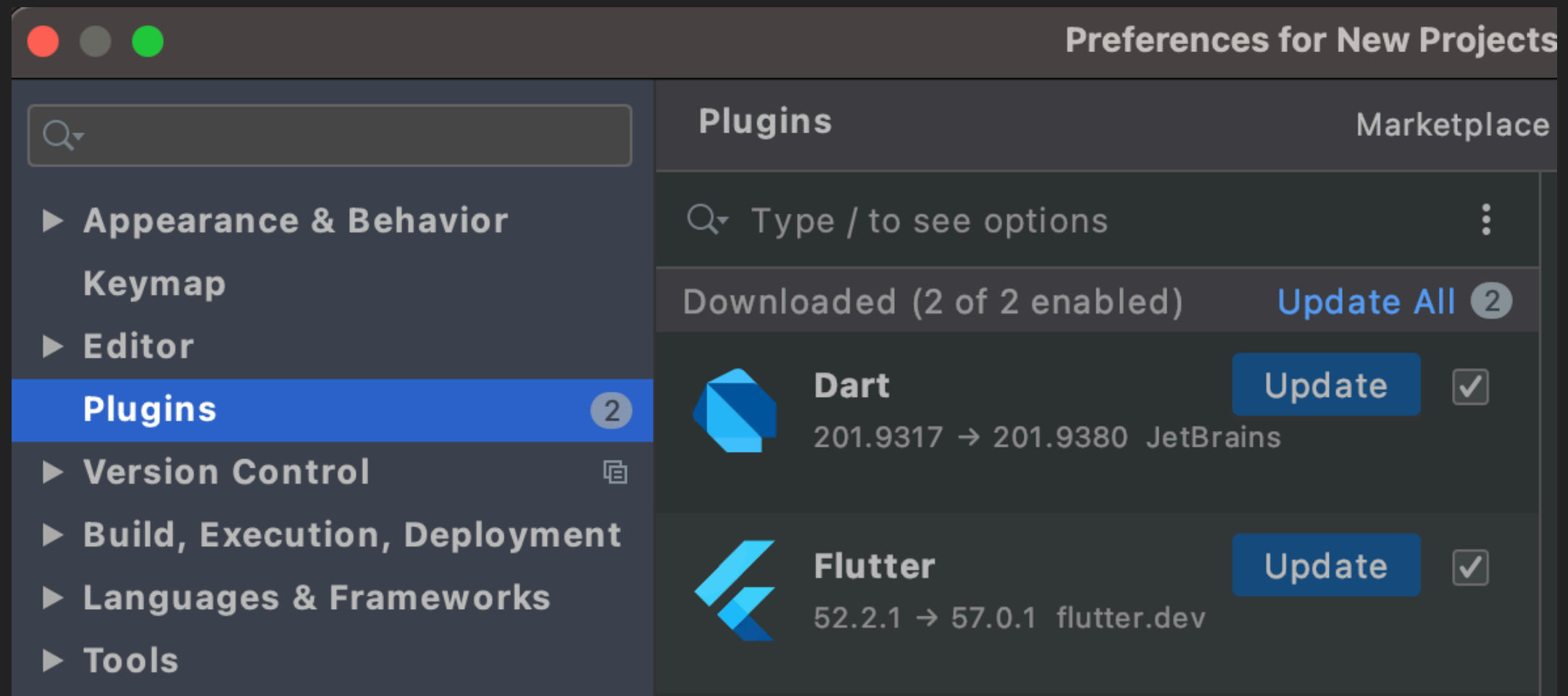


ANDROID STUDIO INTRO

LETS CREATE OUR FIRST FLUTTER
PROJECT



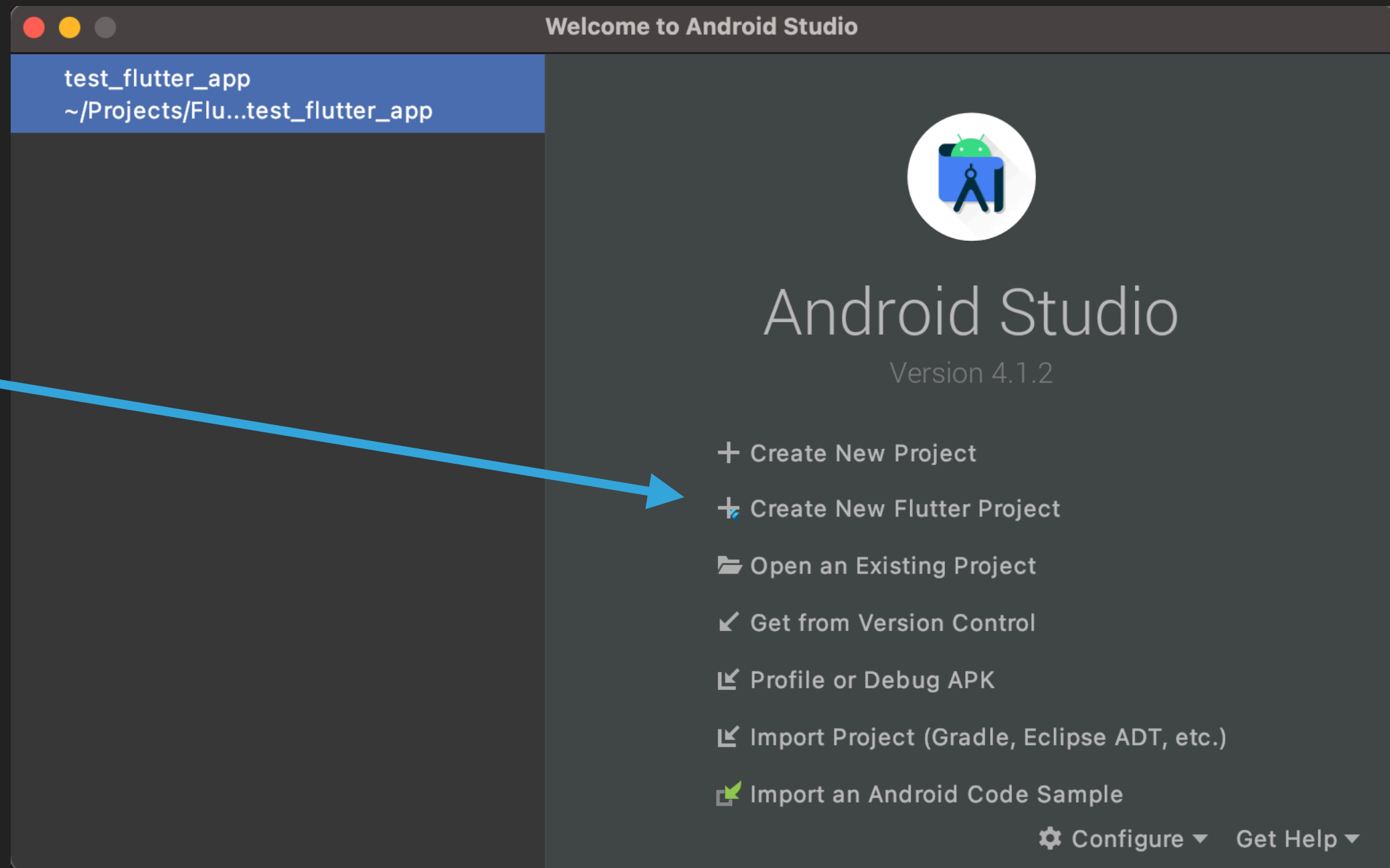
PREFERENCES -> PLUGINS -> MARKETPLACE

DART + FLUTTER SHOULD BE INSTALLED FIRST


LET'S CREATE OUR FIRST PROJECT!




CREATING PROJECT




CREATING PROJECT




New Flutter Project




Flutter Application



Flutter Plugin



Flutter Package




Flutter Module


Select an "Application" when building for end users.
 Select a "Plugin" when exposing an Android or iOS API for developers.
 Select a "Package" when creating a pure Dart component, like a new Widget.
 Select a "Module" when creating a Flutter component to add to an Android or iOS app.

