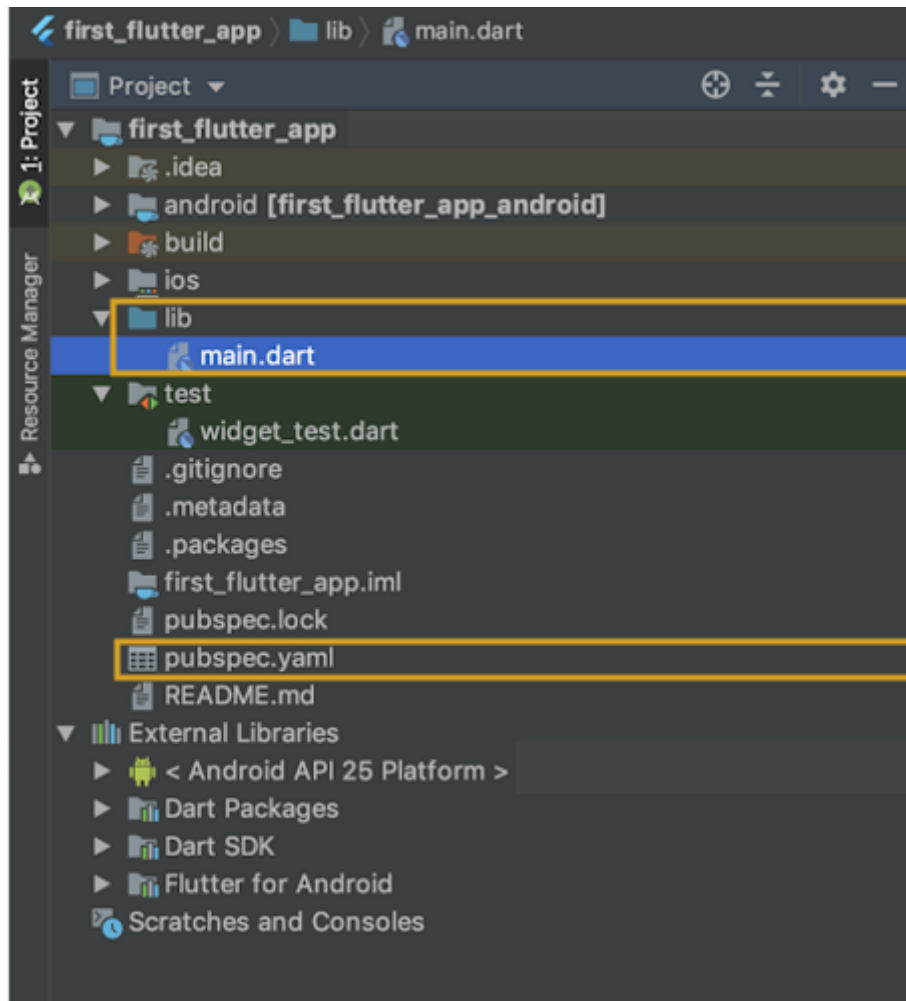


Flutter 프로젝트 구조+앱 구조

플러터 프로젝트 폴더 구조

프로젝트를 만들면 이 안에 프로젝트 파일들이 자동으로 생성됨



ios: iOS 빌드시에 필요한 파일들과 코드들이 생성

android: Android 빌드시에 필요한 파일들과 코드들이 생성

lib: 스켈레톤 앱의 코드가 들어있으며, 이후의 코드 구현은 거의 모두 이 폴더에서 구현함

pubspec.yaml: 플러터 프로젝트의 중심 같은 파일로, 앱 이름, 버전, 빌드, 의존성 (dependencies), 리소스 (이미지, 폰트 파일 등) 등이 모두 등록되어 있는 파일

[플러터 앱 구조]

```
1  import 'package:flutter/material.dart';
2
3  void main() => runApp(MyApp());
4
5  class MyApp extends StatelessWidget {...}
6
7  class MyHomePage extends StatefulWidget {...}
8
9  class _MyHomePageState extends State<MyHomePage> {...}
```

main(), MyApp(), MyHomePage 는 거의 그대로 사용함

모든 코드는 상태클래스를 참조받고 있는 _MyhomePageState에 작성

import 부분

import 'package:flutter/material.dart'; 패키지에는 머티리얼 디자인 위젯들이 포함

필수적으로 임포트해야 하는 패키지임.

