

Do it! 플러터 앱 프로그래밍

1장 플러터 시작하기

목차

- 01-1 플러터의 등장 배경
- 01-2 플러터가 주목받는 이유
- 01-3 개발 환경 준비하기
- 01-4 플러터 프로젝트 시작하기

01-1 플러터의 등장 배경

- 초기의 다양한 OS 시장에서 안드로이드 , iOS 플랫폼 시장으로 변경
- 하나의 소스로 다양한 플랫폼을 이용하고자 하는 욕구 상승

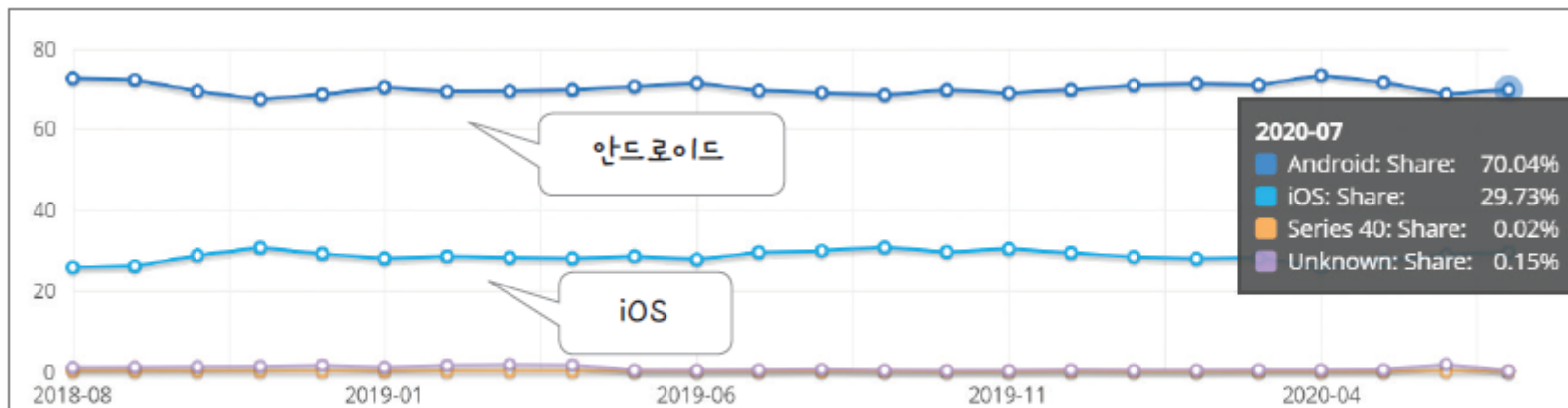


그림 1-1 연도별 모바일 운영체제 시장 점유율(netmarketshare.com, 2019.08~2020.07)