WE WILL NEED JUST APPLICATION ALL THE TIME

CREATING PROJECT



New Flutter Application



NAME SHOULD BE IN SNAKE CASE

Configure the new Flutter application

Project name

flutter_course_project

Flutter SDK path

/Users/andriystefanchuk/Development/flutter

Project location

/Users/andriystefanchuk/Projects/FlutterProjects

Description

A new Flutter application.

☐ Create project offline

CREATING PROJECT



New Flutter Application



Set the package name

Applications and plugins need to generate platform-specific code

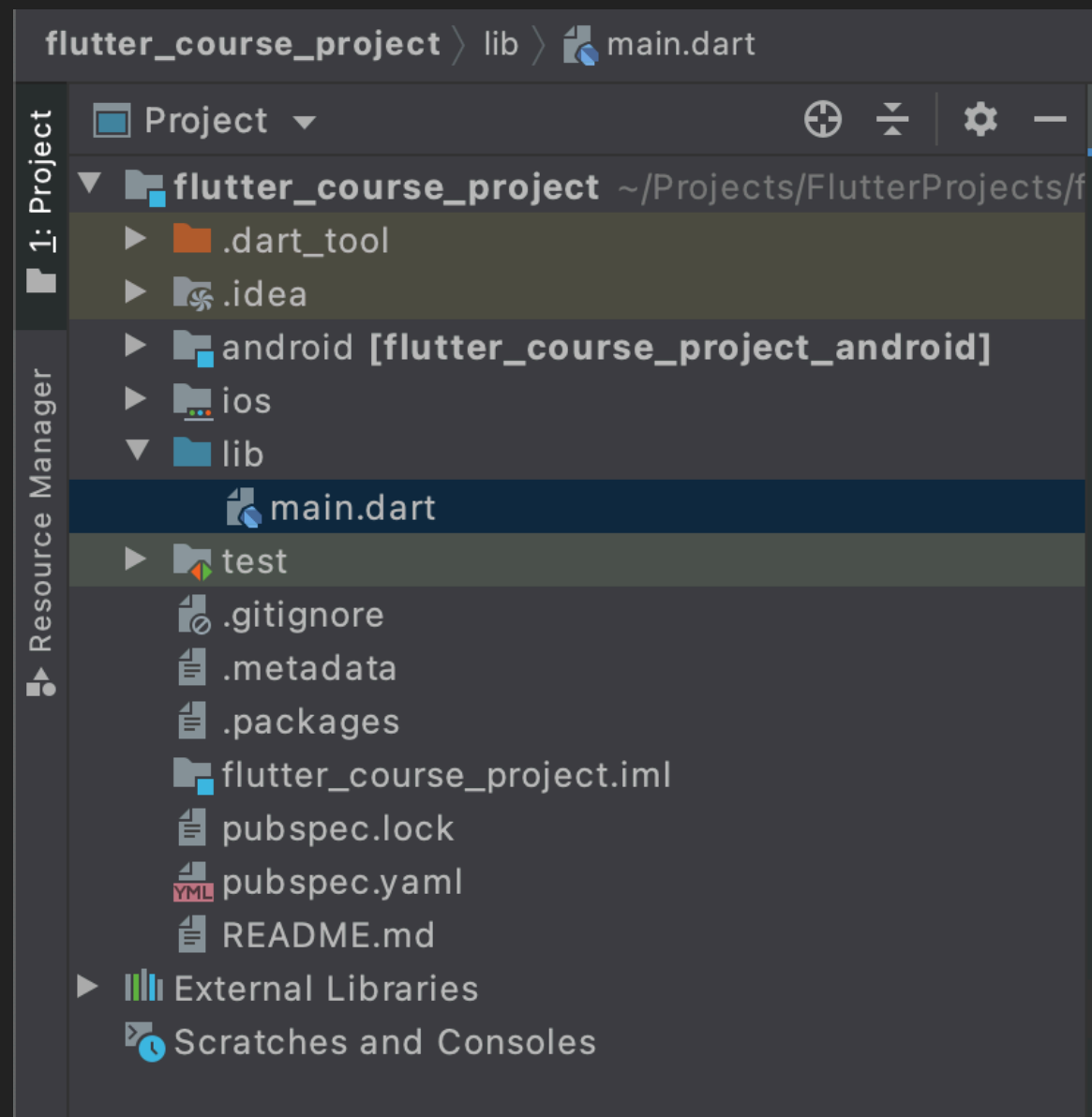
Package name

Platform channel language

- ☒ Include Kotlin support for Android code
- ☒ Include Swift support for iOS code

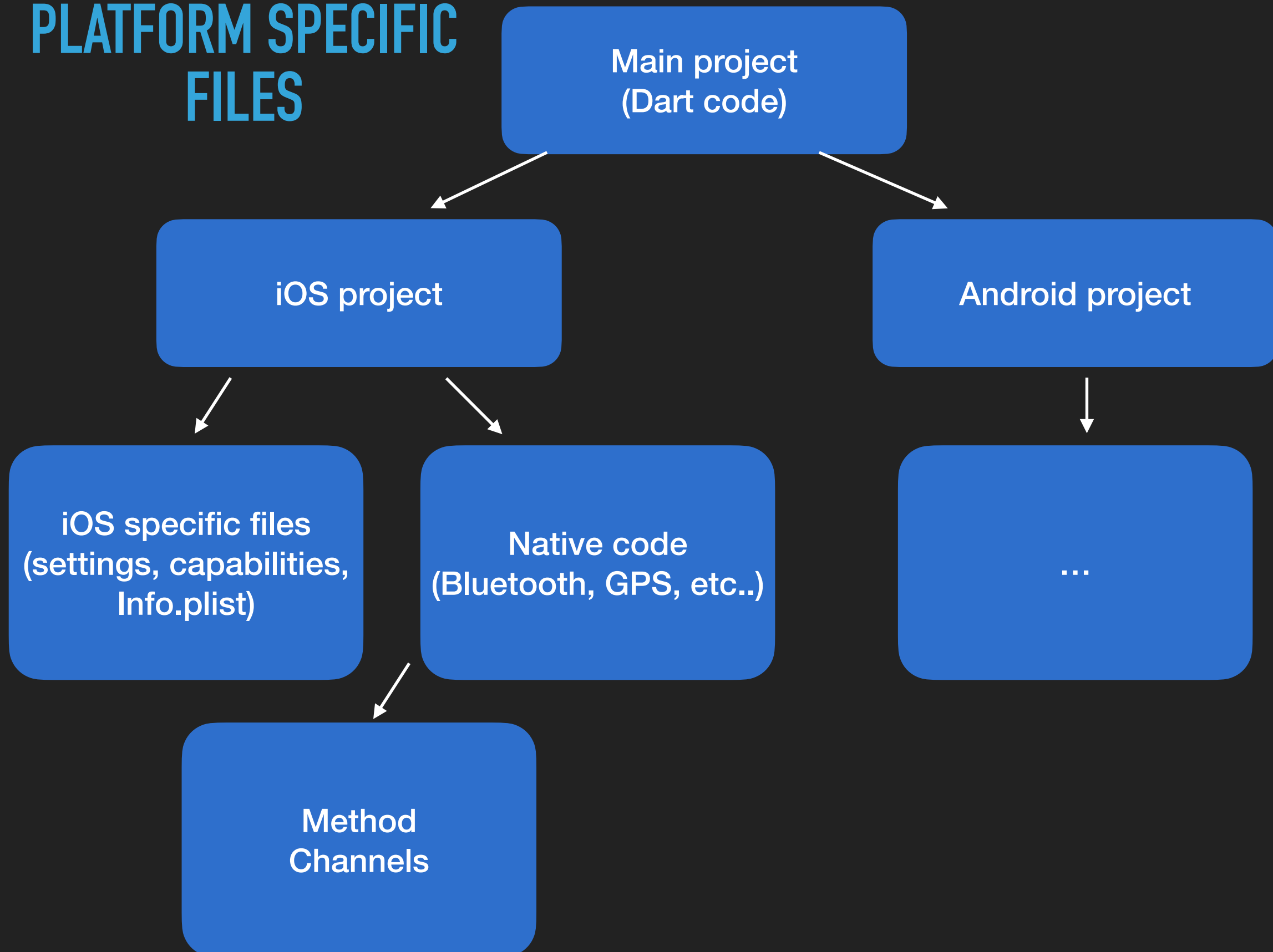
DON'T TOUCH ANYTHING HERE :)

