

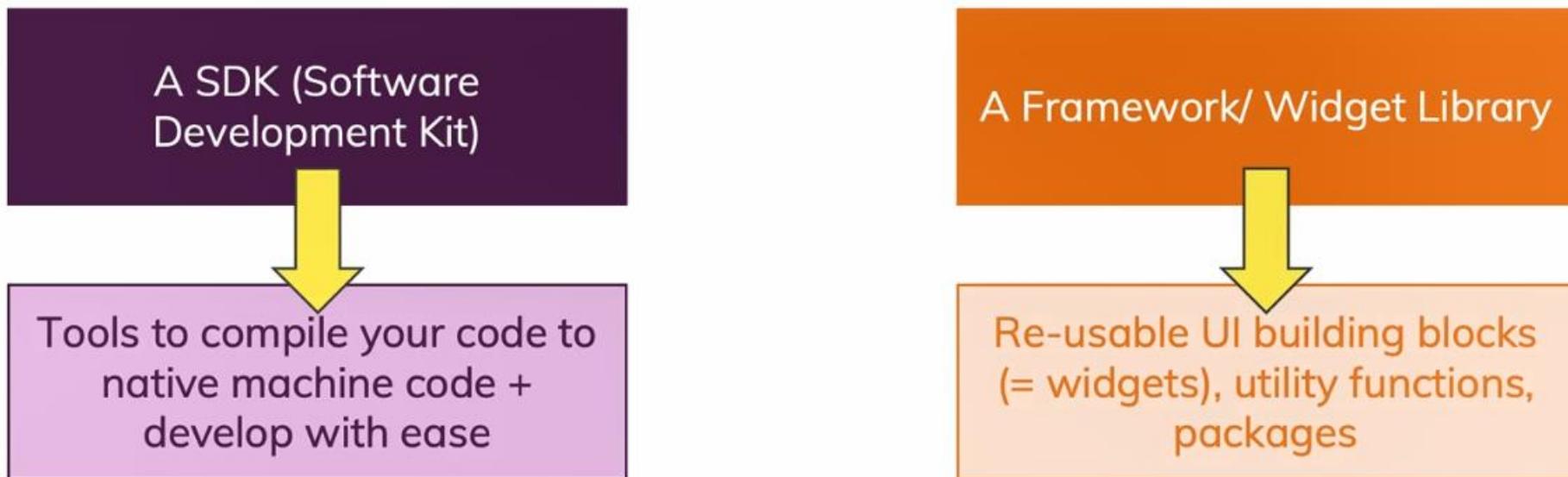
WHAT IS FLUTTER

What is Flutter?

A “tool” that allows you to build native cross-platform (iOS, Android) apps with
one programming language and codebase.

What is Flutter?

A “tool” that allows you to build native cross-platform (iOS, Android) apps with
one programming language and codebase.



Dart?



Focused on **frontend** (mobile apps, web) **user interface (UI)** development

Dart?

Programming language developed by **Google**

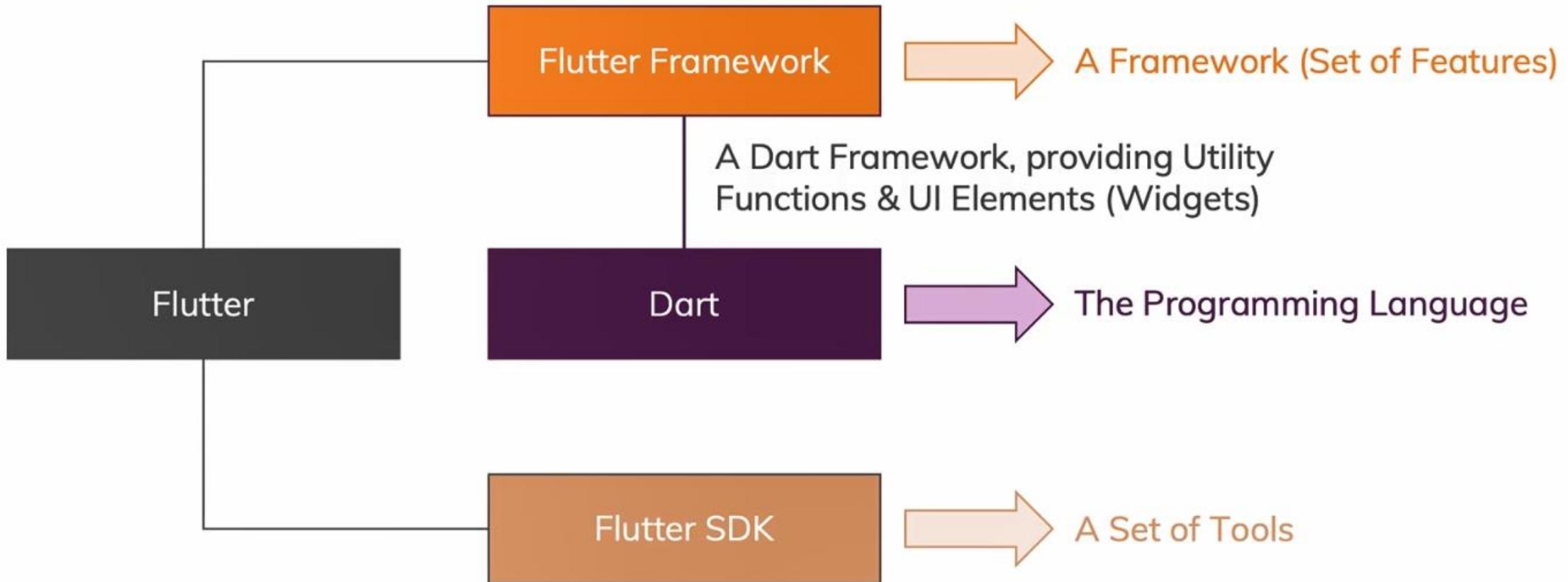
Object-oriented &
Strongly Typed



Syntax is like a mixture of
JavaScript, Java, C# (you don't
need to know these languages!)

