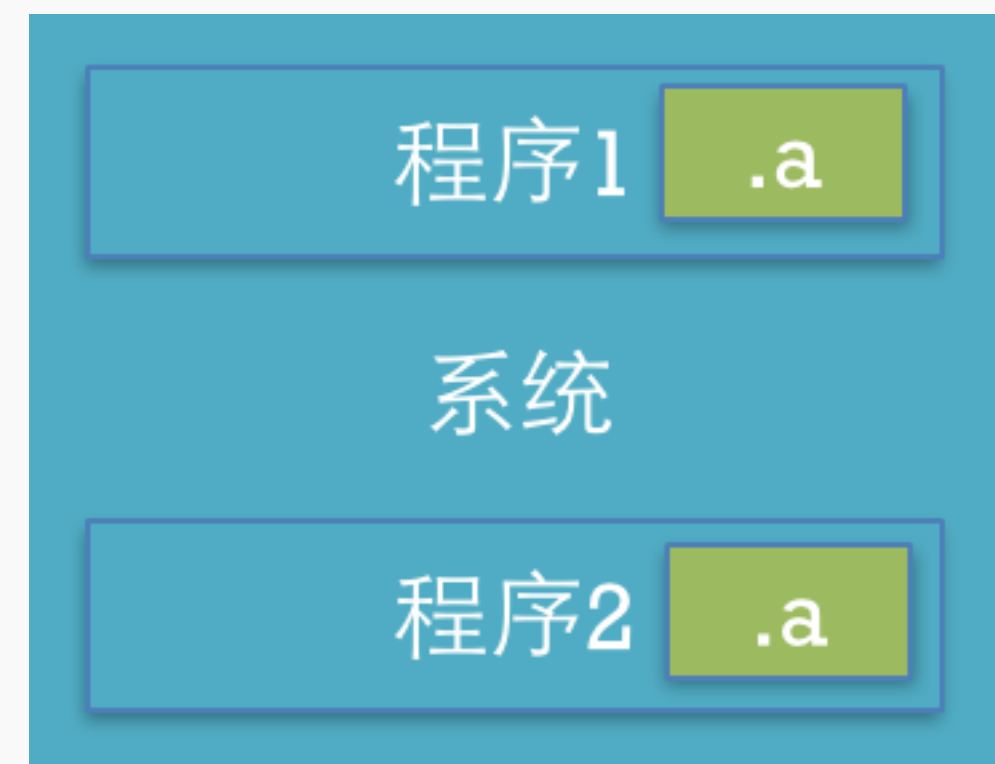
■ iOS 库的介绍 — 课时知识点

- 库的分类
- 制作库
- 使用库

- 动态库和静态库
 1. 动态库在运行时实时部署，静态库必须在编译时连接到代码
 2. 动态库加载到内存后可以重复使用,静态库每次 import 都引入一次
 3. 自定义的动态库只能部署在 iOS8 以上系统

■ iOS 库的介绍 — 库的分类

- .a
- .dylib
- .framework



iOS 库的介绍 — 库的分类

- framework 可以包含资源
- framework 分为两种：dynamic 和 static
- use_frameworks! 生成 dynamic framework

iOS 库的介绍 — 制作库

Build Active Architecture Only (Debug,Release) – 只编译当前设备体系结构的包

iOS 体系结构

armv7: iPhone 4/4s

armv7s: iPhone 5 和 iPhone 5C

arm64: iPhone 5S+

x86-64、i386: 模拟器

iOS 库的介绍 — 制作库

`lipo -info` – 查看库支持的体系结构

`lipo -create -output` - 将模拟器和真机的库合成一个库

iOS 库的介绍 — 使用库

- 手动导入

Linked Frameworks and Libraries、Embedded Binaries

- 使用 pod

use_frameworks!