INITIAL FILES

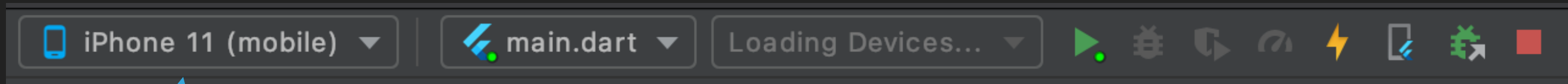


- ▶ main.dart - main entry point into the app
- ▶ pubspec.yaml - all dependencies are listed here
- ▶ All source files should be inside **lib** folder
- ▶ Android and iOS folders contain platform specific configs and files

PLATFORM SPECIFIC FILES

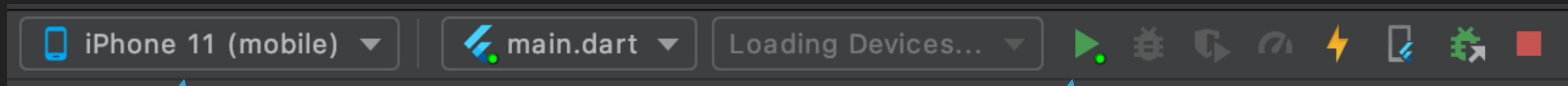


HOW TO RUN PROJECT



TARGET
DEVICE

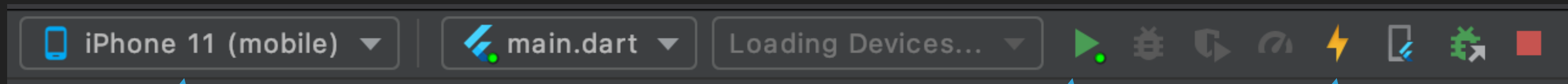
HOW TO RUN PROJECT



**TARGET
DEVICE**

RUN

HOW TO RUN PROJECT



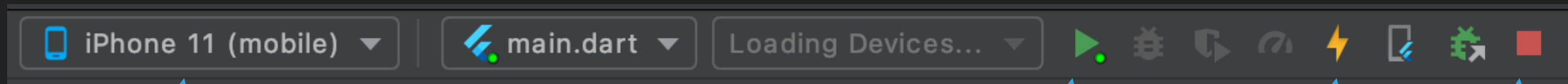
**TARGET
DEVICE**

RUN

**HOT
RELOAD**

(CAN BE DONE ON SAVE)