Focused on **frontend** (mobile apps, web) **user interface (UI)**
development

Flutter vs Dart

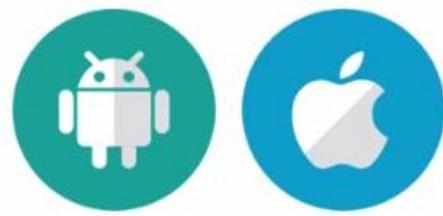


UNDERSTANDING THE FLUTTER ARCHITECTURE

Flutter Architecture



UI as Code: Build a
Widget Tree

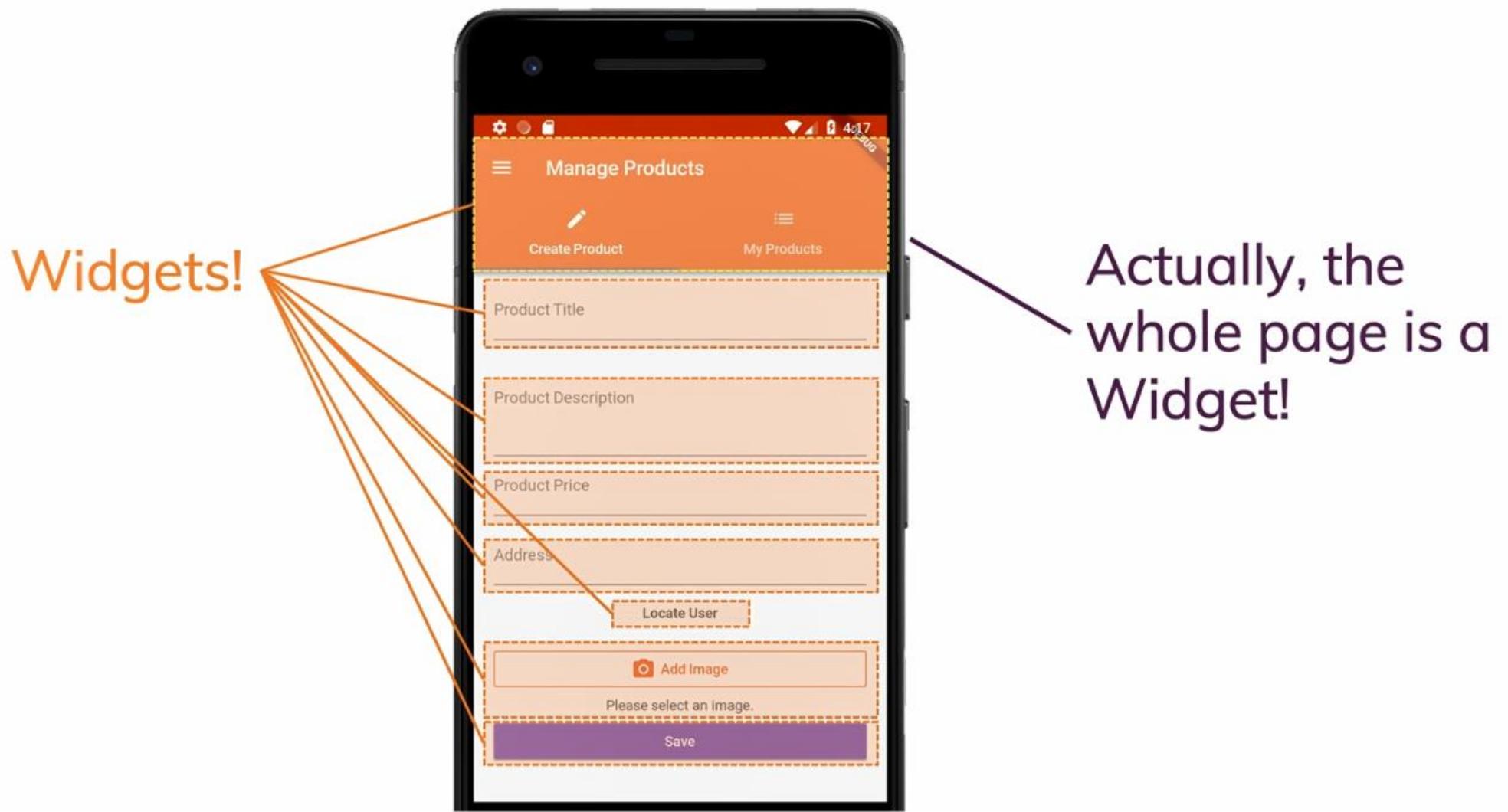


Embrace Platform Differences

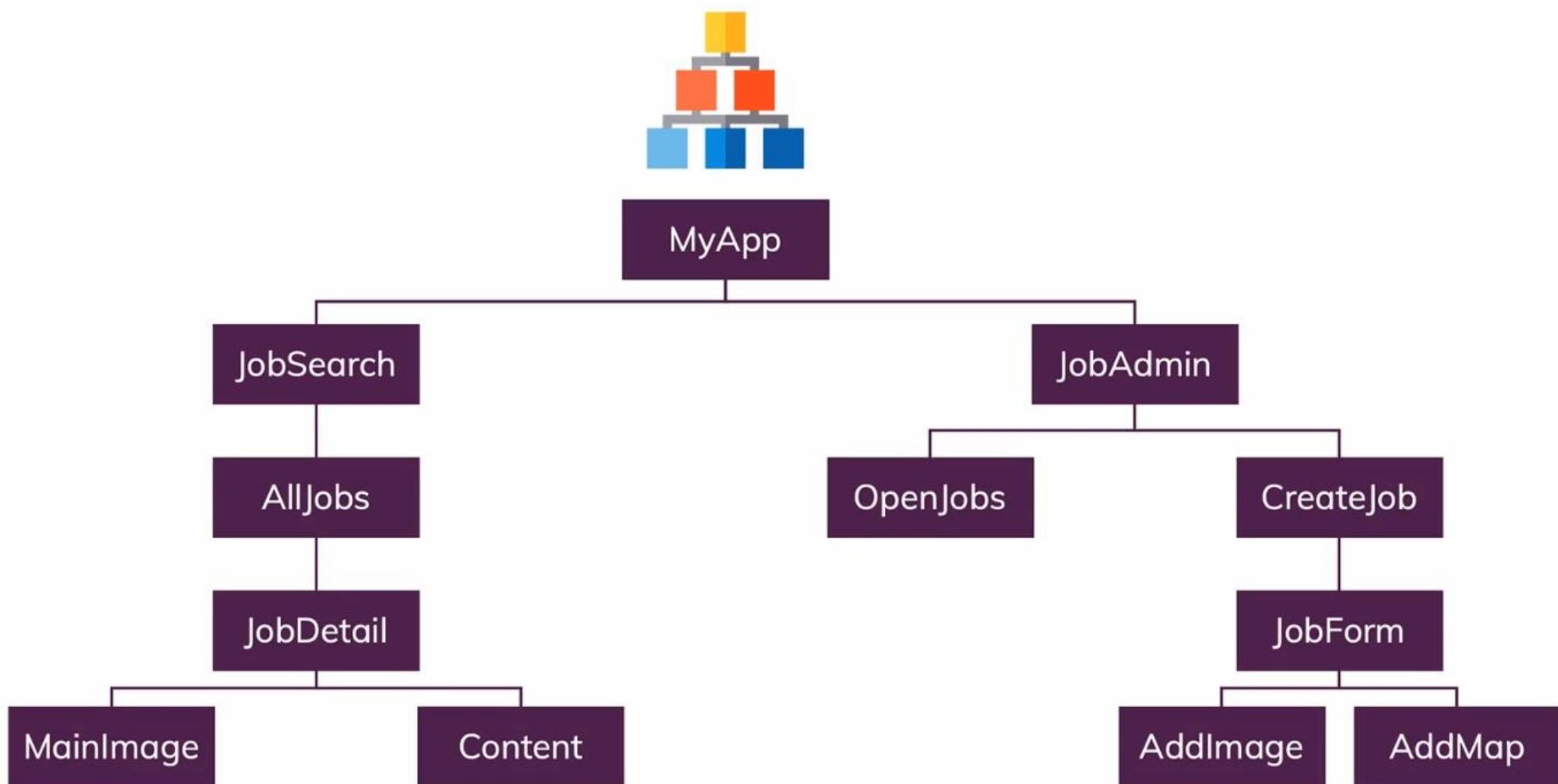


One Codebase

Everything's a Widget!



Your App's UI is a Widget Tree!



“UI as Code”

No Drag & Drop

No Visual Editor

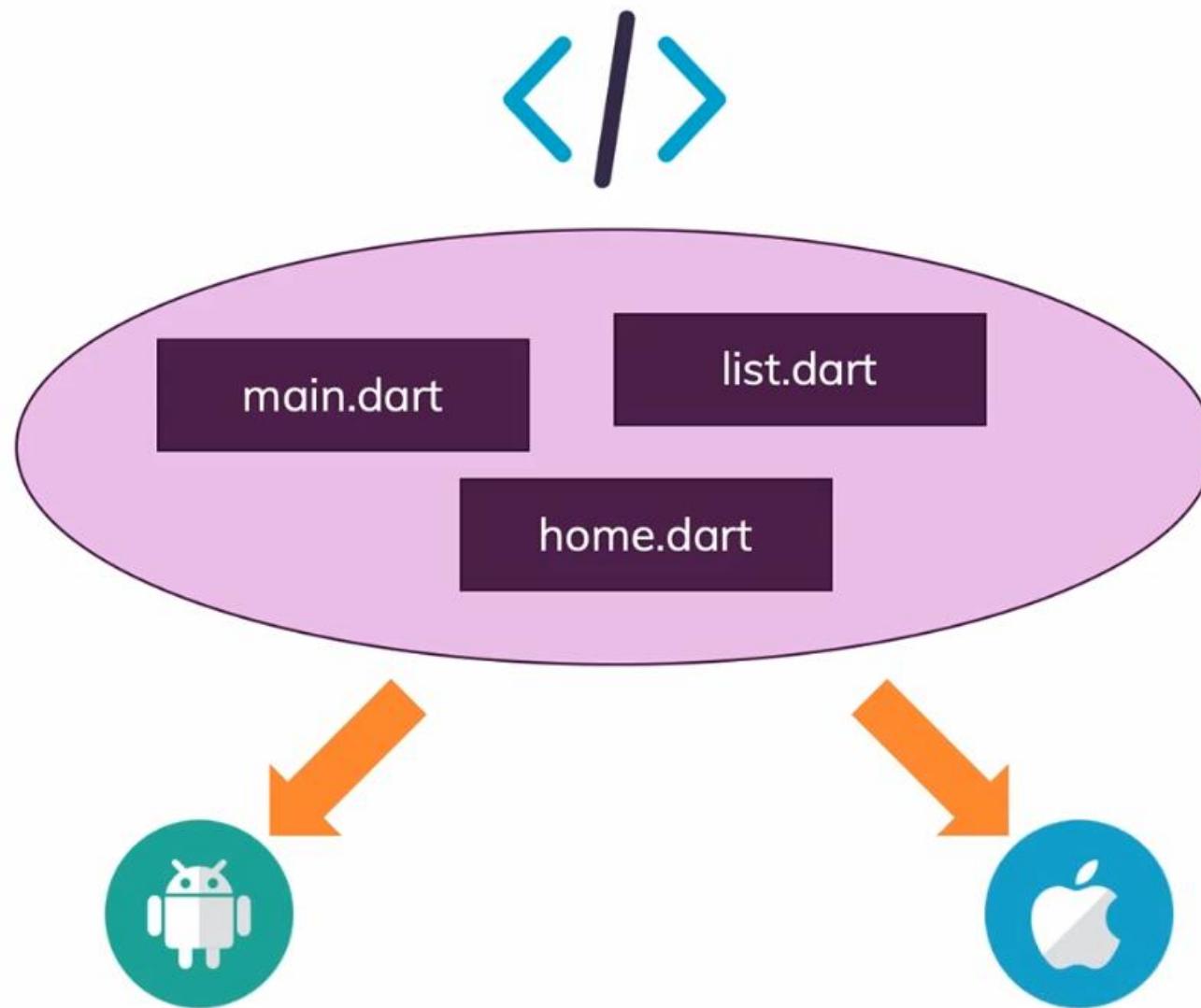
Code only

But extremely
straightforward

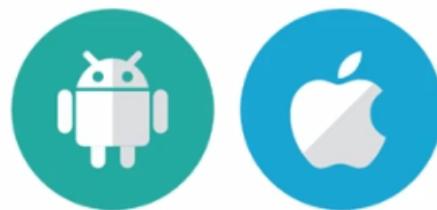


```
body: Stack(  
    children: <Widget>[  
        Container(  
            decoration: BoxDecoration(  
                image: DecorationImage(  
                    image: AssetImage('assets/images/store.jpg'),  
                    fit: BoxFit.cover,  
                    alignment: Alignment.center,  
                ), // DecorationImage  
            ), // BoxDecoration  
        ), // Container  
        Container(  
            // width: double.infinity,  
            // height: double.infinity,  
            decoration: BoxDecoration(  
                gradient: LinearGradient(  
                    colors: [  
                        Color.fromRGBO(215, 117, 255, 1).withOpacity(0.5),  
                        Color.fromRGBO(255, 188, 117, 1).withOpacity(0.9),  
                    ],  
                    begin: Alignment.topLeft,  
                    end: Alignment.bottomRight,  
                    stops: [0, 1],  
                ), // LinearGradient  
            ), // BoxDecoration  
        ), // Container  
        SingleChildScrollView(  
    ]  
)
```

One Codebase



Embrace Platform Differences



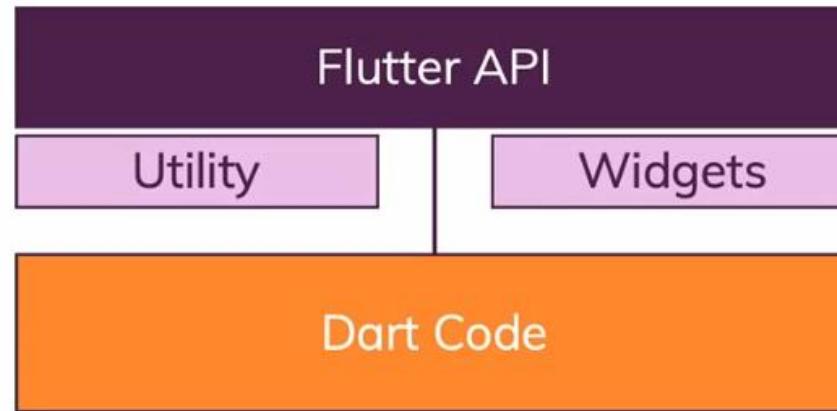
```
child: Theme.of(context).platform == TargetPlatform.iOS  
    ? RaisedButton(...) // Material (Google) look & feel  
    : CupertinoButton(...) // Apple look & feel
```