01-1 플러터의 등장 배경

- 하이브리드 형태의 앱 개발 중 네이티브의 빠른 속도가 필요해짐
- 플러터와 리엑트네이티브같은 크로스 플랫폼 개발

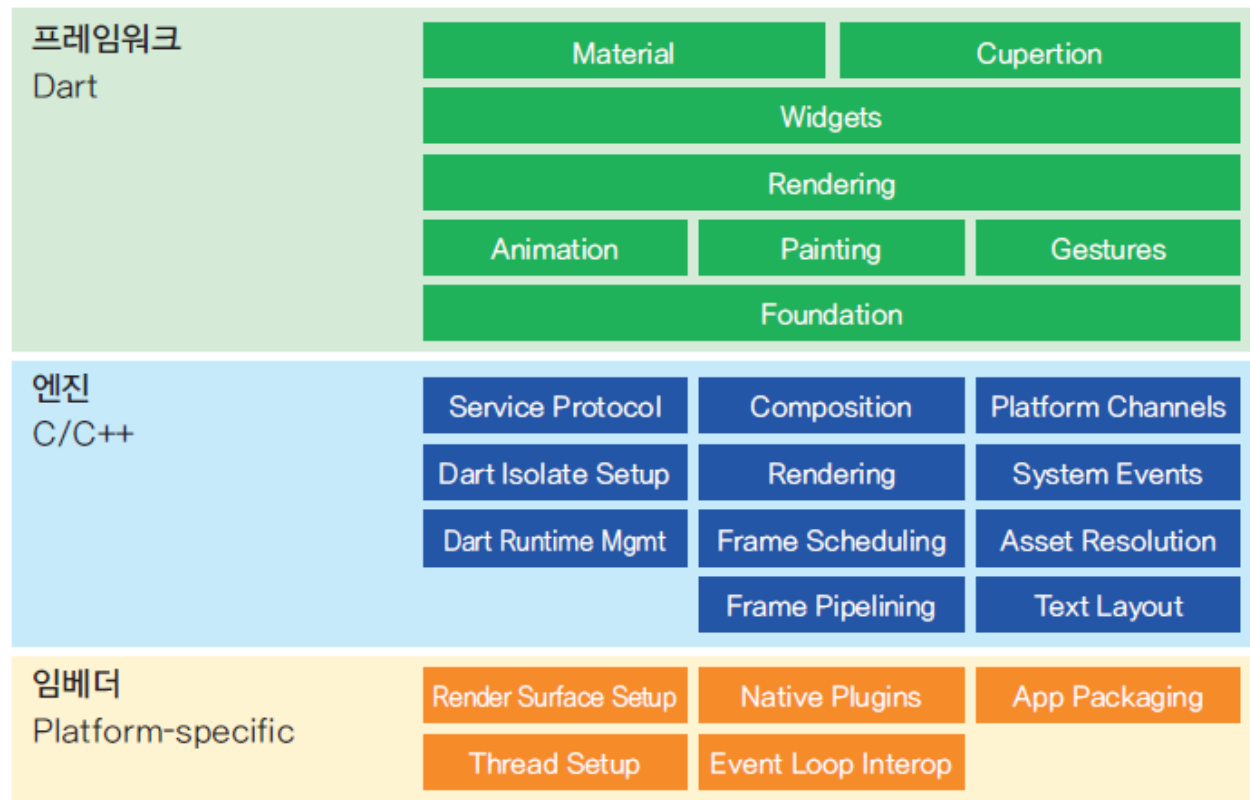


그림 1-2 플러터 프레임워크의 계층

01-1 플러터의 등장 배경

표 1-1 리액트 네이티브와 플러터 비교

구분	리액트 네이티브	플러터
개발 주체	페이스북	구글
언어	자바스크립트	다트
출시	2015년도	2017년도
성능	빠르지만 네이티브 앱만큼은 아님	네이티브 앱에 근접한 속도
학습 곡선	높음(네이티브 앱 개발자 기준)	낮음(네이티브 앱 개발자 기준)
대표 앱	페이스북, 인스타그램, 핀터레스트 등	알리바바, 구글 애드센스, 리플렉틀리 등
장점	<ul style="list-style-type: none">• 저변이 넓은 자바스크립트 생태계• 웹 개발자의 접근성• npm으로 많은 패키지 이용 가능	<ul style="list-style-type: none">• 다양한 위젯• 강력한 애니메이션 성능• 블루투스 등 네이티브 하드웨어와의 연결성
단점	<ul style="list-style-type: none">• 기본 위젯이 부족해 커스텀해 사용• Android/iOS 네이티브 위젯을 이용하기에 OS 판올림에 따른 업데이트 필요• 블루투스 등 네이티브 커스텀해 통신하는 부분 개발이 어려움	<ul style="list-style-type: none">• 플러터 SDK로 앱 크기가 큼(네이티브 대비)• 아직 개발 생태계가 성숙하지 않아 빠른 피드백 얻기가 어려움• 업데이트 주기가 빠름(분기별)
최종 목표	자바스크립트로 웹, 앱, 데스크톱 모든 플랫폼을 개발할 수 있는 통합 솔루션 개발	안드로이드, iOS, 웹, 윈도우 10 앱을 같은 코드로 개발할 수 있는 플랫폼 개발

01-1 플러터의 등장 배경

표 1-1 리액트 네이티브와 플러터 비교

구분	리액트 네이티브	플러터
개발 주체	페이스북	구글
언어	자바스크립트	다트
출시	2015년도	2017년도
성능	빠르지만 네이티브 앱만큼은 아님	네이티브 앱에 근접한 속도
학습 곡선	높음(네이티브 앱 개발자 기준)	낮음(네이티브 앱 개발자 기준)
대표 앱	페이스북, 인스타그램, 핀터레스트 등	알리바바, 구글 애드센스, 리플렉틀리 등
장점	<ul style="list-style-type: none">• 저변이 넓은 자바스크립트 생태계• 웹 개발자의 접근성• npm으로 많은 패키지 이용 가능	<ul style="list-style-type: none">• 다양한 위젯• 강력한 애니메이션 성능• 블루투스 등 네이티브 하드웨어와의 연결성
단점	<ul style="list-style-type: none">• 기본 위젯이 부족해 커스텀해 사용• Android/iOS 네이티브 위젯을 이용하기에 OS 판올림에 따른 업데이트 필요• 블루투스 등 네이티브 커스텀해 통신하는 부분 개발이 어려움	<ul style="list-style-type: none">• 플러터 SDK로 앱 크기가 큼(네이티브 대비)• 아직 개발 생태계가 성숙하지 않아 빠른 피드백 얻기가 어려움• 업데이트 주기가 빠름(분기별)
최종 목표	자바스크립트로 웹, 앱, 데스크톱 모든 플랫폼을 개발할 수 있는 통합 솔루션 개발	안드로이드, iOS, 웹, 윈도우 10 앱을 같은 코드로 개발할 수 있는 플랫폼 개발

01-2 플러터가 주목받는 이유

- 높은 개발 효율

- 플러터를 이용하면 안드로이드와 iOS 앱을 동시에 개발할 수 있어서 효율적입니다
- 핫 리로드(hot reload)

- 유연한 사용자 인터페이스

- 다양한 위젯(widget) 을 제공하므로 사용자 맞춤형 앱을 쉽게 제작가능
- 강력한 애니메이션 기능을 제공하여 복잡한 계산식 없이 적은 노력으로 만족스러운 사용자 경험 제공