HOW TO RUN PROJECT



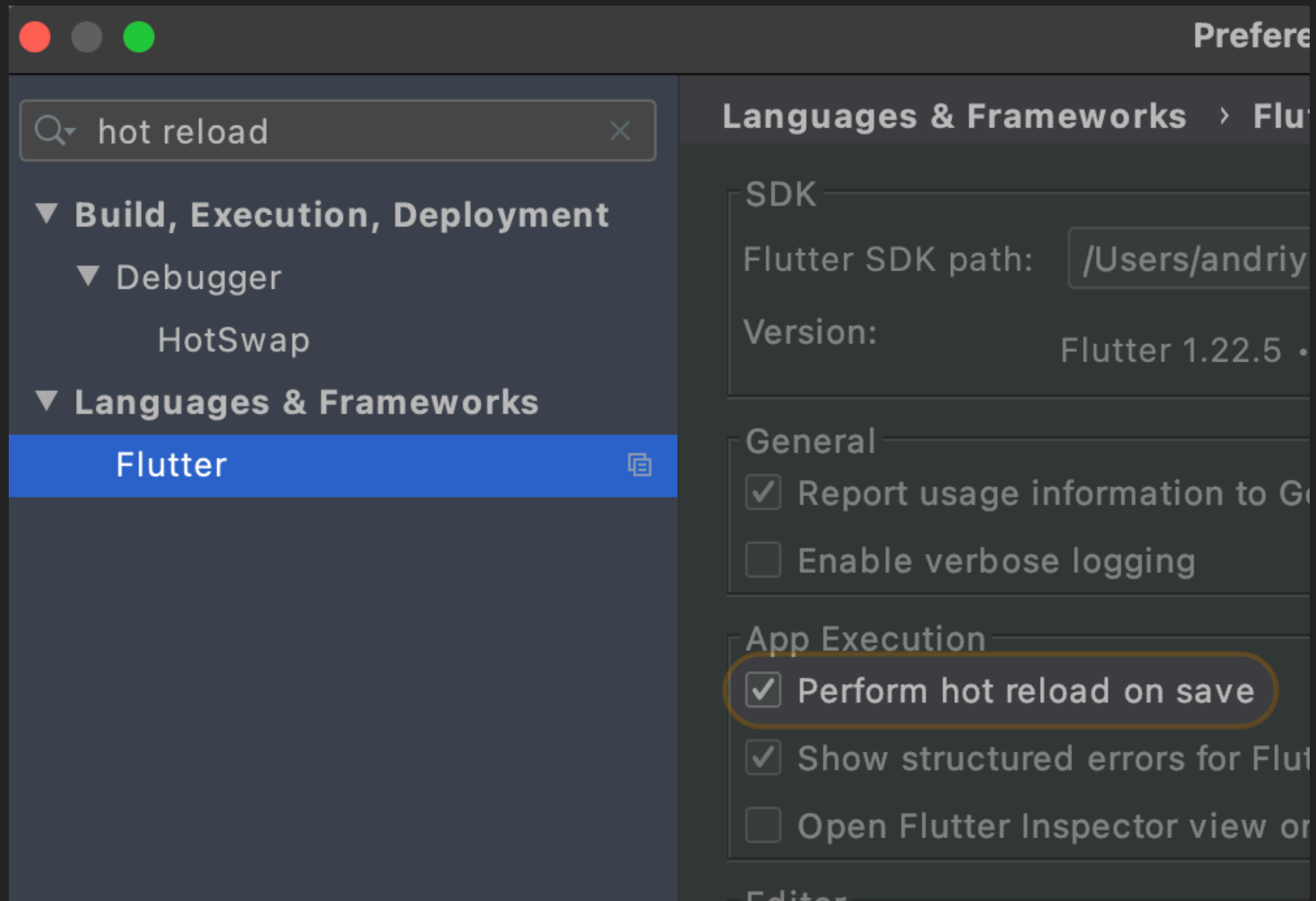
TARGET
DEVICE

RUN