HOW FLUTTER AND DART CODES GET COMPILED TO NATIVE APPS

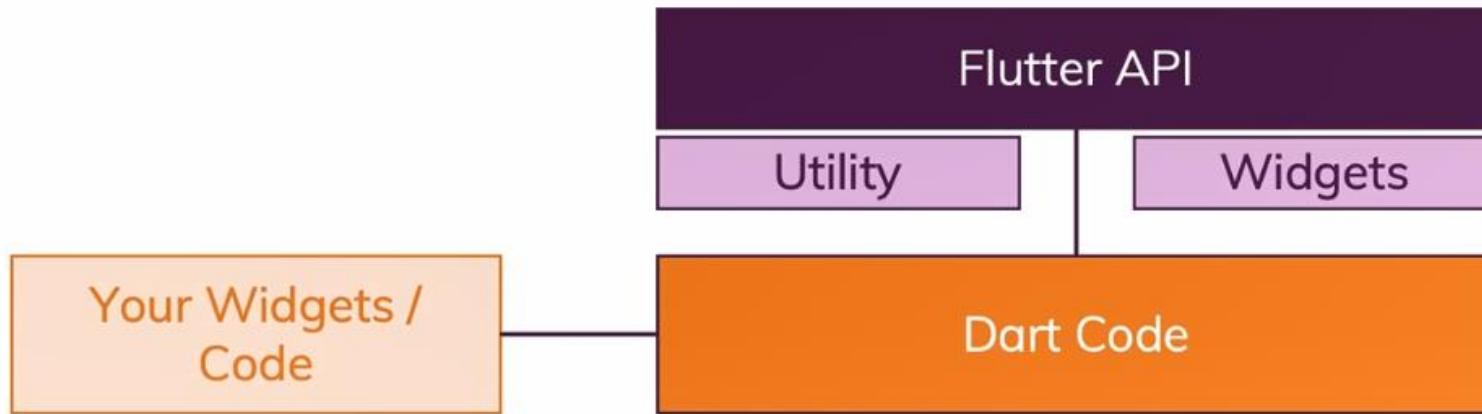
How is Flutter/ Dart “transformed” to a Native App?

Dart Code

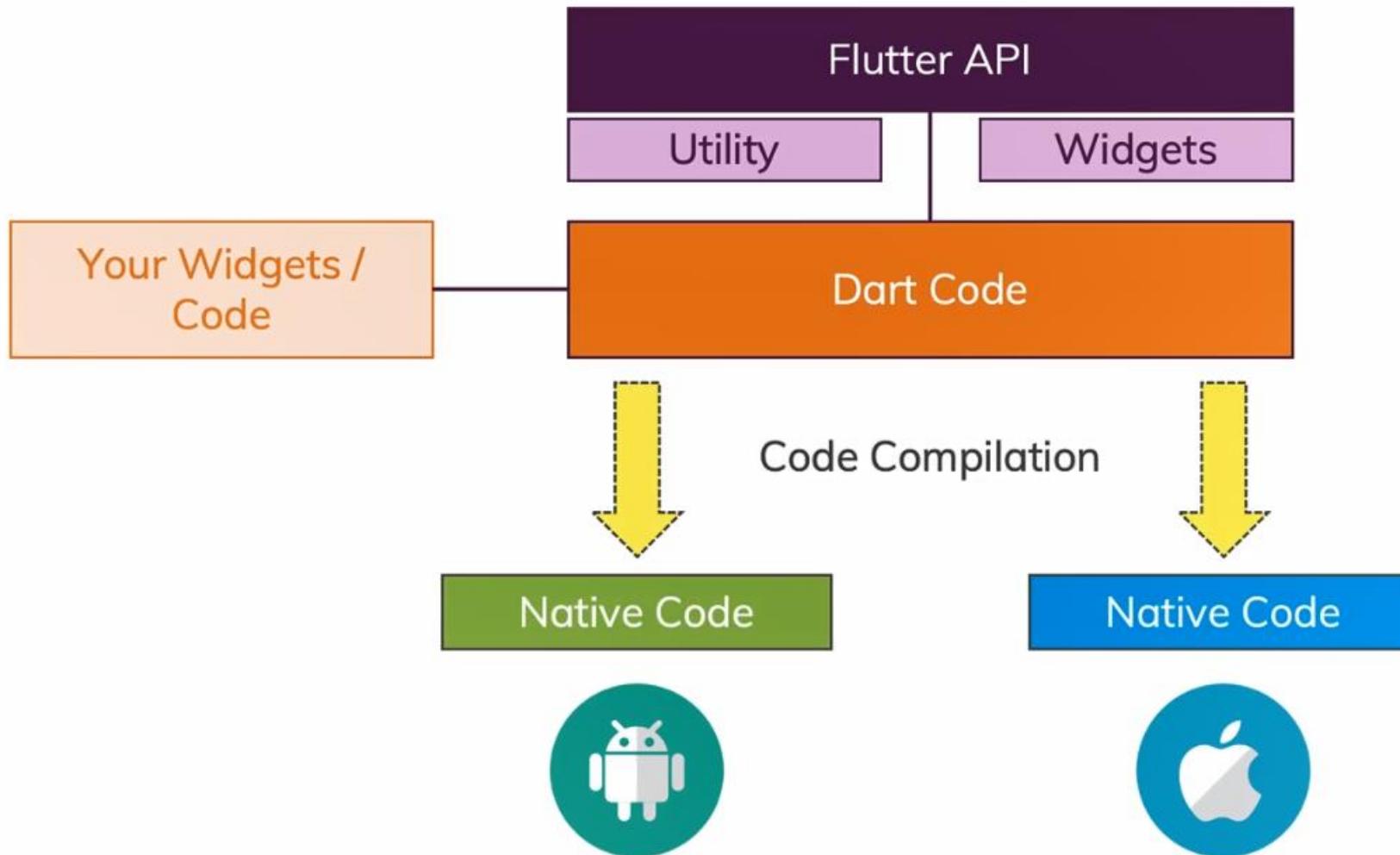
How is Flutter/ Dart “transformed” to a Native App?



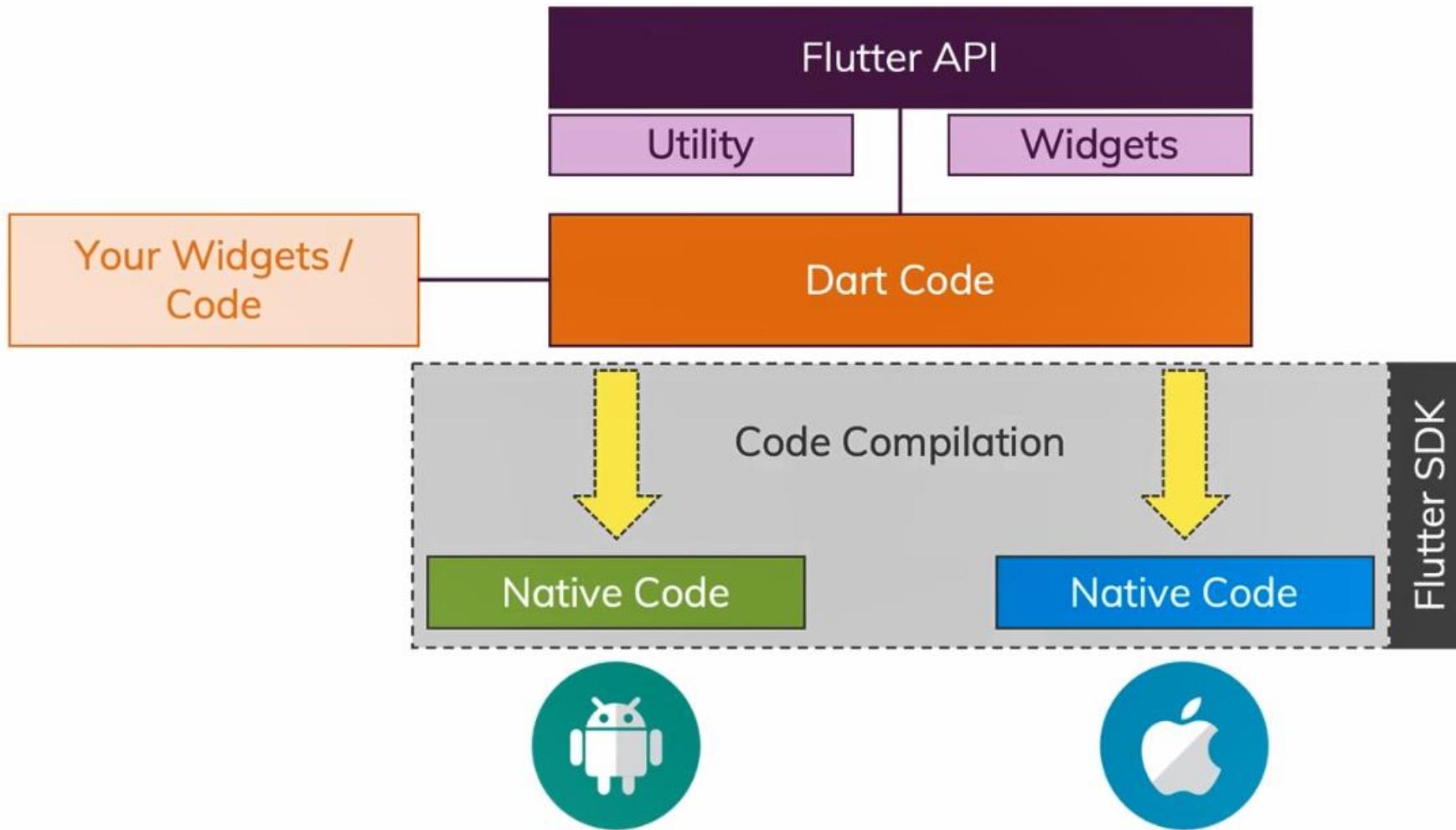
How is Flutter/ Dart “transformed” to a Native App?