- 빠른 속도

- 스킴아 (skia) 엔진을 이용한 초당 60프레임 이상의 속도로 화면을 갱신

01-2 플러터가 주목받는 이유

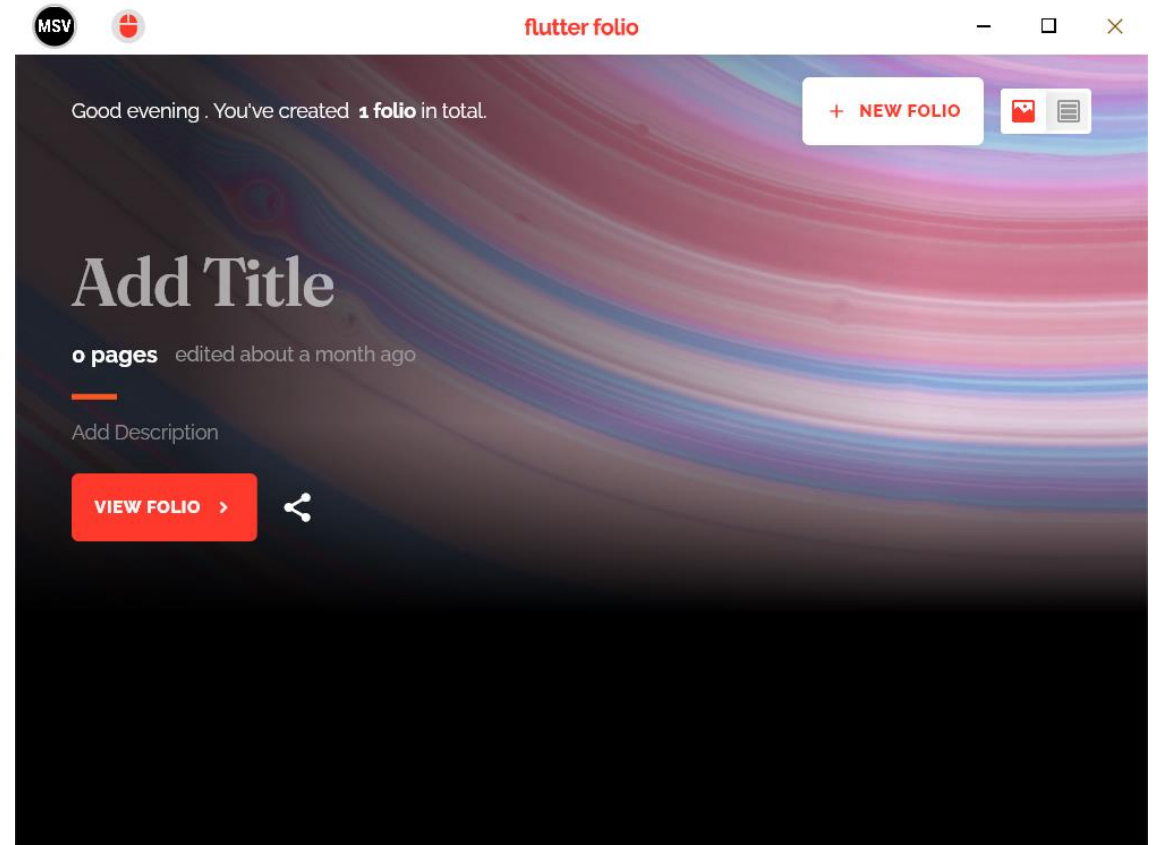
- Mobile , Web , Desktop, Embedded 다양한 플랫폼에서 하나의 소스를 이용한 개발 가능