利用 dynamic 库实时部署

```
let libPath = "\(NSHomeDirectory())/Documents/MyFramework.framework/MyFramework"
if let bundle = NSBundle(path: libPath) {
    do {
        try bundle.loadAndReturnError()
        if let myFramework = NSClassFromString("MyFramework") as?
MyFramework.Type{
            myFramework.init().whereIsFrom()
        }
    } catch {
        print("动态库加载失败")
    }
}
```

使用 CocoaPods 管理 iOS 库

CocoaPods 基础知识

CocoaPods 基础知识 — 课时知识点

- CocoaPods 作用
- 安装
- 使用镜像索引
- 使用步骤
- Podfile
- 查找第三方库
- Podfile.lock
- pod install 和 pod update

CocoaPods 基础知识 — CocoaPods 作用

- 自动设置和更新库以及库依赖
- 下载并链接源代码来生成 workspace 设置好相应的系统依赖和编译参数

CocoaPods 基础知识 — 安装

```
gem sources --remove https://rubygems.org/
```

```
gem sources -a https://ruby.taobao.org/
```

```
gem sources -l
```

```
sudo gem install cocoapods
```

```
pod setup
```

CocoaPods 基础知识 — 安使用镜像索引

```
pod repo remove master
```

```
pod repo add master https://gitcafe.com/akuandev/Specs.git
```

```
pod repo update
```


CocoaPods 基础知识 — 使用步骤

- 创建或打开 Xcode 项目
- 命令行 cd 到项目目录
- 使用 pod init 创建 Podfile
- 修改 Podfile
- pod install
- 打开 pod 创建的 xcworkspace

CocoaPods 基础知识 — Podfile

```
platform :ios '9.0'  
use_frameworks!  
inhibit_all_warnings!  
target 'PodTest' do  
  pod 'AFNetworking', '~> 3.0'  
  pod 'FBSDKCoreKit', '~> 4.9'  
end
```

CocoaPods 基础知识 — 查找第三方库

pod search JSONKit

CocoaPods 基础知识 — Podfile.lock

- 第一次 pod install 时生成记录每个 Pod 版本
- Podfile.lock 锁定当前各依赖库的版本之后 pod install 不会更改版本pod update 才会改版本
- 多人协作时防止第三方库升级时造成版本不一致

CocoaPods 基础知识 — **pod install 和 pod update**

- `pod install` : 安装 Podfile.lock 中锁定的版本的 pods
- `pod update`: 会更新到最新版本或 podfile 中指定版本
- 使用 `pod update podName` 来更新指定 pod 的版本

■ 使用 CocoaPods 管理 iOS 库

podspec 和私有 pods

podspec 和私有 pods — 课时知识点

- podspec
- source 参数说明
- 使用私有 pods
- 不更新 podspec

podspec 和私有 pods — **podspec**

```
Pod::Spec.new do |s|
  s.name      = "MyFramework"
  s.version   = "0.0.1"
  s.summary   = '测试库程序'
  s.source    = { :git => 'https://github.com/hefeijinbo/MyFramework.git' }
  s.platform  = :ios
  s.source_files = 'MyFramework/**/*.*swift'
  s.resources  = 'MyFramework/**/*.{png,xib,storyboard}'
end
```


podspec 和私有 pods — **source** 参数说明

- :git 使用默认 master 分支
- :branch 指定分支
- :tag 指定 tag
- :commit 指定 commit
- :path 本地的资源库

具体示例看底部注释

■ podspec 和私有 pods — 使用私有 pods

```
pod 'MyCommon', :podspec =>  
  'https://yuantiku.com/common/myCommon.podspec'
```

podspec 和私有 pods — **不更新 podspec**

```
pod install --no-repo-update
```

```
pod update --no-repo-update
```

使用 CocoaPods 管理 iOS 库 — 课程总结

本套课程中我们学习了 CocoaPods 相关知识。你应当掌握了以下知识：

- 掌握 iOS 库相关知识
- CocoaPods 基础知识
- podspec 和私有 pods

如果想继续提高，你可以继续在极客学院学习后续相关课程。

极客学院

jikexueyuan.com

中国最大的IT职业在线教育平台