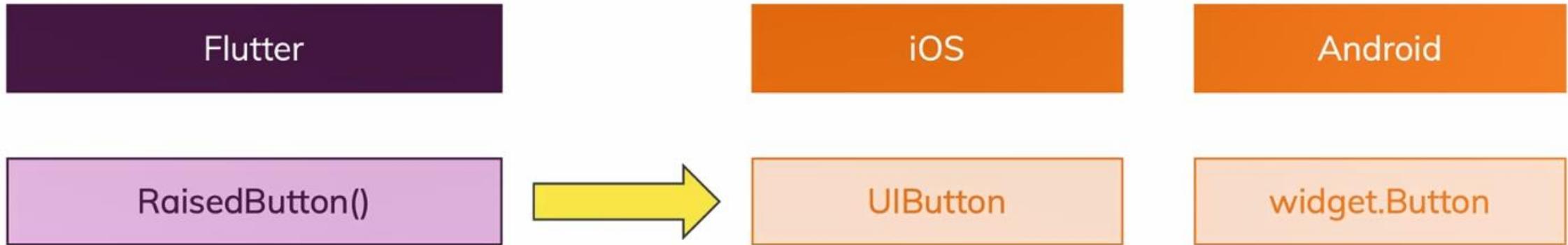
How is Flutter/ Dart “transformed” to a Native App?



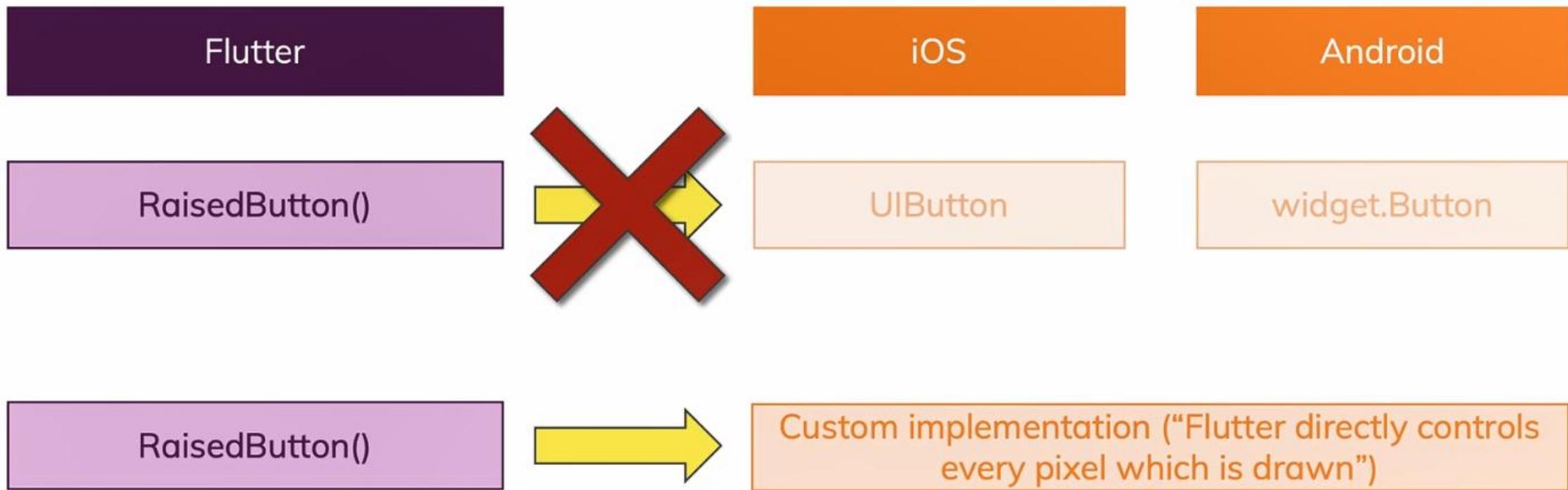
How is Flutter/ Dart “transformed” to a Native App?



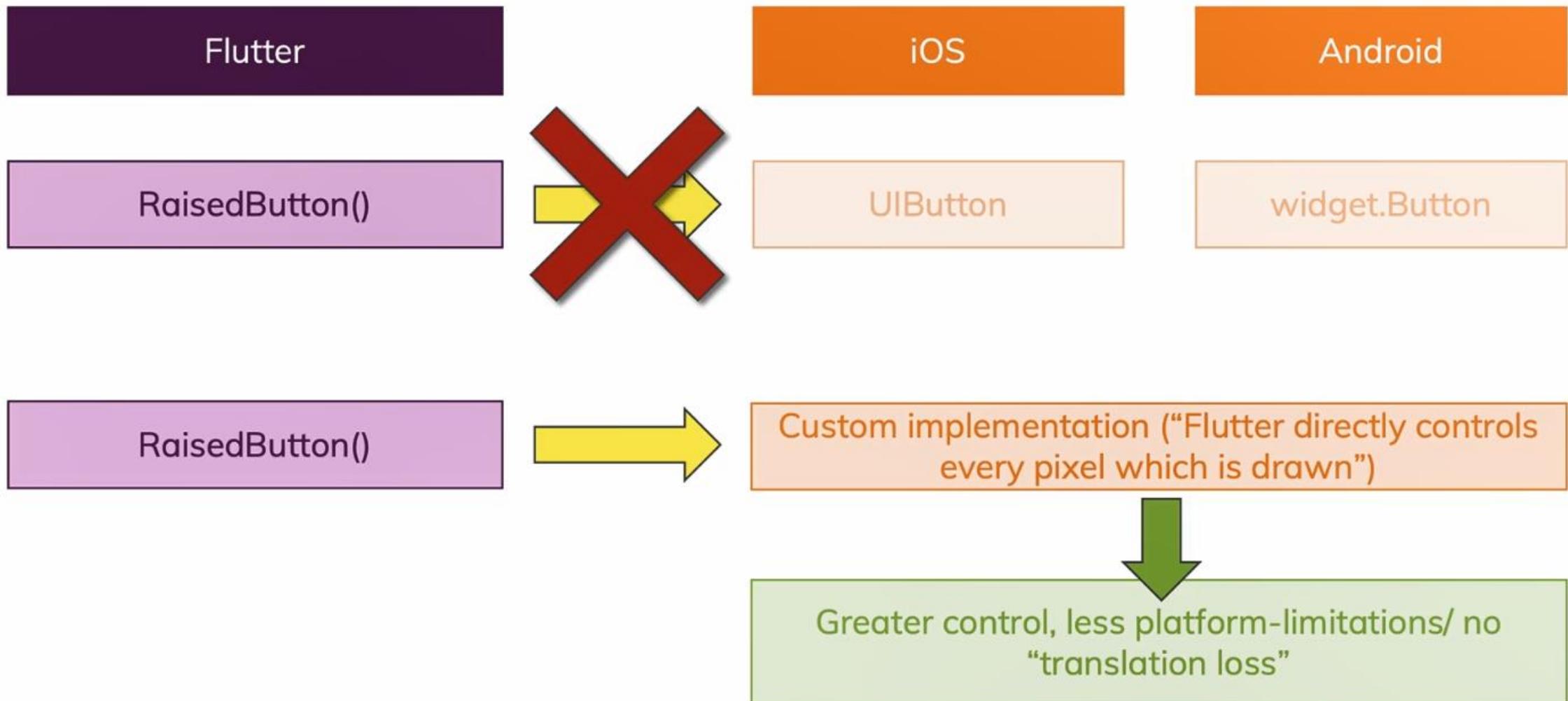
Flutter Does NOT Use Platform Primitives



Flutter Does NOT Use Platform Primitives



Flutter Does NOT Use Platform Primitives



UNDERSTANDING FLUTTER VERSIONS

Flutter Versions

Flutter Versions change frequently



This does NOT mean, that Flutter changes all
the time



Flutter is an **open-source project**. You can view all the code + development progress on Github!

review code, manage projects, and build software together.

[Sign up](#)

Flutter makes it easy and fast to build beautiful mobile apps. <https://flutter.dev>

mobile android ios material-design dart

14,682 commits

10 branches

183 releases

403 contributors

[View license](#)

Branch: master ▾

New pull request

Find File

Clone or download ▾

	engine-flutter-autoroll Roll engine 0f2ab22abb2c..4444c9cb6c9f (2 commits) (#34970)	...	Latest commit 39597ce 2 hours ago
	.github Updating dart.dev related links (#32641)		last month
	bin Roll engine 0f2ab22abb2c..4444c9cb6c9f (2 commits) (#34970)		2 hours ago
	dev Shard gradle tests (#34857)		3 days ago
	examples Update Xcode projects to recommended Xcode 10 project settings (#34738)		4 days ago
	packages Remove flutter_tools support for old AOT snapshotting (#34895)		yesterday
	.cirrus.yml Shard gradle tests (#34857)		3 days ago
	.gitattributes Add git config option to automatically convert CRLF to LF (#8122)		2 years ago
	.gitignore initial work on coverage generating script for tool (#29494)		2 months ago
	AUTHORS added `scrimColor` property in Scaffold widget (#31025)		2 months ago
	CODE_OF_CONDUCT.md Fix spelling typo (#25514)		6 months ago