그림 1-6 플러터의 최종 목표

01-2 플러터가 주목받는 이유

- 2.0 부터 Windows , Web 개발이 beta -> Stable 로 변경
- 계속 개발 및 변화중



01-3 개발 환경 준비하기

- Android Studio 설치

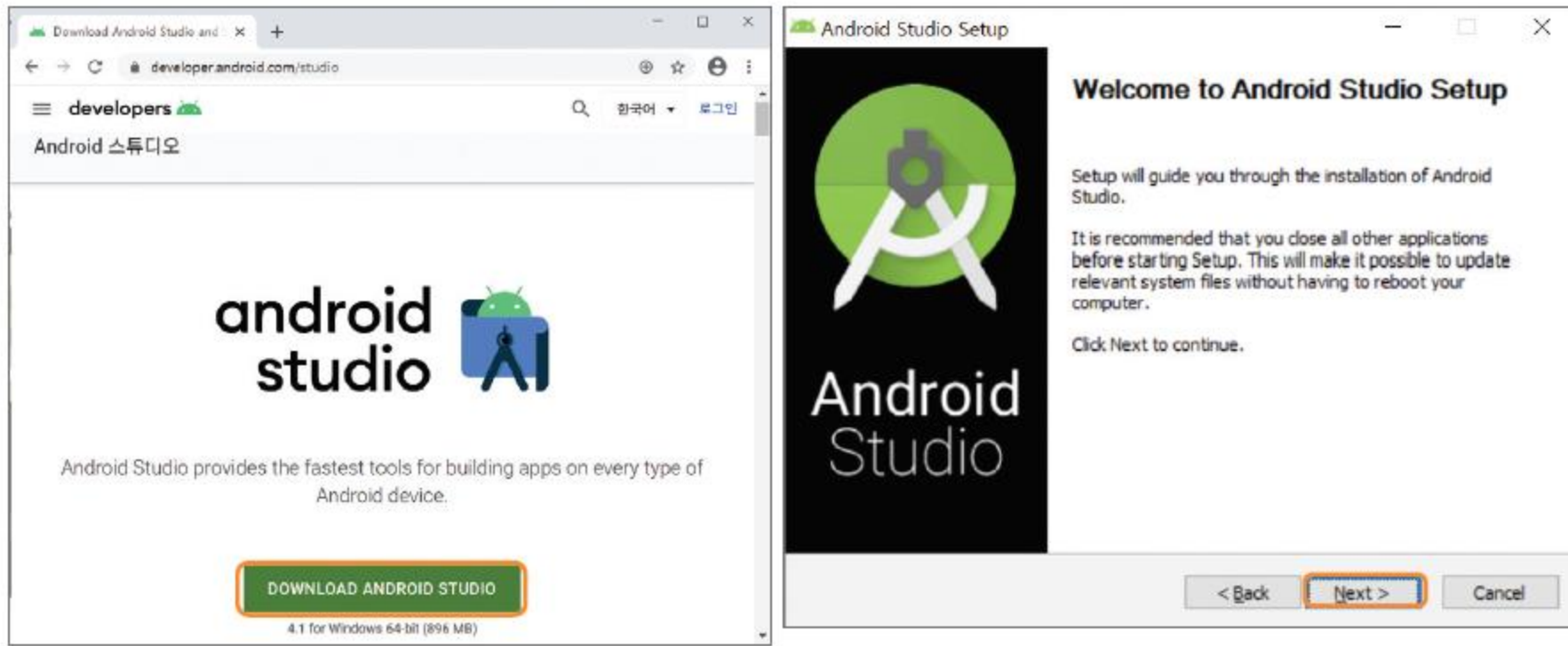


그림 1-8 안드로이드 스튜디오 설치

01-3 개발 환경 준비하기

- 플러터 SDK 설치 (flutter.dev/docs/get-started/install)

