

一、flutter工程类型

二、flutter application

1、目录结构

2、目录介绍

(1) lib

(2) android

(3) ios

(4) test

(5) pubspec.yaml文件

3、总结

三、Flutter plugin

1、目录结构

2、目录介绍

(1) android

(2) ios

(3) example

(4) lib

(5) pubspec.yaml文件

四、Flutter package

1、目录结构

2、目录介绍

(1) lib

(2) test

(3) pubspec.yaml

五、Flutter module

1、目录结构

2、目录介绍

(1) .android

(2) .ios

(3) lib

(4) test目录

(5) pubspec.yaml

一、flutter工程类型

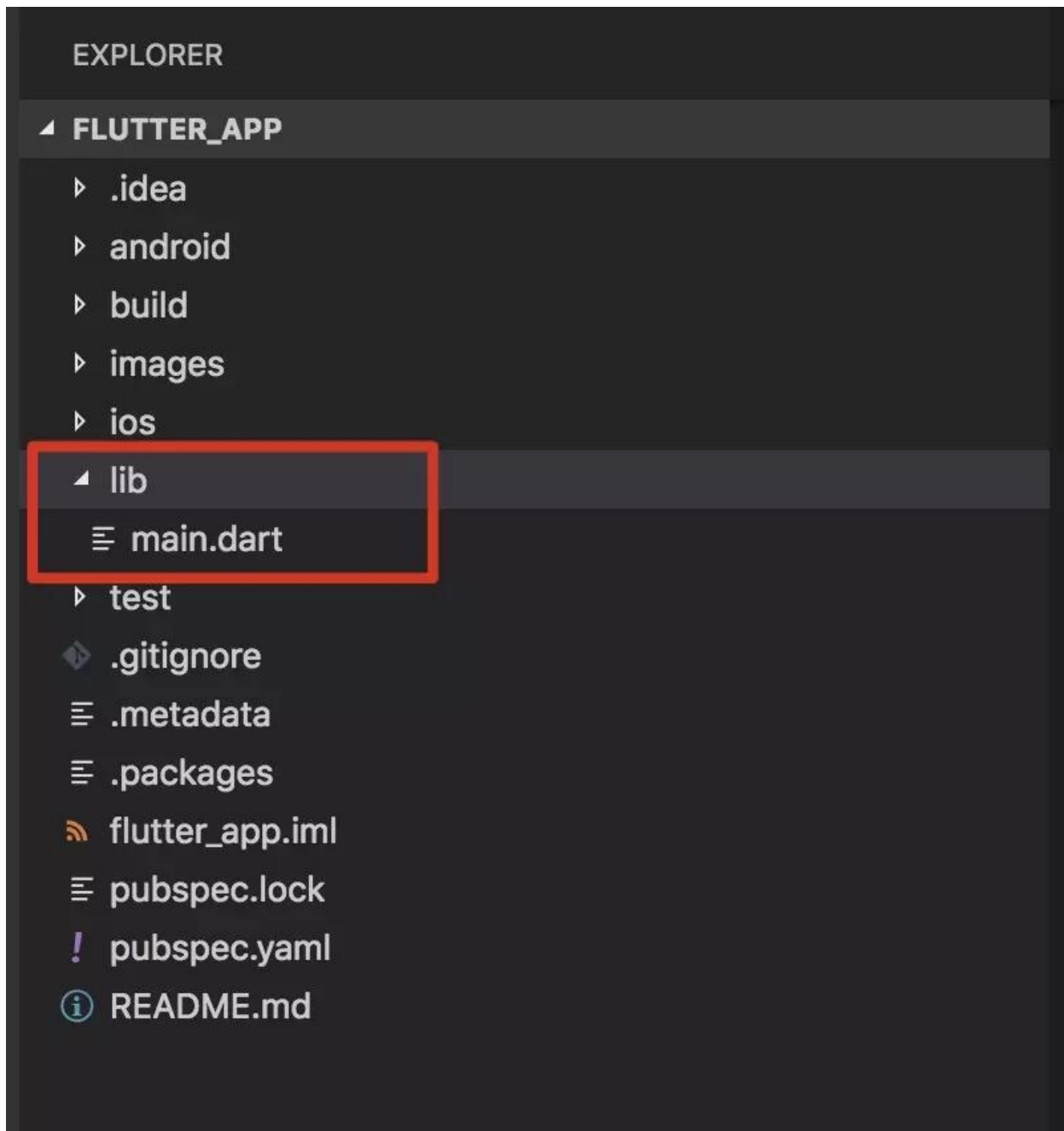
flutter工程主要分为四种：

- flutter application
- flutter plugin
- flutter package
- flutter module

二、flutter application

flutter application是一个纯flutter app，可以直接运行在Android或者ios上给用户使用（区别于后面的flutter module）