Flutter Versions

Flutter Versions change frequently

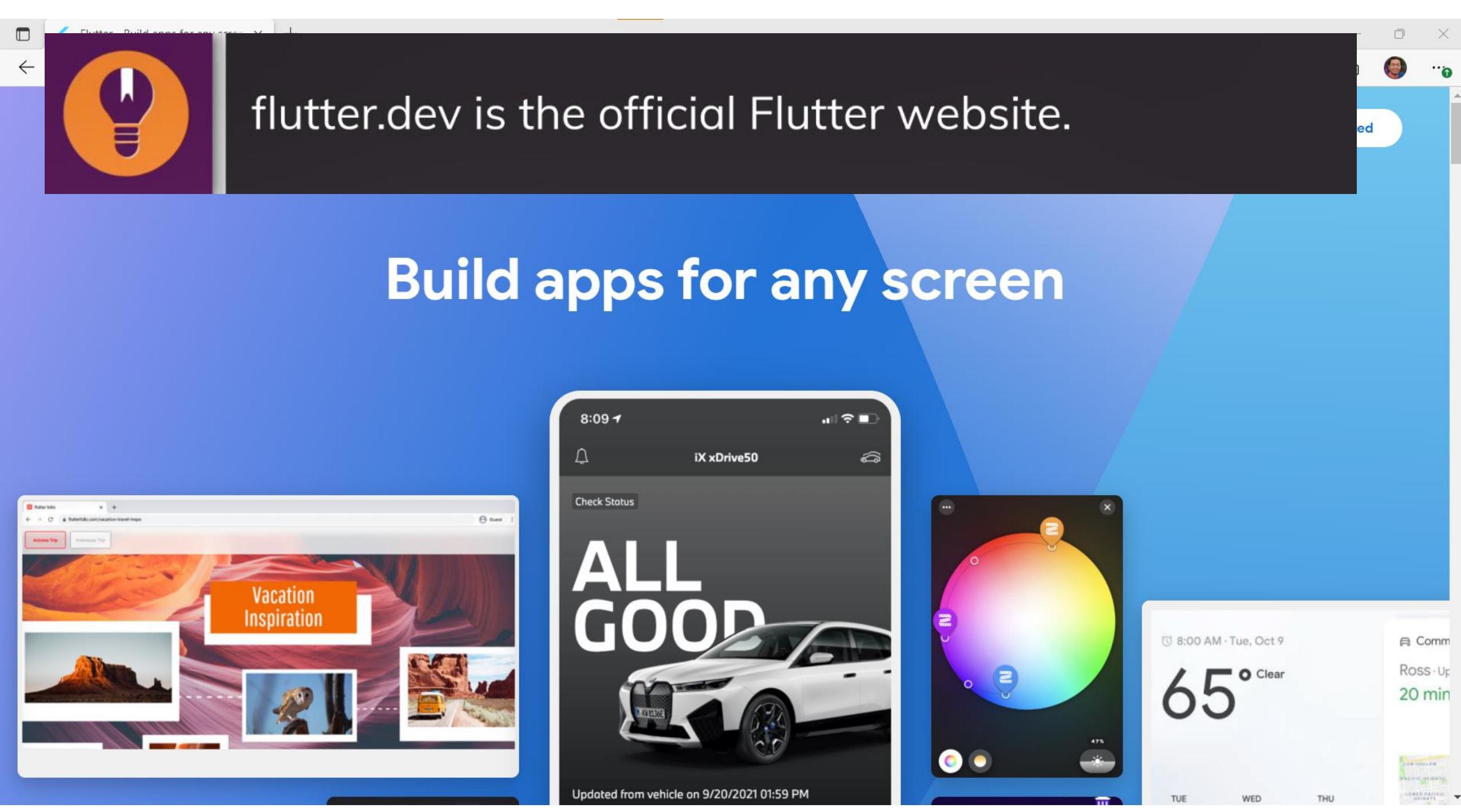


This does NOT mean, that Flutter changes all
the time



Bugfixes, small improvements, “behind-the-
scenes” changes, new (niche) features, ...

FLUTTER MACOS SETUP



Install | Flutter

https://docs.flutter.dev/get-started/install

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Free access to best-selling book Flutter Apprentice, for a limited time only! [Learn more...](#)

Get started

1. Install

2. Set up an editor

3. Test drive

4. Write your first app

5. Learn more

From another platform?

Flutter for Android devs

Flutter for iOS devs

Flutter for React Native devs

Flutter for web devs

Flutter for Xamarin.Forms devs

Introduction to declarative UI

Dart language overview ↗

Building a web app

Samples & tutorials

Set up an editor)

Install

Get started > Install

Select the operating system on which you are installing Flutter:

Windows macOS Linux Chrome OS

Important: If you're in China, first read [Using Flutter in China](#).

Set up an editor)

macOS install | Flutter

https://docs.flutter.dev/get-started/install/macos

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Free access to best-selling book *Flutter Apprentice*, for a limited time only! [Learn more...](#)

Get started

- 1. [Install](#)
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs

Introduction to declarative UI

Dart language overview ↗

Building a web app

Samples & tutorials

Set up an editor ↗

Contents

[System requirements](#)

[Get the Flutter SDK](#)

[Run flutter doctor](#)

[Downloading straight from GitHub instead of using an archive](#)

[Update your path](#)

[Platform setup](#)

[iOS setup](#)

[Install Xcode](#)

[Set up the iOS simulator](#)

[Create and run a simple Flutter app](#)

[Deploy to iOS devices](#)

[Android setup](#)

[Install Android Studio](#)

[Set up your Android device](#)

[Set up the Android emulator](#)

[Agree to Android Licenses](#)

macOS install

Get started > Install > macOS

System requirements

To install and run Flutter, your development environment must meet these minimum requirements:

- **Operating Systems:** macOS
- **Disk Space:** 2.8 GB (does not include disk space for IDE/tools).
- **Tools:** Flutter uses [git](#) for installation and upgrade. We recommend installing [Xcode](#), which includes [git](#), but you can also [install git separately](#).

Important: If you're installing on a Mac with the latest [Apple M1 processor](#), you may find [these supplementary notes](#) useful reading as we complete support for the new Apple Silicon architecture.



git --distributed-is-the-new-centralized

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient staging areas, and **multiple workflows**.



About

The advantages of Git compared to other source control systems.



Documentation

Command reference pages, Pro Git book content, videos and other material.



Downloads

GUI clients and binary releases for all major platforms.



Community

Get involved! Bug reporting, mailing list, chat, development and more.



Pro Git by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).



Latest source Release

2.34.0

[Release Notes \(2021-11-15\)](#)

[Download for Windows](#)

[Windows GUIs](#)

[Tarballs](#)

[Mac Build](#)

[Source Code](#)

 git --distributed-even-if-your-workflow-isnt

[About](#)

[Documentation](#)

[Downloads](#)

- GUI Clients
- Logos

[Community](#)

The entire [Pro Git book](#) written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Download for macOS

There are several options for installing Git on macOS. Note that any non-source distributions are provided by third parties, and may not be up to date with the latest source release.

Homebrew

Install [homebrew](#) if you don't already have it, then:

```
$ brew install git
```

Xcode

Apple ships a binary package of Git with [Xcode](#).

Binary installer

Tim Harper provides an [installer](#) for Git. The latest version is [2.33.0](#), which was released 3 months ago, on 2021-08-30.

Building from Source

If you prefer to build from source, you can find tarballs on [kernel.org](#). The latest version is [2.34.0](#).

Installing git-gui

If you would like to install [git-gui](#) and [gitk](#), git's commit GUI and interactive history browser, you can do so using [homebrew](#)

```
$ brew install git-gui
```



Search bar: xcode



Results for "xcode"

Filters ▾

Discover

Arcade

Create

Work

Play

Develop

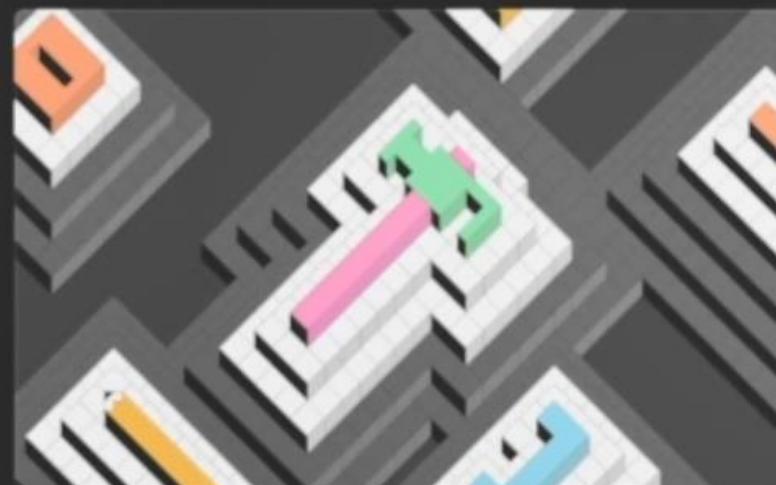
Categories

Updates

STORY

What's Xcode?

This suite of tools gives devs serious superpowers.



Xcode

Developer Tools

★ ★ ★ ★ 2.1K

OPEN

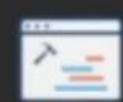


DevCleaner for Xcode

Free your disk space

GET

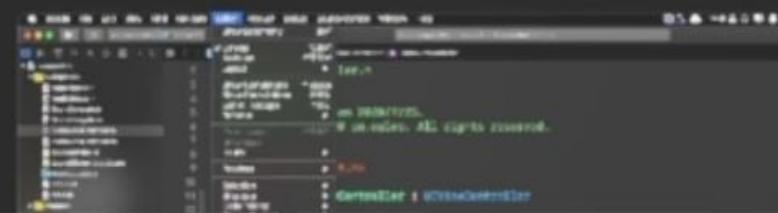
In-App Purchases



Tools for Xcode

Developer Tools

GET



macOS install | Flutter Git - Downloading Package

https://docs.flutter.dev/get-started/install/macos

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

- 1. Install
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

User interface

Get the Flutter SDK

1. Download the following installation bundle to get the latest stable release of the Flutter SDK:

[flutter_macos_2.5.3-stable.zip](#)

For other release channels, and older builds, see the [SDK releases page](#).

2. Extract the file in the desired location, for example:

```
$ cd ~/development  
$ unzip ~/Downloads/flutter_macos_2.5.3-stable.zip
```

3. Add the `flutter` tool to your path:

```
$ export PATH="$PATH:`pwd`/flutter/bin"
```

This command sets your `PATH` variable for the *current* terminal window only. To permanently add Flutter to your path, see [Update your path](#).

You are now ready to run Flutter commands!

Note: To update an existing version of Flutter, see [Upgrading Flutter](#).