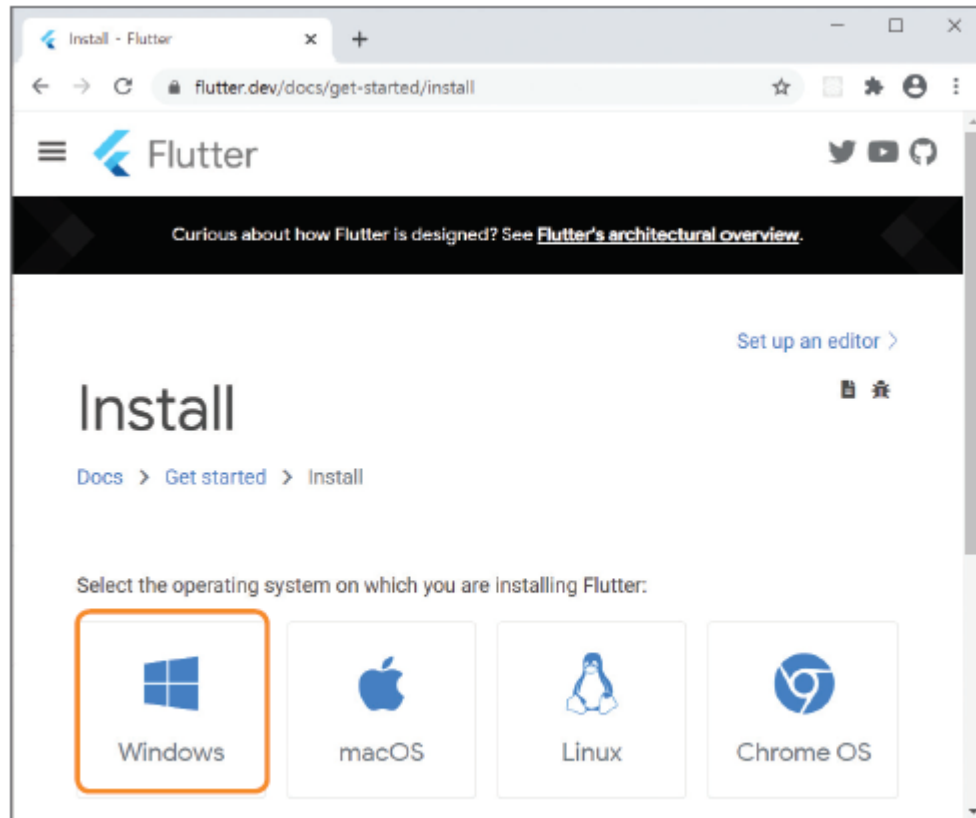


그림 1-13 플러터 설치 사이트

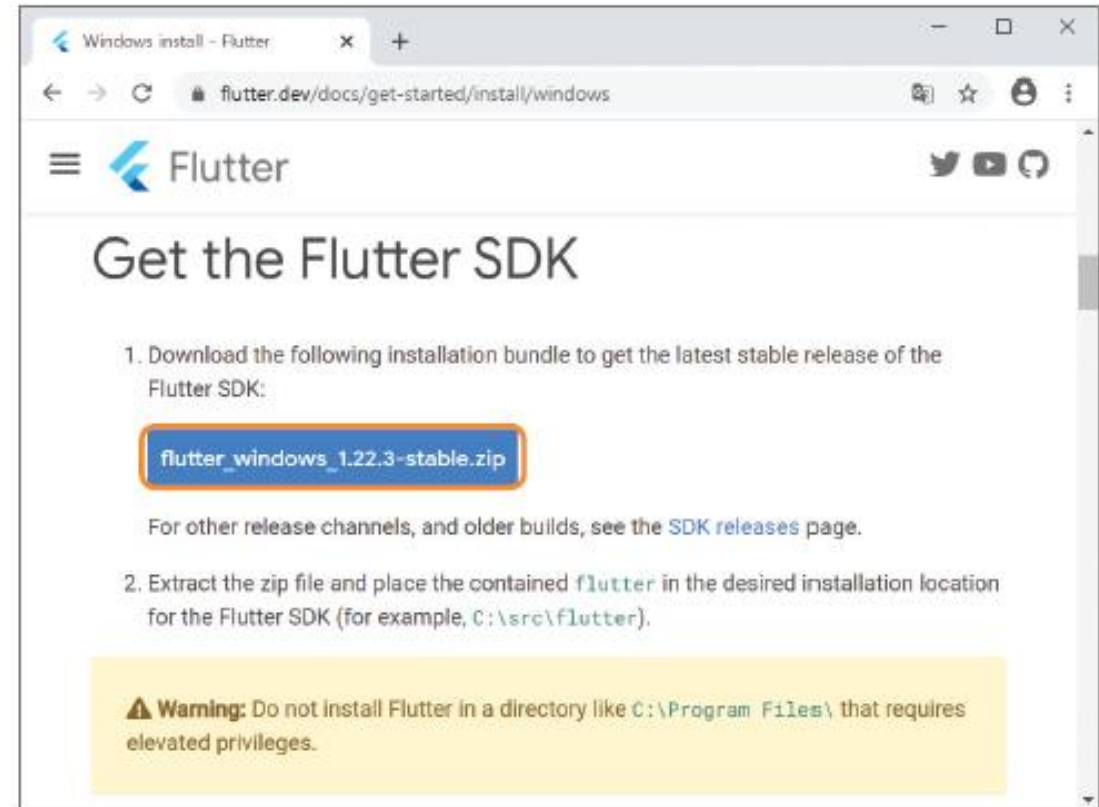


그림 1-14 플러터 SDK 설치 파일 내려받기

01-3 개발 환경 준비하기

- 설치된 파일에서 flutter doctor 실행



```
명령 프롬프트
D:\flutter\bin>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 1.20.2, on Microsoft Windows [Version 10.0.18363.1016], locale ko-KR)

[!] Android toolchain - develop for Android devices (Android SDK version 29.0.3)
    ! Some Android licenses not accepted. To resolve this, run: flutter doctor --android-licenses
[!] Android Studio (version 4.0)
    X Flutter plugin not installed; this adds Flutter specific functionality.
    X Dart plugin not installed; this adds Dart specific functionality.
[!] Connected device
    ! No devices available

! Doctor found issues in 3 categories.
D:\flutter\bin>
```