1、目录结构

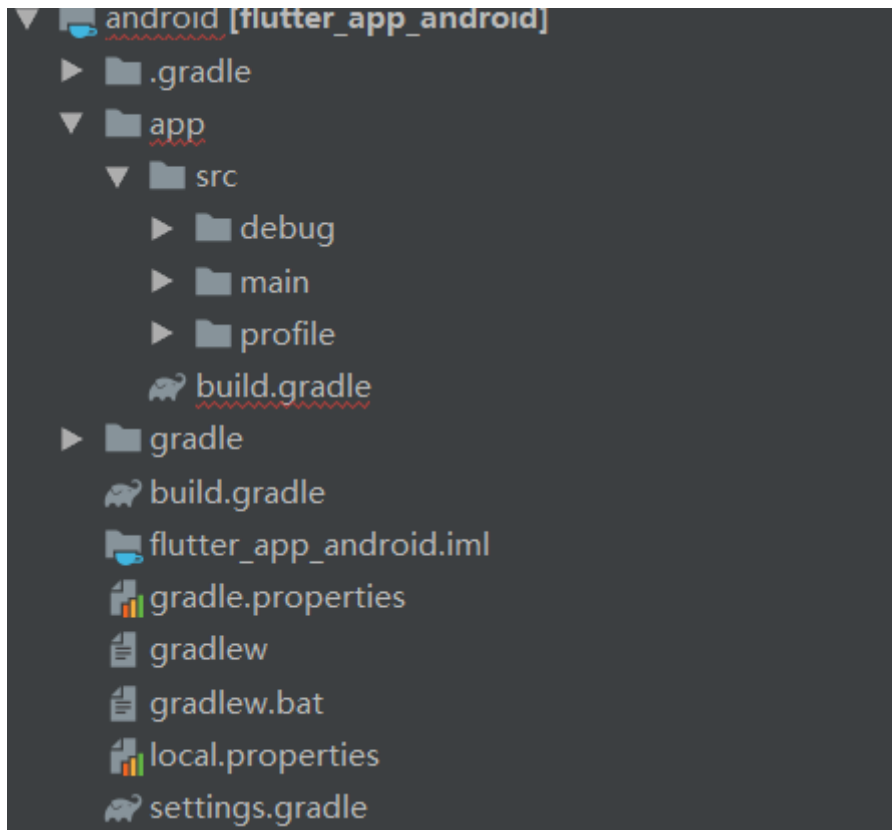


2、目录介绍

(1) lib

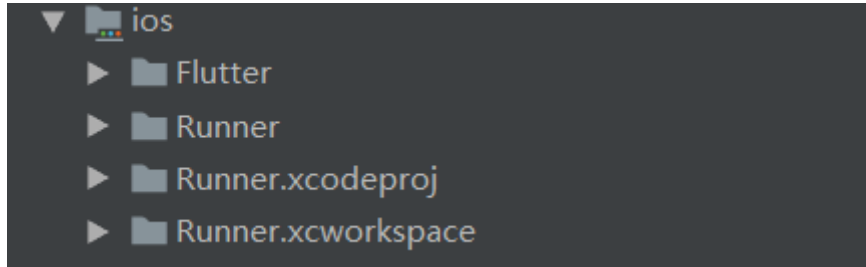
lib文件夹中为flutter代码，我们编写的flutter代码必须放在该目录下，这里需要注意，入口文件必须为main.dart，否则会报错

(2) android



该目录下是一个完整的android app工程代码，它是lib目录中flutter代码在Android平台上的壳子，该目录下的代码会被打包进flutter的Android安装包中。

(3) ios



该目录下是一个完整的ios app工程代码，它是lib目录中flutter代码在ios平台运行的壳子，该目录下代码会被打包进flutter的ios安装包中。

(4) test

该目录下是flutter的测试代码，使用dart来编写

(5) pubspec.yaml文件

```

# In iOS, build-name is used as CFBundleShortVersionString while build-number used as CFBundleVersion.
# Read more about iOS versioning at
# https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/CoreFoundationKeys.html
version: 1.0.0+1

environment:
  sdk: ">=2.1.0 <3.0.0"

dependencies:
  flutter:
    sdk: flutter

  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^0.1.2

dev_dependencies:
  flutter_test:
    sdk: flutter

# For information on the generic Dart part of this file, see the
# following page: https://dart.dev/tools/pub/pubspec

# The following section is specific to Flutter.
flutter:

  # The following line ensures that the Material Icons font is
  # included with your application, so that you can use the icons in
  # the material Icons class.
  uses-material-design: true

  # To add assets to your application, add an assets section, like this:
  # assets:
  #   - images/a_dot_burr.jpeg

