

[返回目录:全网各大厂 iOS 面试题-题集大全](#)

如何优化 App 的包大小?

直接上建议:

- 资源优化
 - 删除无用图片
 - 使用 [LSUnusedResources](#) 查找无用图片。注意 `[UIImage imageNamed:[NSString stringWithFormat:@"icon_%d.png",index]]`; 这种使用图片的方式, 可能会被误删。
 - 删除重复资源
 - 压缩图片资源
 - 使用 [ImageOptim](#) 无损压缩图片。
 - 使用 [TinyPNG](#) 有损压缩图片。使用的时候直接执行 `tinypng *.png -k token` 脚本即可。
 - 其他技巧 (选用)
 - 用 `LaunchScreen.storyboard` 替换启动图片。
 - 本地大图片都使用 `webp`。
 - 资源按需加载, 非必要资源都等到使用时再从服务端拉取。
- 编译选项优化
 - Optimization Level 在 release 状态设置为 Fastest/Smallest。
 - Strip Debug Symbols During Copy 在 release 状态设置为 YES。
 - Strip Linked Product 在 release 状态设为 YES。
 - Make String Read-Only 在 release 状态设为 YES。
 - Dead Code Stripping 在 release 状态设为 YES。
 - Deployment PostProcessing 在 release 状态设为 YES。
 - Symbols hidden by default 在 release 状态设为 YES。
- 可执行文件优化
 - 使用 [LinkMap](#) 分析库的使用情况
 - 三方库优化
 - 删除不使用的三方库。
 - 功能用的少但是体积大的三方库可以考虑自己重写。
 - 合并功能重复的三方库。
 - 代码分析
 - 用 `AppCode` 进行代码扫描。

- 去掉无用的类及文件。
- 清理 import。
- 去掉空方法。
- 去掉无用的 log。
- 去掉无用的变量。
- 其他技巧（选用）
 - 将业务打包成动态库。如果动态库的加载时机不控制好，会影响 App 的启动速度，权衡使用。
 - 动态化。将一部分 Native 界面用 RN/Weex 重写。
 - 去除 Swift 代码，Swift 的标准库是打包在安装包里的，一般都有 10M+。然后苹果官方说等到 Swift Runtime 稳定之后会合并到 iOS 系统里，那时候使用 Swift 就不会显著增加包大小了。
 - 在 target -> Build Settings -> Other Link Flags 里添加如下指令，会把 TEXT 字段的部分内容转移到 RODATA 字段，避免苹果对 TEXT 字段的审核限制。当然其实跟安装包瘦身好像没有什么关系，所以除非快不行了否则不建议操作。-Wl,-rename_section,__TEXT,__cstring,__RODATA,__cstring -Wl,-rename_section,__TEXT,__gcc_except_tab,__RODATA,__gcc_except_tab -Wl,-rename_section,__TEXT,__const,__RODATA,__const -Wl,-rename_section,__TEXT,__objc_methname,__RODATA,__objc_methname -Wl,-rename_section,__TEXT,__objc_classname,__RODATA,__objc_classname -Wl,-rename_section,__TEXT,__objc_meth_type,__RODATA,__objc_meth_type
- 苹果官方的策略
 - App Thinning
 - 使用 xcasset 管理图片。
 - 开启 BitCode

[返回目录:全网各大厂 iOS 面试题-题集大全](#)