그림 1-16 flutter doctor 실행 결과

01-3 개발 환경 준비하기

- 플러터와 다트 플러그인 설치

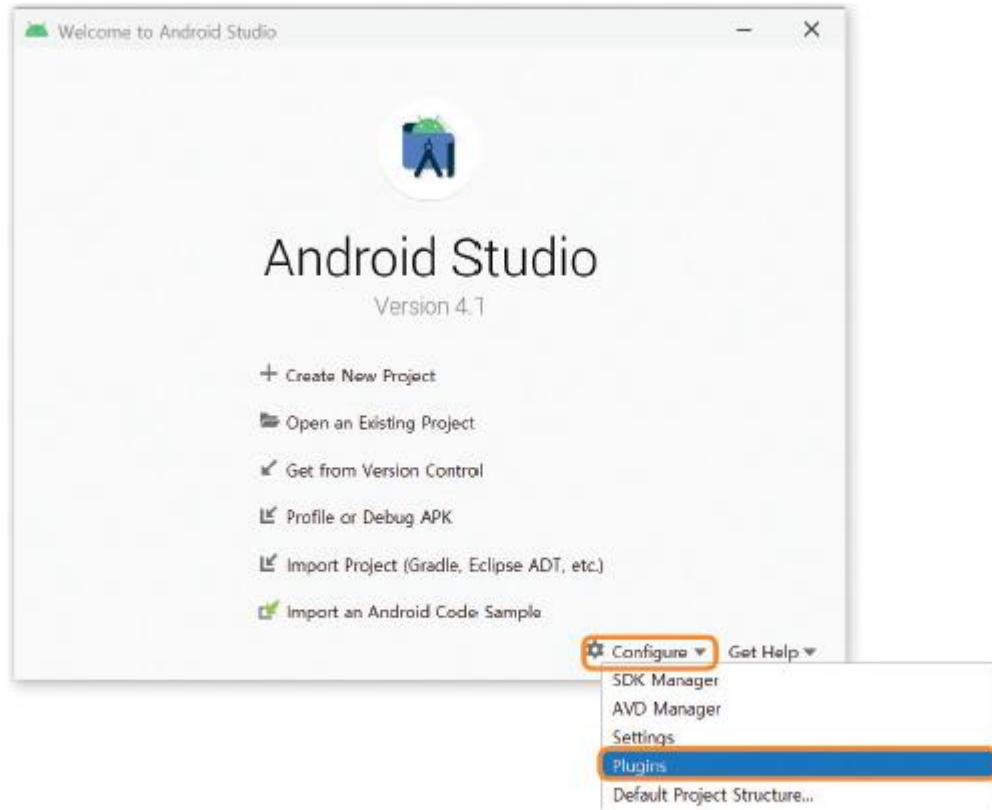


그림 1-17 안드로이드 스튜디오 플러그인 창 열기

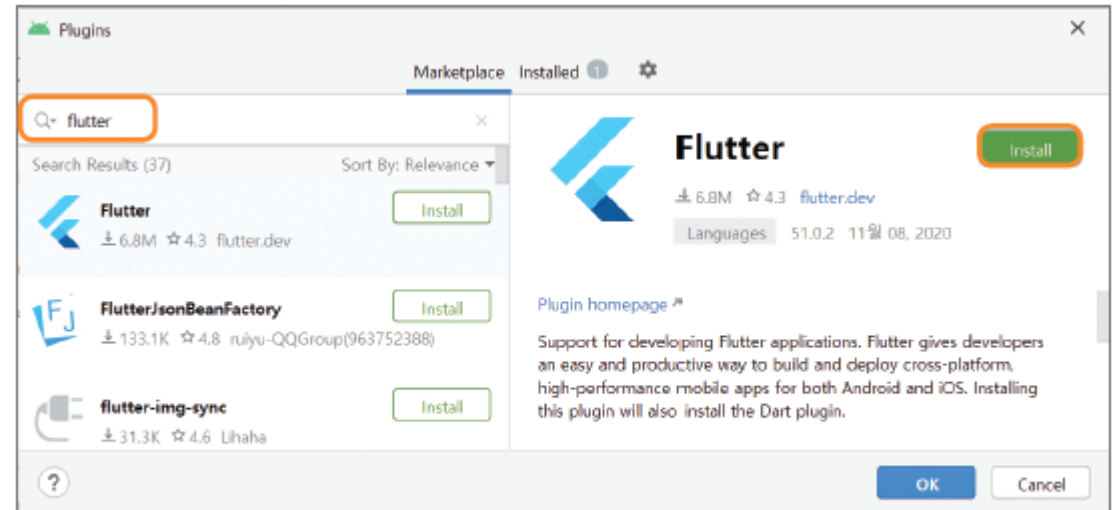


그림 1-18 안드로이드 스튜디오 플러터 플러그인 설치

01-3 개발 환경 준비하기

- 재시작 후 Flutter 프로젝트 시작하기

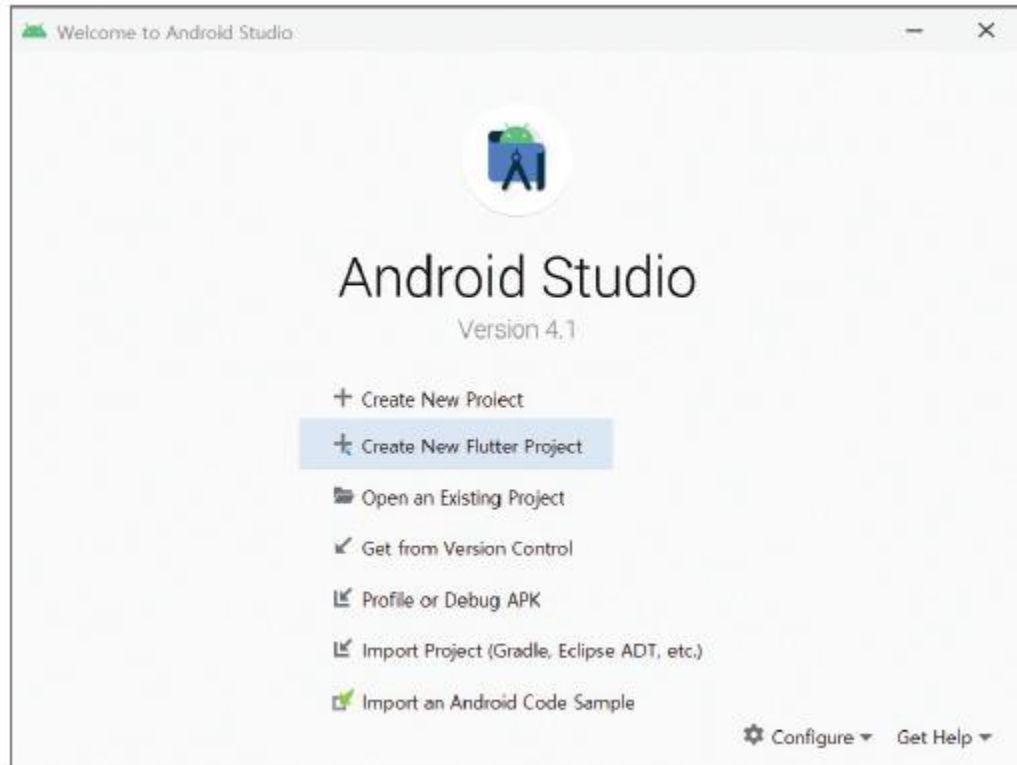


그림 1-20 플러터 프로젝트 생성 메뉴

01-4 플러터 프로젝트 시작하기

1단계 안드로이드 스튜디오를 실행하고 시작 화면에서 [Create New Flutter Project]를 선택하면 다음처럼 4가지 옵션이 나옵니다. 우리는 앱을 만드는 것이 목표이므로 [Flutter Application]을 선택한 후 <Next>를 클릭합니다.