Contents

System requirements

Get the Flutter SDK

Run flutter doctor

Downloading straight from GitHub instead of using an archive

Update your path

Platform setup

iOS setup

Install Xcode

Set up the iOS simulator

Create and run a simple Flutter app

Deploy to iOS devices

Android setup

Install Android Studio

Set up your Android device

Set up the Android emulator

Agree to Android Licenses

macOS setup

Additional macOS requirements

macOS install | Flutter Windows install | Flutter Git - Downloading Package

https://docs.flutter.dev/get-started/install/macos#update-your-path

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

- 1. Install
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

User interface

Update your path

You can update your PATH variable for the current session at the command line, as shown in [Get the Flutter SDK](#). You'll probably want to update this variable permanently, so you can run `flutter` commands in any terminal session.

The steps for modifying this variable permanently for all terminal sessions are machine-specific. Typically you add a line to a file that is executed whenever you open a new window. For example:

1. Determine the path of your clone of the Flutter SDK. You need this in Step 3.
2. Open (or create) the `rc` file for your shell. Typing `echo $SHELL` in your Terminal tells you which shell you're using. If you're using Bash, edit `$HOME/.bash_profile` or `$HOME/.bashrc`. If you're using Z shell, edit `$HOME/.zshrc`. If you're using a different shell, the file path and filename will be different on your machine.
3. Add the following line and change `[PATH_OF_FLUTTER_GIT_DIRECTORY]` to be the path of your clone of the Flutter git repo:

```
$ export PATH="$PATH:[PATH_OF_FLUTTER_GIT_DIRECTORY]/bin"
```
4. Run `source $HOME/.<rc file>` to refresh the current window, or open a new terminal window to automatically source the file.
5. Verify that the `flutter/bin` directory is now in your PATH by running:

```
$ echo $PATH
```

Verify that the `flutter` command is available by running:

```
$ which flutter
```

Downloading straight from GitHub instead of using an archive
Update your path

Platform setup

iOS setup

- Install Xcode
- Set up the iOS simulator
- Create and run a simple Flutter app
- Deploy to iOS devices

Android setup

- Install Android Studio
- Set up your Android device
- Set up the Android emulator
- Agree to Android Licenses

macOS setup

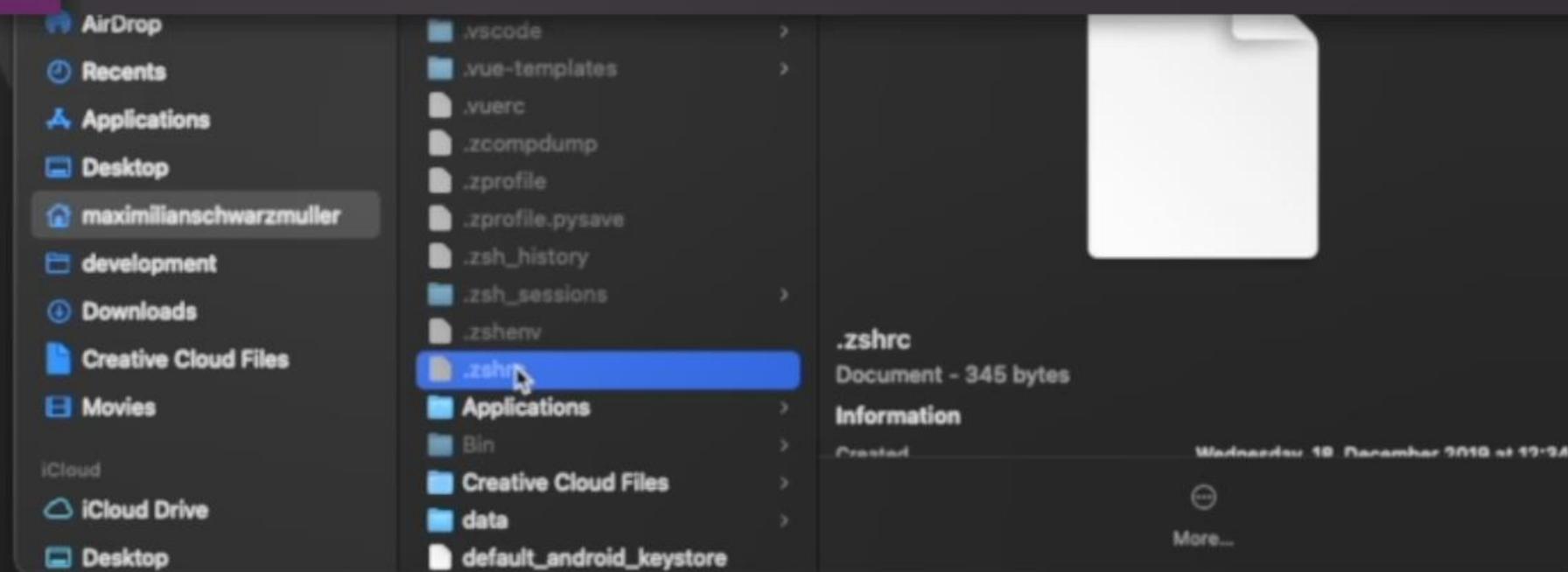
- Additional macOS requirements
- Enable desktop support

Web setup

Next step



If you DON'T have that file, you can simply create it. Create a new, empty text file and save it as ".zshrc" (no file extension). Also make sure you're displaying hidden files, in case you don't see it!



macOS install | Flutter Windows install | Flutter Git - Downloading Package

https://docs.flutter.dev/get-started/install/macos#update-your-path

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

- 1. Install
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

User interface

Install Xcode

To develop Flutter apps for iOS, you need a Mac with Xcode installed.

1. Install the latest stable version of Xcode (using [web download](#) or the [Mac App Store](#)).
2. Configure the Xcode command-line tools to use the newly-installed version of Xcode by running the following from the command line:

```
$ sudo xcode-select --switch /Applications/Xcode.app/Contents/Developer  
$ sudo xcodebuild -runFirstLaunch
```

This is the correct path for most cases, when you want to use the latest version of Xcode. If you need to use a different version, specify that path instead.

3. Make sure the Xcode license agreement is signed by either opening Xcode once and confirming or running `sudo xcodebuild -license` from the command line.

Versions older than the latest stable version may still work, but are not recommended for Flutter development. Using old versions of Xcode to target bitcode is not supported, and is likely not to work.

With Xcode, you'll be able to run Flutter apps on an iOS device or on the simulator.

Set up the iOS simulator

To prepare to run and test your Flutter app on the iOS simulator, follow these steps:

1. On your Mac, find the Simulator via Spotlight or by using the following command:

Downloading straight from GitHub instead of using an archive
Update your path

Platform setup

iOS setup

- Install Xcode
- Set up the iOS simulator
- Create and run a simple Flutter app
- Deploy to iOS devices

Android setup

- Install Android Studio
- Set up your Android device
- Set up the Android emulator
- Agree to Android Licenses

macOS setup

- Additional macOS requirements
- Enable desktop support

Web setup

Next step

macOS install | Flutter Windows install | Flutter Git - Downloading Package

https://docs.flutter.dev/get-started/install/macos#update-your-path

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

- 1. Install
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

- User interface

Set up the iOS simulator

To prepare to run and test your Flutter app on the iOS simulator, follow these steps:

1. On your Mac, find the Simulator via Spotlight or by using the following command:

```
$ open -a Simulator
```
2. Make sure your simulator is using a 64-bit device (iPhone 5s or later) by checking the settings in the simulator's **Hardware > Device** menu.
3. Depending on your development machine's screen size, simulated high-screen-density iOS devices might overflow your screen. Grab the corner of the simulator and drag it to change the scale. You can also use the **Window > Physical Size** or **Window > Pixel Accurate** options if your computer's resolution is high enough.
 - If you are using a version of Xcode older than 9.1, you should instead set the device scale in the **Window > Scale** menu.

Create and run a simple Flutter app

To create your first Flutter app and test your setup, follow these steps:

1. Create a new Flutter app by running the following from the command line:

```
$ flutter create my_app
```
2. A `my_app` directory is created, containing Flutter's starter app. Enter this directory:

Downloading straight from GitHub instead of using an archive
Update your path

Platform setup

iOS setup

- Install Xcode
- Set up the iOS simulator
- Create and run a simple Flutter app
- Deploy to iOS devices

Android setup

- Install Android Studio
- Set up your Android device
- Set up the Android emulator
- Agree to Android Licenses

macOS setup

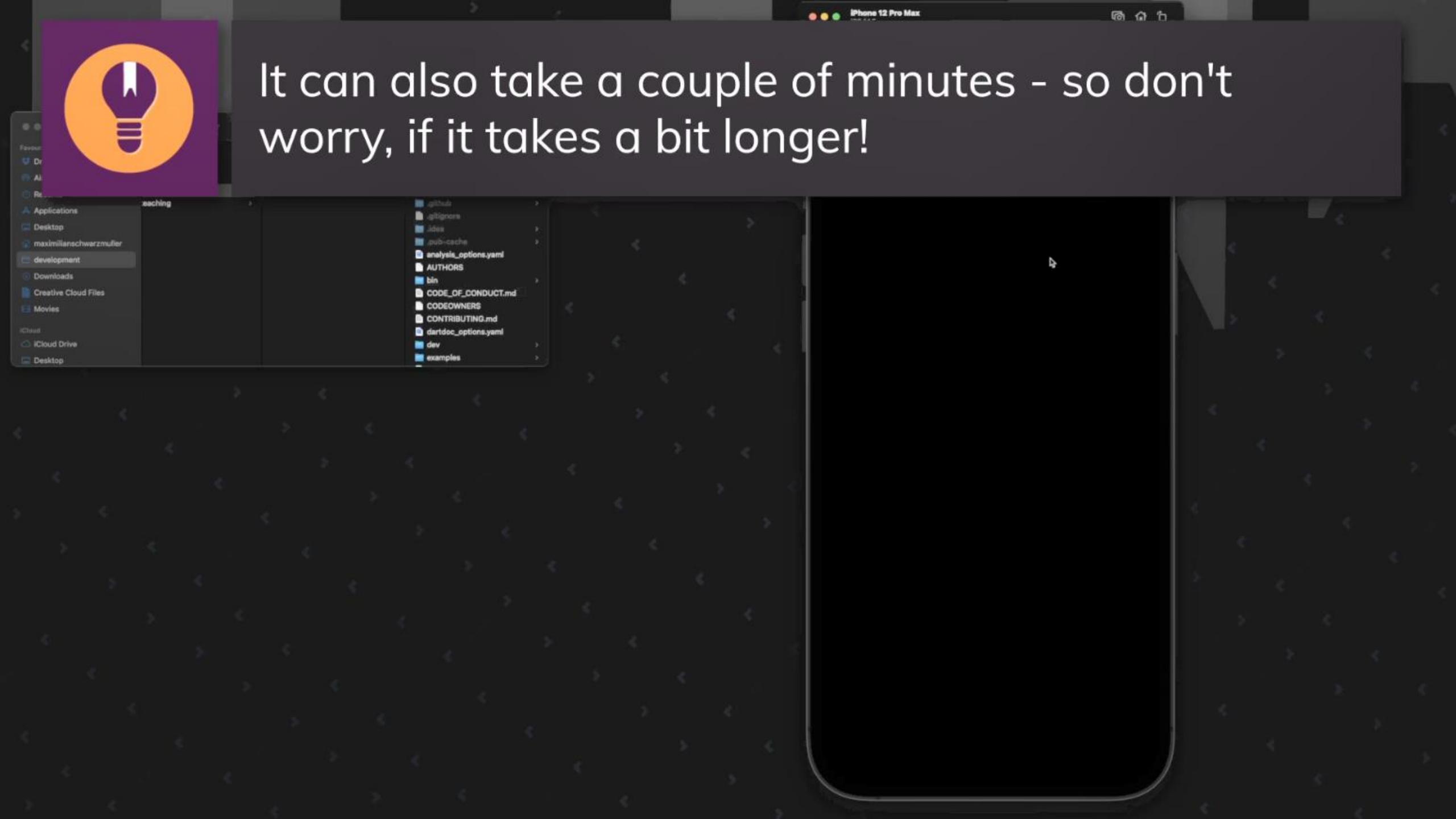
- Additional macOS requirements
- Enable desktop support