```

该文件为Flutter的配置文件，类似于Android的module的build.gradle，用于声明Flutter App的名称、版本、作者等元数据文件，此外还包括声明依赖库和指定本地资源（图片、字体、音频、视频等）

3、总结

flutter application包含Dart代码、Android代码、ios代码，可以直接生成Android安装包和ios安装包

三、Flutter plugin

Flutter plugin主要是flutter的插件包，主要是为flutter app来调用android或者ios的功能而封装成插件，便于其他flutter app调用。相当于android中的SDK的功能

例如，目录中的android文件夹中完成音乐播放功能，然后在lib目录中的flutter代码汇总使用platform channel来使用android中的播放音乐功能，这里flutter app就可以通过使用这个插件来播放音乐

1、目录结构

FLUTTER_PLUGIN

▸ .idea

▸ android

▸ example

▸ ios

▸ lib

📁 .gitignore

≡ .metadata

≡ .packages

🕒 CHANGELOG.md

📡 flutter_plugin.iml

🔑 LICENSE

≡ pubspec.lock

! pubspec.yaml

📘 README.md

2、目录介绍

(1) android

该目录下是一个Android module工程的代码，该目录的代码会被打包进flutter plugin

(2) ios

该目录下是一个ios 静态库，该目录的代码会被打包进flutter plugin

(3) example

该目录下是一个完整的Flutter App的应用代码，为了方便plugin开发，开发时plugin的代码运行在这个flutter app上，但是打包时该目录代码不会被打包进plugin中。