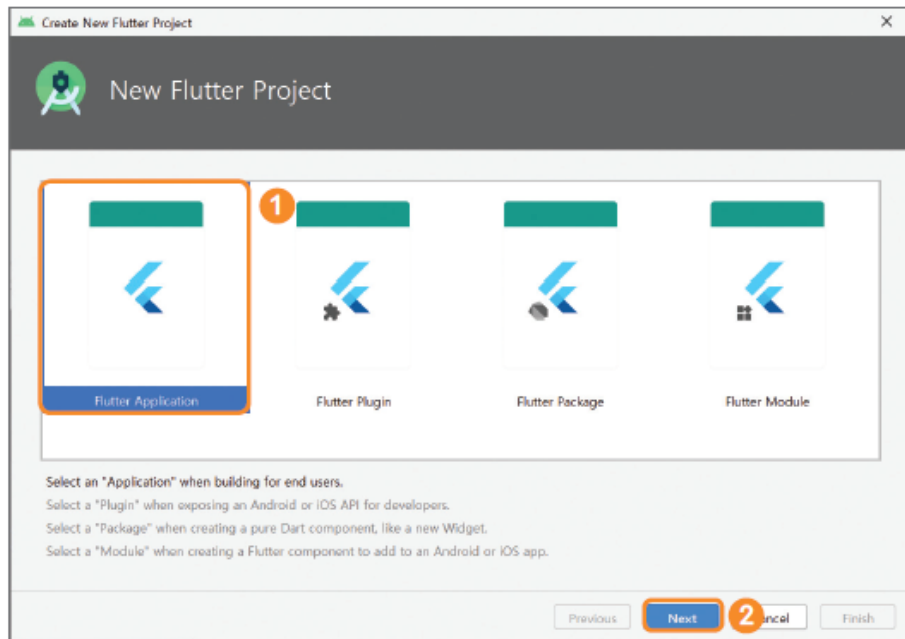


그림 1-21 플러터 프로젝트 생성 화면

01-4 플러터 프로젝트 시작하기

- 프로젝트 설정화면 입력

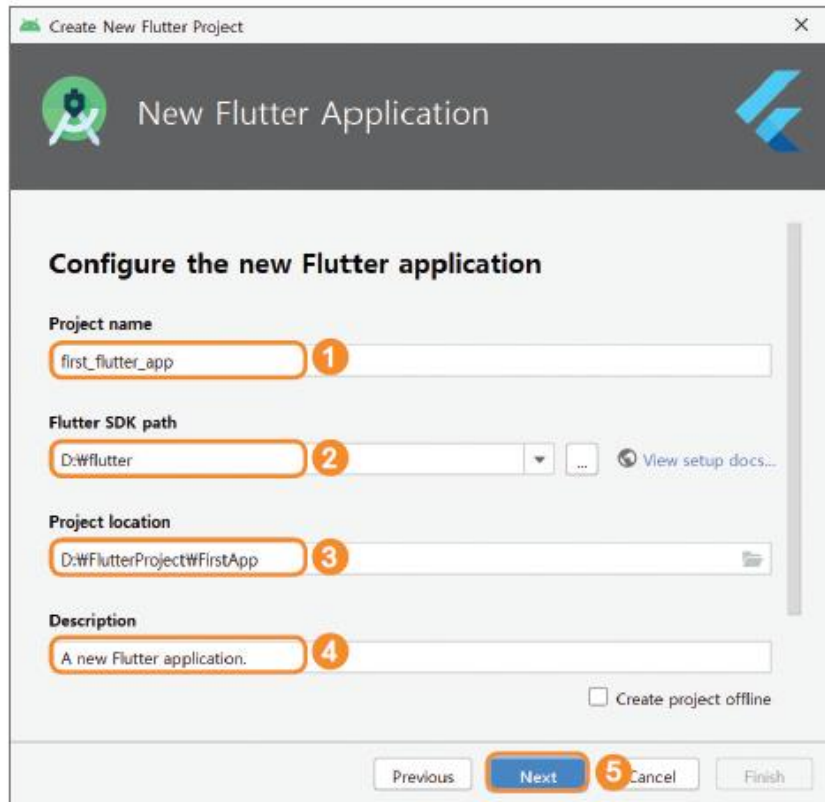


그림 1-22 플러터 프로젝트 설정 화면

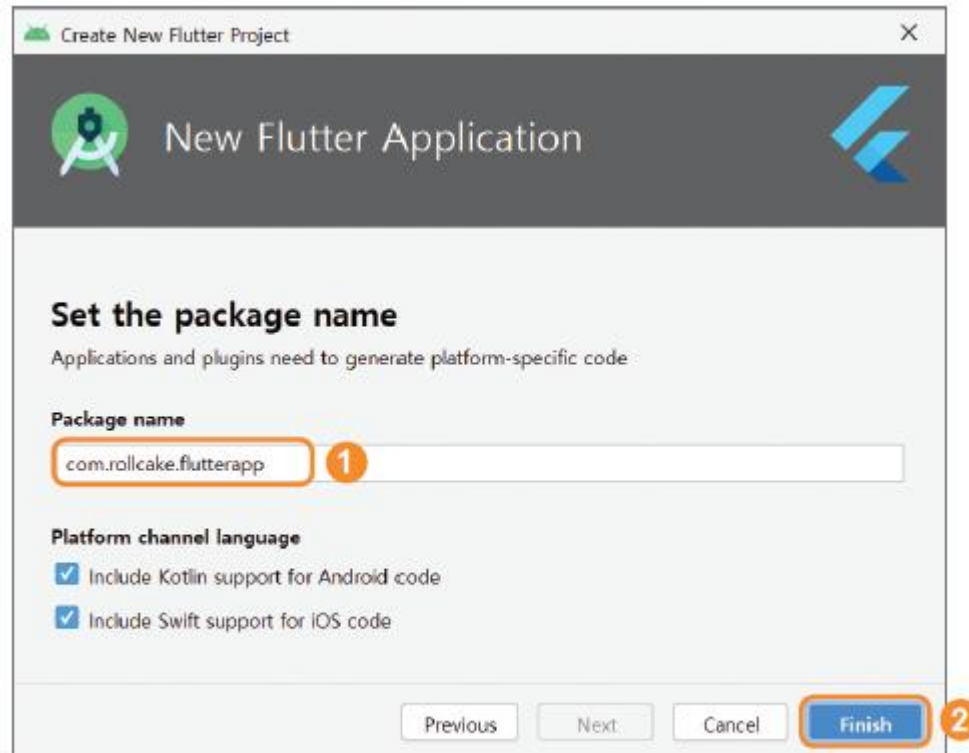


그림 1-23 플러터 프로젝트 패키지 이름 설정 화면