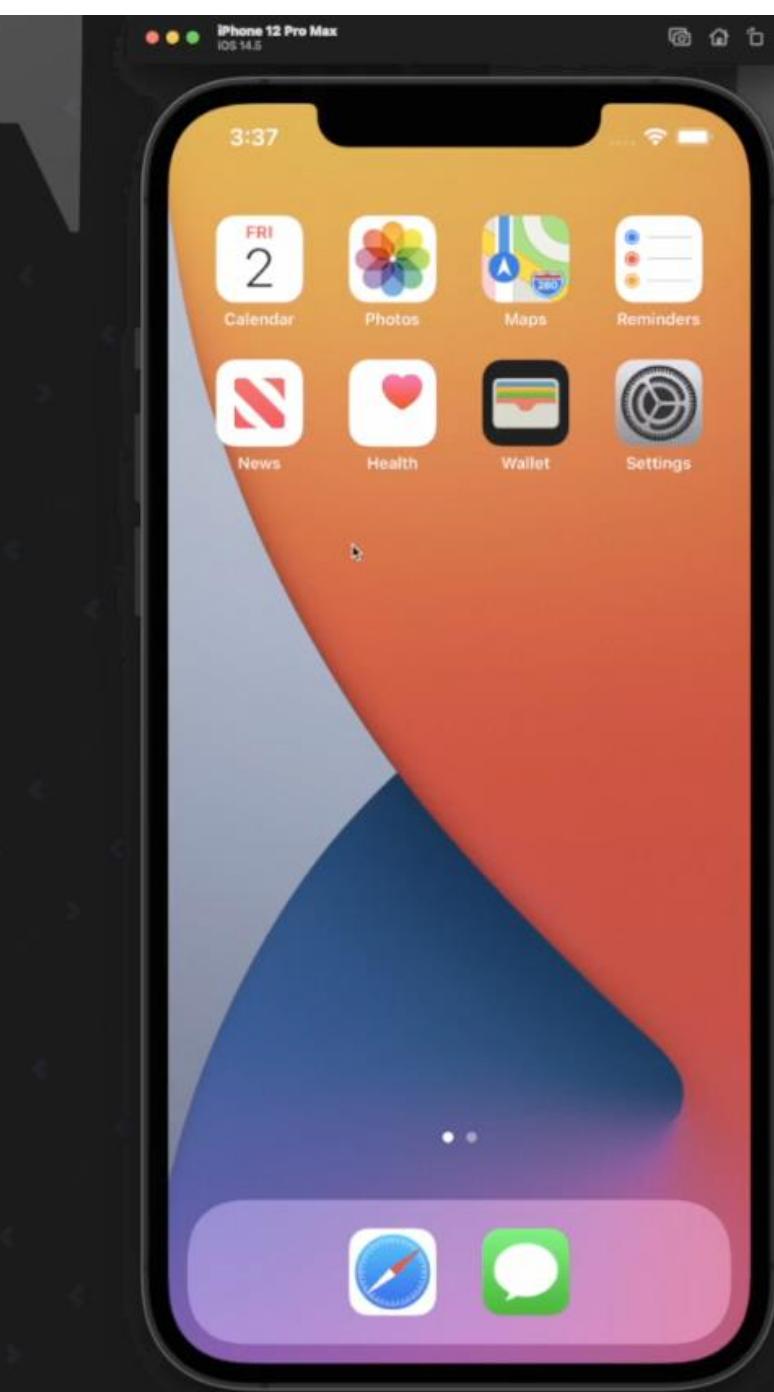
Web setup

Next step



It can also take a couple of minutes - so don't worry, if it takes a bit longer!





macOS install | Flutter Windows install | Flutter Git - Downloading Package

https://docs.flutter.dev/get-started/install/macos#update-your-path

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

1. [Install](#)
2. Set up an editor
3. Test drive
4. Write your first app
5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs

Introduction to declarative UI

Dart language overview ↗

Building a web app

Samples & tutorials

Development

User interface

Android setup

Note: Flutter relies on a full installation of Android Studio to supply its Android platform dependencies. However, you can write your Flutter apps in a number of editors; a later step discusses that.

Install Android Studio

1. Download and install [Android Studio](#).
2. Start Android Studio, and go through the 'Android Studio Setup Wizard'. This installs the latest Android SDK, Android SDK Command-line Tools, and Android SDK Build-Tools, which are required by Flutter when developing for Android.
3. Run `flutter doctor` to confirm that Flutter has located your installation of Android Studio. If Flutter cannot locate it, run `flutter config --android-studio-dir <directory>` to set the directory that Android Studio is installed to.

Set up your Android device

To prepare to run and test your Flutter app on an Android device, you need an Android device running Android 4.1 (API level 16) or higher.

1. Enable **Developer options** and **USB debugging** on your device. Detailed instructions are available in the [Android documentation](#).
2. Windows-only: Install the [Google USB Driver](#).
3. Using a USB cable, plug your phone into your computer. If prompted on your device, authorize your computer to access your device.
4. In the terminal, run the `flutter devices` command to verify that Flutter recognizes your connected Android device. By

Downloading straight from GitHub instead of using an archive
Update your path

Platform setup

iOS setup

[Install Xcode](#)
[Set up the iOS simulator](#)
[Create and run a simple Flutter app](#)
[Deploy to iOS devices](#)

Android setup

[Install Android Studio](#)
[Set up your Android device](#)
[Set up the Android emulator](#)
[Agree to Android Licenses](#)

macOS setup

[Additional macOS requirements](#)
[Enable desktop support](#)

Web setup

Next step

A screenshot of a web browser window showing the Android Studio download page. The address bar contains the URL <https://developer.android.com/studio>. The page header includes tabs for "macOS install | Flutter", "Windows install | Flutter", "Git - Downloading Package", and "Download Android Studio and S...". The main navigation menu has items like "developers" (with an Android icon), "Platform", "Android Studio" (underlined in green), "Google Play", "Jetpack", "Kotlin", "More", "Search" (with a magnifying glass icon), "English" (with a dropdown arrow), and "Sign in".

ANDROID STUDIO

[Download](#) [What's new](#) [User guide](#) [Preview](#)



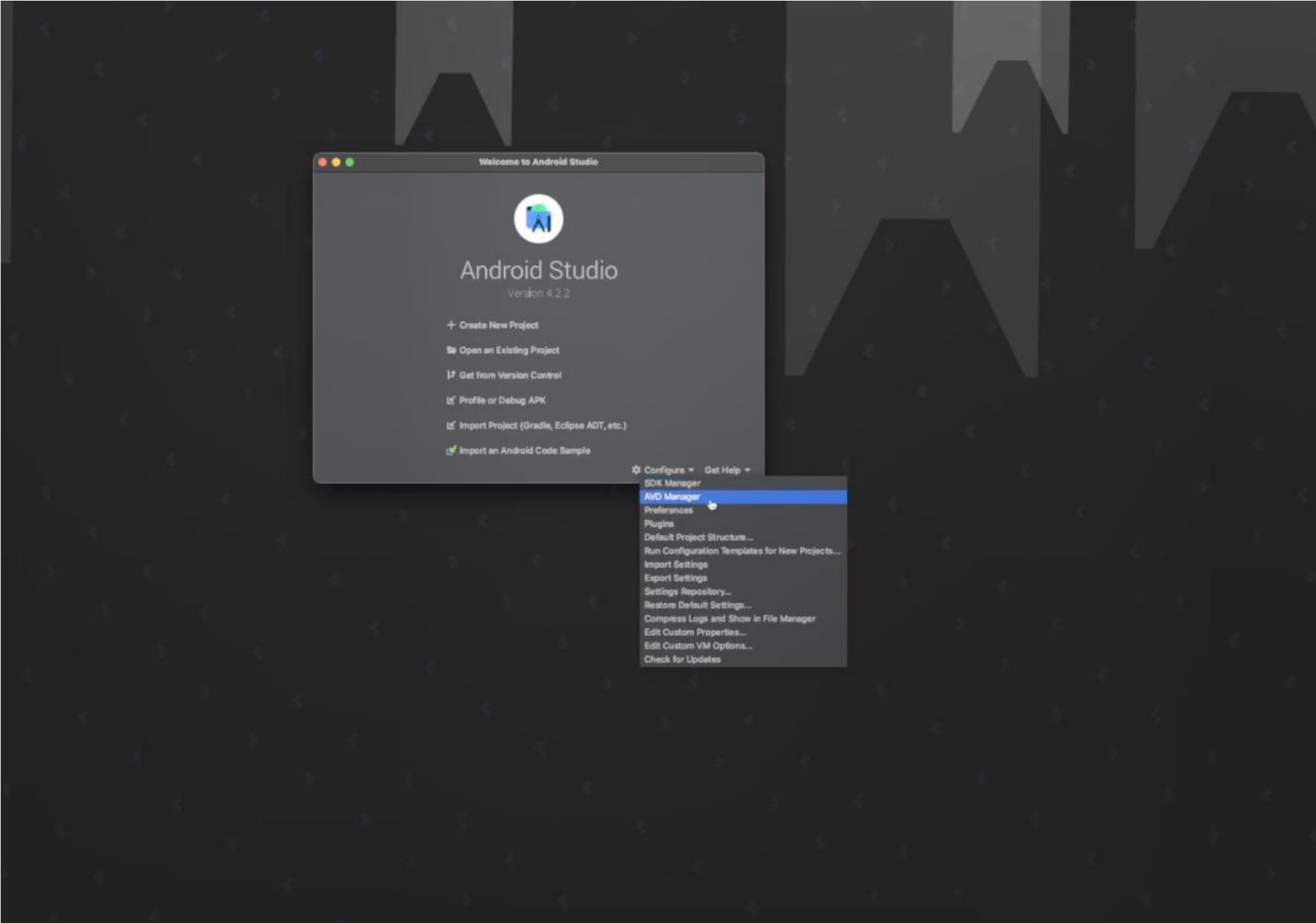
Android Studio provides the fastest tools for building apps on every type of Android device.