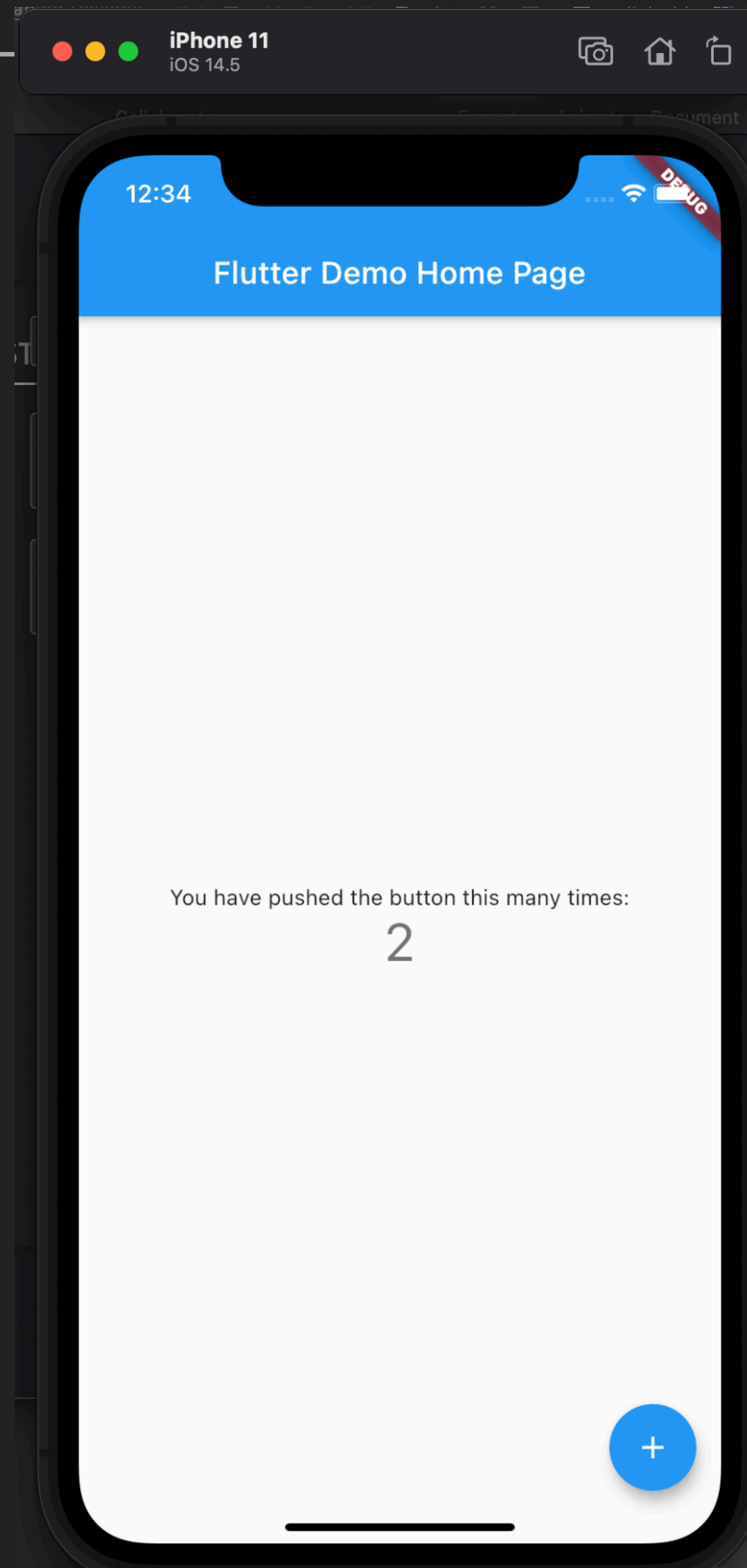
HOT
RELOAD

STOP
APP

(CAN BE DONE ON SAVE)