01-4 플러터 프로젝트 시작하기

- 완성된 프로젝트에 안드로이드 에뮬레이터로 빌드하기

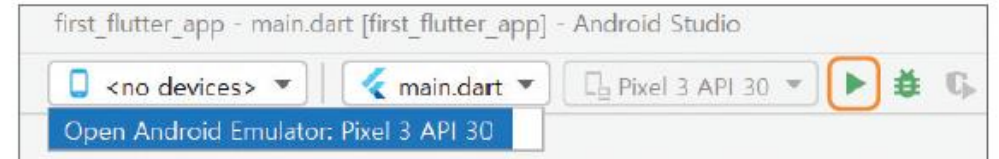
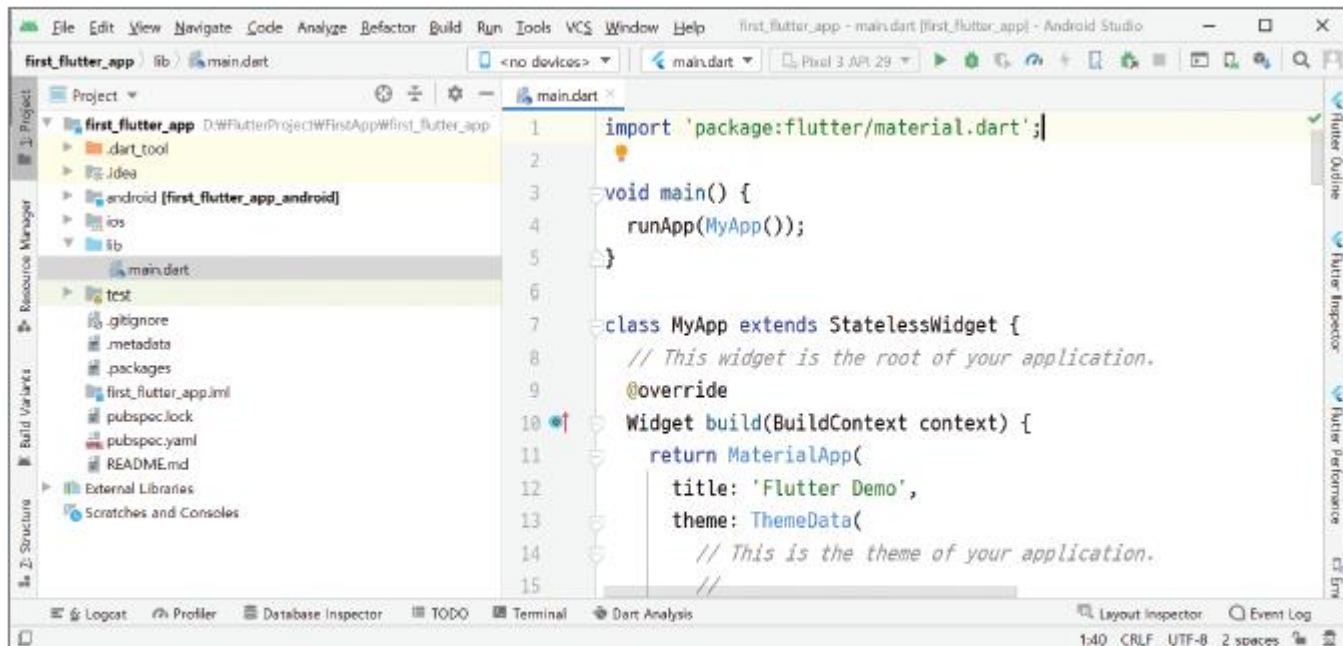
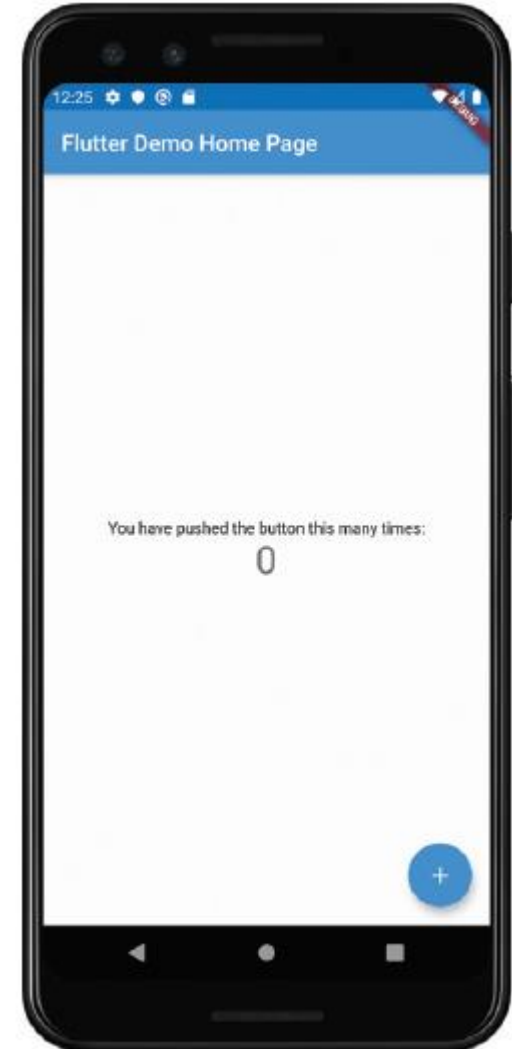


그림 1-24 플러터 데모 프로젝트가 열린 안드로이드 스튜디오

01-4 플러터 프로젝트 시작하기

- 빌드 완료



▶ 실행 결과