필요에 따라서 ios 쿠퍼티노 패키지를 import 할 수도 있음

StatelessWidget 클래스

StatelessWidget 클래스는 클래스 이름 그대로 상태가 없는 위젯을 정의하는데 사용

상태를 가지지 않는 위젯을 구성하는 기본 클래스임

상태를 가지지 않는다는 것은 한 번 그려진 후 다시 그리지 않는 경우이며

따라서 속성변수를 가지지 않음!!!

build 메서드가 반환하는 MaterialApp

```
1 class MyApp extends StatelessWidget {  
2   @override  
3   Widget build(BuildContext context) {  
4     return MaterialApp(  
5       title: 'Welcome to Flutter',  
6       theme: ThemeData(primarySwatch: Colors.blue,  
7         home: Scaffold( ... );  
8     ),  
9   );  
1  }  
0 }  
1  
1
```

MaterialApp은 위젯은 Material design의 기본 골격을 쉽게 추가할 수 있는 위젯

child로 **Scaffold** 위젯을 추가할 수 있음.

Flutter 앱내의 모든 것들은 위젯으로 구성됨.

간단한 텍스트부터 버튼 그리고 스크린 레이아웃까지 전부 위젯이며

계층적인 순서_를 가지고 배열함

Scaffold (스캐폴드)는

기본적인 **Material design** 구조를 잡아주는 뼈대임

title은 제목임.

theme는 테마를 지정

home에 작성하는 위젯이 실제 이 앱이 표시하는 위젯임.

플러터에서 이름이 있는 인수는 클래스의 프로퍼티에 값을 할당하는 것이며,

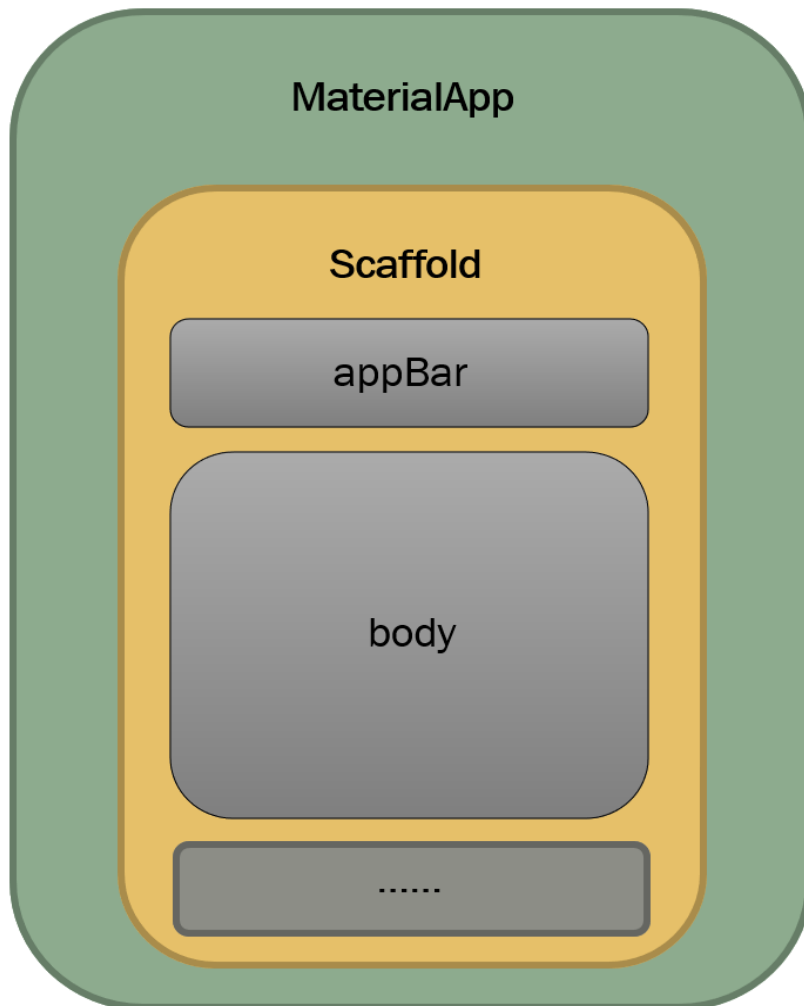
그것은 클래스의 속성을 의미함

Stateful 클래스

상태가 있는 위젯을 정의할 때는 **Stateful** 클래스를 사용함

statefulwidget은 위젯의 생명주기동안 값이 변할 수 있는 위젯 구현하기 위해서는 반드시 **State** 인스턴스를 생성하는 **StatefulWidget** 클래스를 생성해야함. **StatefulWidget** 자체는 값이 변하지 않지만 내부의 **State** 클래스가 생명주기동안 값이 변함.

MyApp(StatelessWidget)



MaterialApp의 기본 형태