(4) lib

这里是plugin中我们编写的flutter代码，使用dart来编写

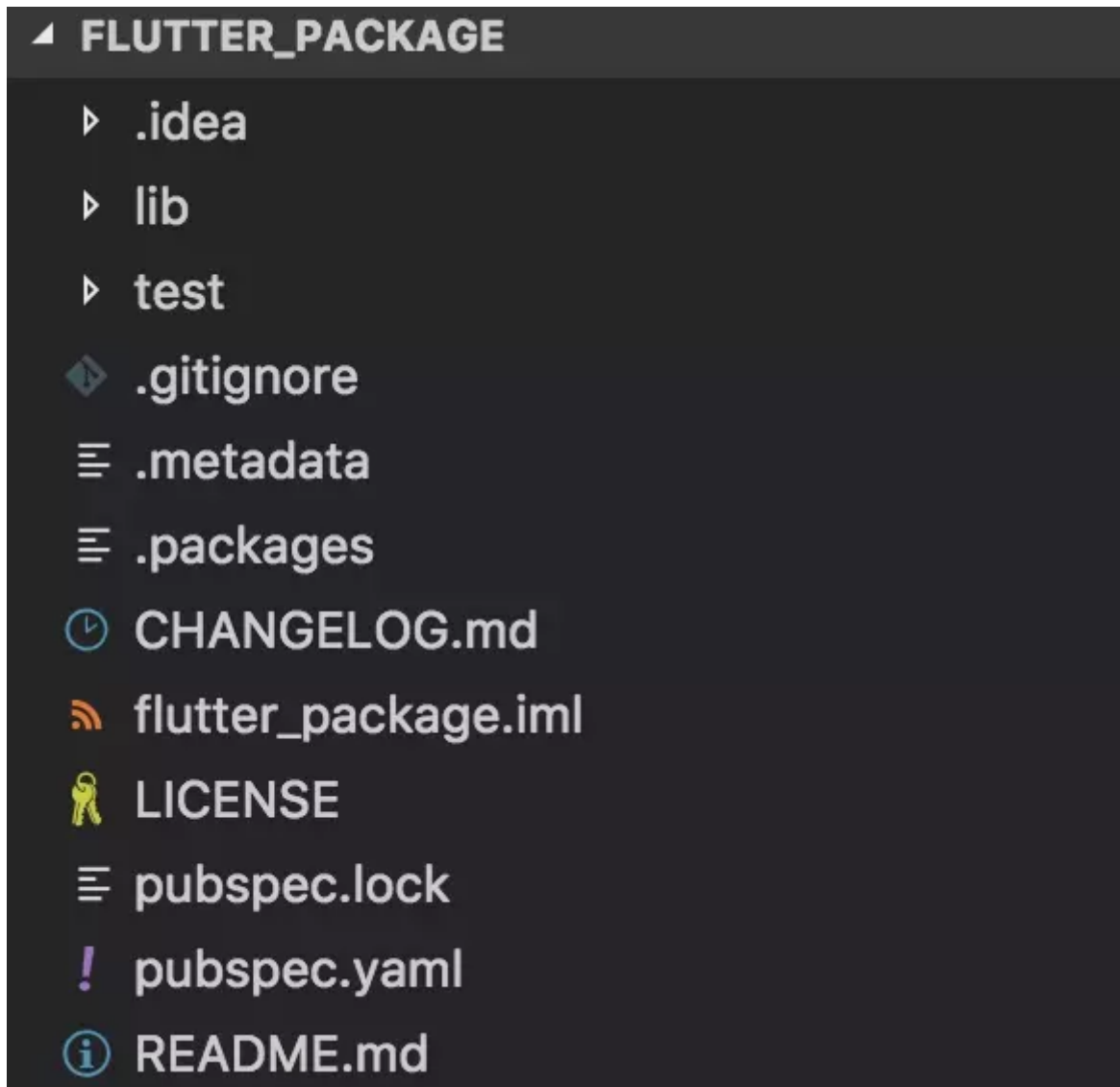
(5) pubspec.yaml文件

flutter的配置文件，前面已经介绍

四、Flutter package

flutter package是纯Dart模块，只能实现flutter相关功能。例如编写一些flutter的widget等，相当于Android项目中的module