HOW DEMO PROJECT LOOKS LIKE



BASE CODE EXPLANATION

```
1 import 'package:flutter/material.dart';
```

- ▶ That's how you include other files, system modules or 3rd party frameworks

BASE CODE EXPLANATION

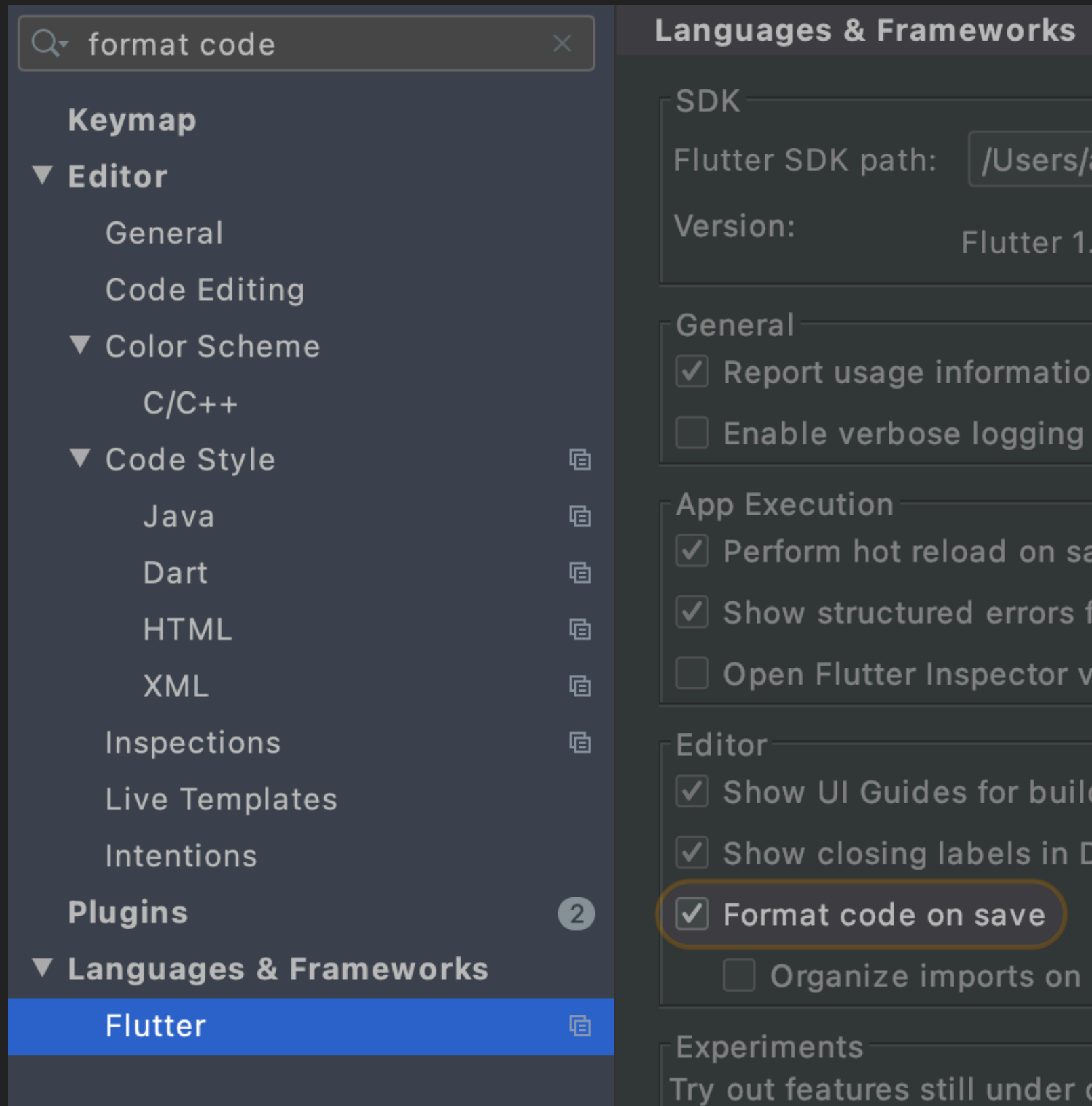
```
void main() {  
    runApp(MyApp());  
}
```

- ▶ That's how you specify initial screen shown when app starts
- ▶ MyApp is widget
- ▶ Everything in Flutter is widget
 - app, screens, UI elements, buttons, etc.

BASE CODE EXPLANATION

```
@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text(widget.title),
    ), // AppBar
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: <Widget>[
          Text(
            'You hassss pushed the button this many times:',
          ), // Text
          Text(
            '$_counter',
          ), // Text
        ], // <Widget>[]
      ), // Column
    ), // Center
    floatingActionButton: FloatingActionButton(
      onPressed: _incrementCounter,
      tooltip: 'Increment',
      child: Icon(Icons.add),
    ), // This trailing comma makes auto-formatting nicer for build methods.
  ); // Scaffold
}
```

- Our lego blocks that are included inside each other. More about that -> later





That's all Folks!