1、目录结构



2、目录介绍

(1) lib

这里是我们编写的flutter代码

(2) test

flutter测试代码

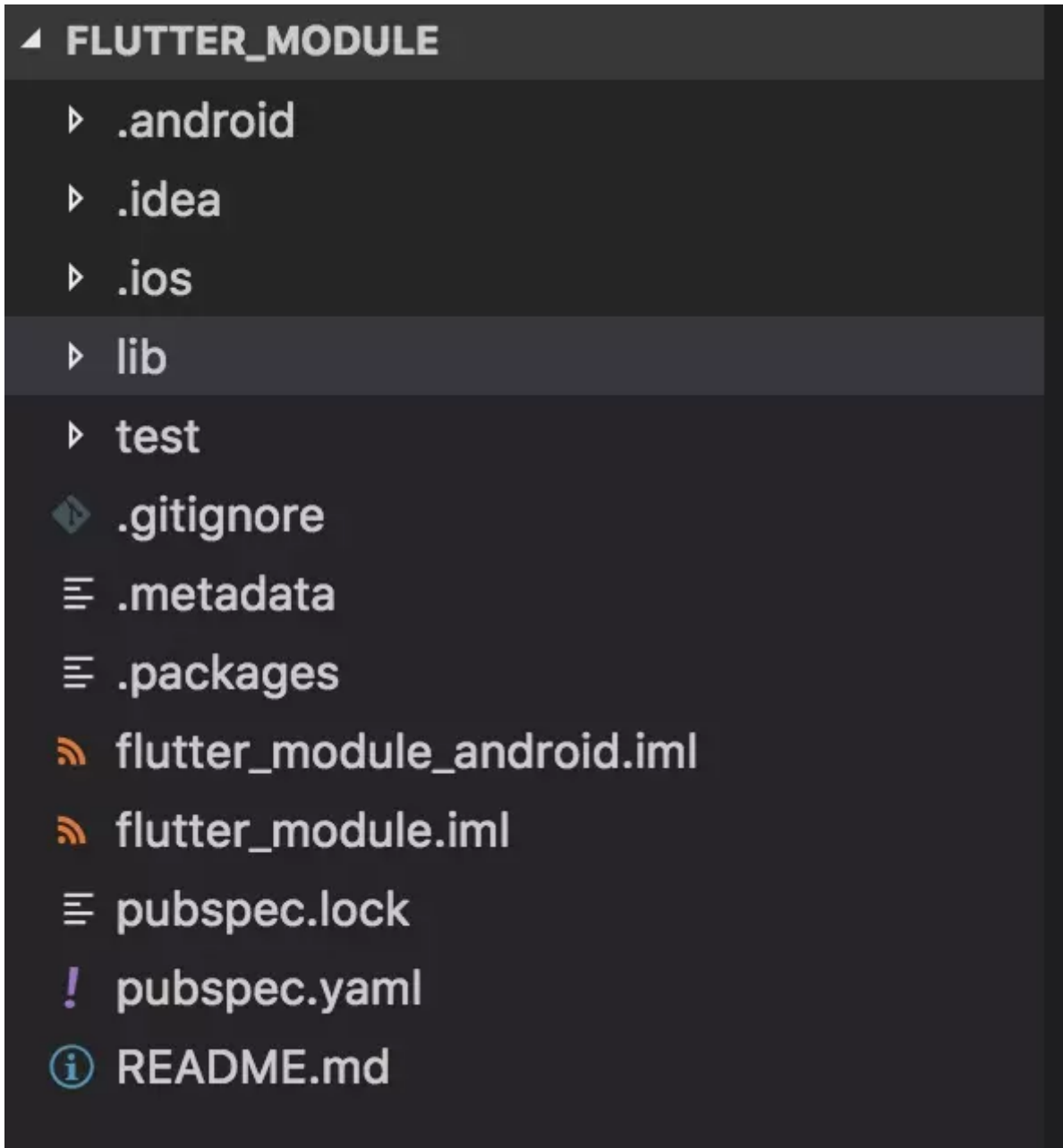
(3) pubspec.yaml

五、Flutter module

Flutter module主要是用于flutter 和Android或者ios的混合开发，可以将flutter直接添加到Android App或者ios App中。

在Android中使用时，将flutter module打包为aar包，而在ios则打包为Framework，便于集成进原有的native工程

1、目录结构



2、目录介绍

(1) .android

该目录下是一个完成的Android app工程代码，该目录只是为了便于开发和测试，因为flutter module在android上使用时是aar包，无法运行，因此加了.android目录，这

样flutter开发的时候可以直接运行在app上，但是打包的时候只有flutter目录下的android代码才会被打包进aar，.ios不会被打包进aar

(2) .ios

该目录下是一个完整的ios App工程代码，同样是为了方便开发和测试，最后打包只有ios代码会被打包进framework

(3) lib

这里是我们要编写的flutter代码

(4) test目录

测试代码

(5) pubspec.yaml

配置文件

六、pubspec.yaml介绍

pubspec.yaml文件是flutter的配置文件，使用YAML语言来编写。

```
1 name: flutter_doubanmovie
2 description: A new Flutter project.
3
4 version: 1.0.0+1
5
6 authors:
7   - Natalie Weizenbaum <nweiz@google.com>
8   - Bob Nystrom <rnystrom@google.com>
9
10 homepage: https://flutter.dev/
11
12 environment:
13   sdk: ">=2.1.0 <3.0.0"
14
15 dependencies:
16   flutter:
17     sdk: flutter
18
19   cupertino_icons: ^0.1.2
20   http: ^0.12.0+2
21   shared_preferences: ^0.5.2
22
23 dev_dependencies:
24   flutter_test:
```

```

25   sdk: flutter
26
27   flutter:
28
29   uses-material-design: true

```

下表是 pubspec.yaml 支持的字段：

字段名	含义	可选/必选
name	工程的名字	必选
description	工程的描述	想要发布到 Pub 上，就是必须的
version	工程的版本号	想要发布到 Pub 上，就是必须的
author or authors	作者名字	可选
homepage	主页	可选
environment	指定 Dart 的版本，因为 随着时间的推移，Dart 不断发展，一个软件包可能只适用于某些版本的平台。	必选
repository	指向工程的源代码的地址	可选
issue_tracker	指向跟踪工程issue的地址	可选
documentation	指向工程文档的地址	可选
dependencies	依赖的开发库	如果你的工程没有依赖的话，可以省略

dev_dependencies	依赖的测试库	如果你的工程没有依赖的话，可以省略
dependency_overrides	在开发过程中，您可能需要暂时覆盖依赖项。	如果你的工程不需要覆盖依赖的话，可以省略
executables	用于将包的可执行文件放在PATH上：可以将其一个或多个脚本公开为可以直接从命令行运行的可执行文件。	可选
publish_to	指定发布包的位置，默认是 Pub	可选
flutter	flutter 资源相关的配置，包括图片、字体等，后面会有具体场景	必选

综合来说：

- 1、可以在配置文件中指定依赖库
- 2、在配置文件中定义常量，用于在其他位置引用，例如其他plugin 等
- 3、用于配置flutter的图片、视频等引用