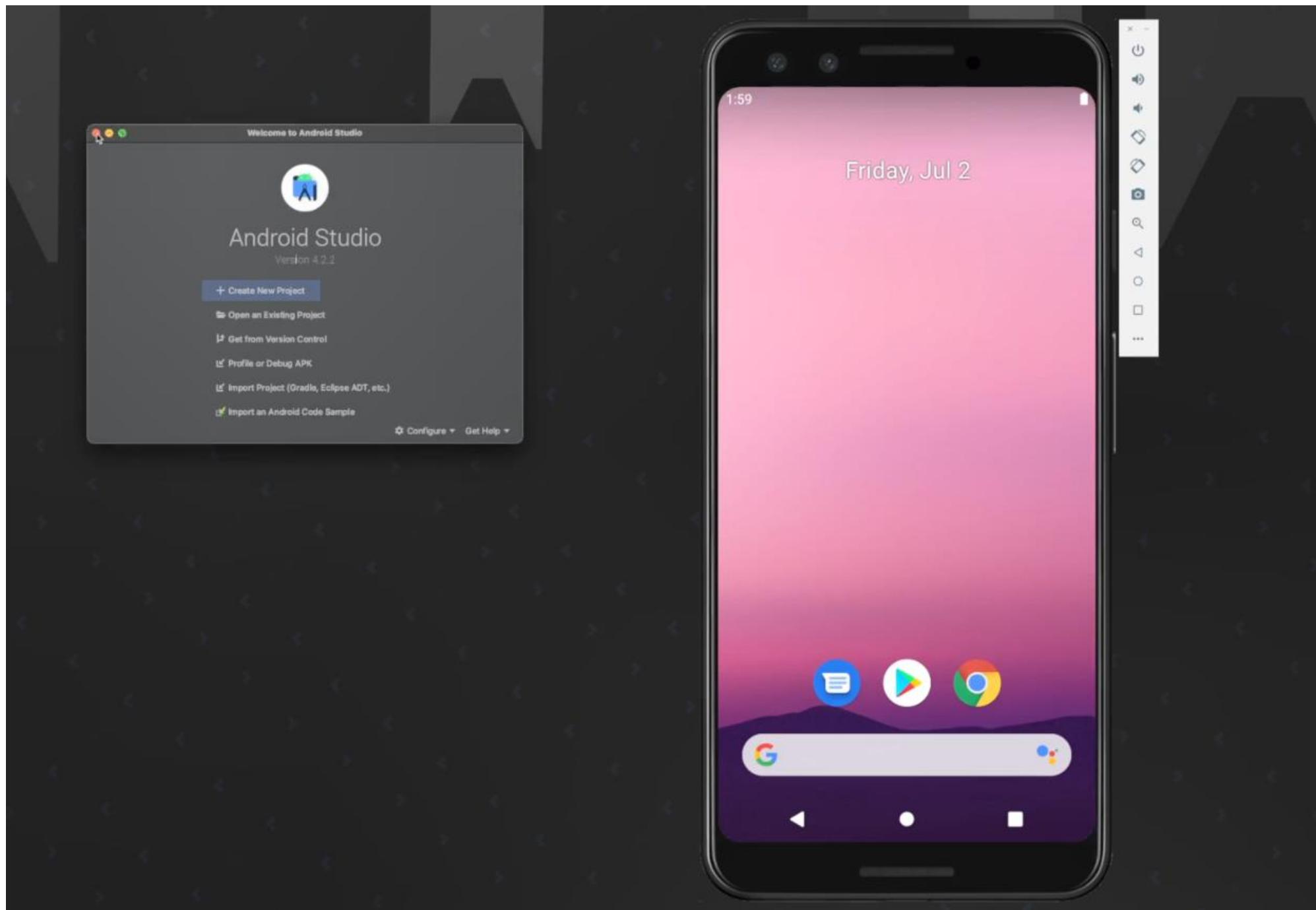
[Download Android Studio](#)

2020.3.1 for Windows 64-bit (914 MiB)

[Download options](#)

[Release notes](#)





macOS install | Flutter Windows install | Flutter Git - Downloading Package Download Android Studio and S... +

https://docs.flutter.dev/get-started/install/macos#update-your-path

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

- 1. Install
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

User interface

Agree to Android Licenses

Before you can use Flutter, you must agree to the licenses of the Android SDK platform. This step should be done after you have installed the tools listed above.

1. Make sure that you have a version of Java 8 installed and that your `JAVA_HOME` environment variable is set to the JDK's folder.
Android Studio versions 2.2 and higher come with a JDK, so this should already be done.
2. Open an elevated console window and run the following command to begin signing licenses.
`$ flutter doctor --android-licenses`
3. Review the terms of each license carefully before agreeing to them.
4. Once you are done agreeing with licenses, run `flutter doctor` again to confirm that you are ready to use Flutter.

macOS setup

⚠ Warning: Beta! This area covers desktop support, which is available as a beta release. Beta support still has notable feature gaps, including accessibility support. You can try a beta snapshot of desktop support on the stable channel, or you can keep up with the latest changes to desktop on the beta channel. For more information, see the [Desktop](#) section in [What's new in Flutter 2](#), a free article on Medium.

Contents

- System requirements
- Get the Flutter SDK
 - Run flutter doctor
 - Downloading straight from GitHub instead of using an archive
 - Update your path
- Platform setup
- iOS setup
 - Install Xcode
 - Set up the iOS simulator
 - Create and run a simple Flutter app
 - Deploy to iOS devices
- Android setup
 - Install Android Studio
 - Set up your Android device
 - Set up the Android emulator
 - Agree to Android Licenses
- macOS setup
- Additional macOS

```
Running "flutter pub get" in flutter_tools...                                6.1s
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 2.2.3, on macOS 11.4 20F71 darwin-x64, locale en-GB)
[!] Android toolchain - develop for Android devices (Android SDK version 30.0.3)
    ✘ Android license status unknown.
      Run `flutter doctor --android-licenses` to accept the SDK licenses.
      See https://flutter.dev/docs/get-started/install/macos#android-setup for more
      details.
[!] Xcode - develop for iOS and macOS
    ! CocoaPods 1.9.1 out of date (1.10.0 is recommended).
      CocoaPods is used to retrieve the iOS and macOS platform side's plugin code
      that responds to your plugin usage on the Dart side.
      Without CocoaPods, plugins will not work on iOS or macOS.
      For more info, see https://flutter.dev/platform-plugins
      To upgrade see
      https://guides.cocoapods.org/using/getting-started.html#installation for
      instructions.
[✓] Chrome - develop for the web
[✓] Android Studio (version 4.2)
[✓] VS Code (version 1.57.1)
[✓] Connected device (1 available)

! Doctor found issues in 2 categories.
```

Preferences for New Projects

[Reset](#)

Appearance & Behavior

[Appearance](#)[Menus and Toolbars](#)

System Settings

[Passwords](#)[HTTP Proxy](#)[Data Sharing](#)[Date Formats](#)[Updates](#)

Android SDK

[Memory Settings](#)[Notifications](#)[Quick Lists](#)[Path Variables](#)

Keymap

> Editor

[Plugins](#)

Appearance & Behavior > System Settings > Android SDK

Manager for the Android SDK and Tools used by Android Studio
Android SDK Location: /Users/maximilianschwarzmueller/Library/Android/sdk[Edit](#) [Optimize disk space](#)[SDK Platforms](#) [SDK Tools](#) [SDK Update Sites](#)

Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
<input type="checkbox"/> Android SDK Build-Tools 31-rc5		Update Available: 31.0.0 rc5
<input type="checkbox"/> NDK (Side by side)		Not Installed
<input checked="" type="checkbox"/> Android SDK Command-line Tools (latest)		Not Installed
<input type="checkbox"/> CMake		Not Installed
<input type="checkbox"/> Android Auto API Simulators	1	Not installed
<input type="checkbox"/> Android Auto Desktop Head Unit Emulator	1.1	Not installed
<input checked="" type="checkbox"/> Android Emulator	30.7.5	Installed
<input checked="" type="checkbox"/> Android SDK Platform-Tools	31.0.2	Installed
<input checked="" type="checkbox"/> Android SDK Tools	26.1.1	Installed
<input checked="" type="checkbox"/> Google Play APK Expansion library	1	Installed
<input type="checkbox"/> Google Play Instant Development SDK	1.9.0	Not installed
<input type="checkbox"/> Google Play Licensing Library	1	Not installed
<input checked="" type="checkbox"/> Google Play services	49	Installed
<input type="checkbox"/> Google Web Driver	2	Not installed
<input checked="" type="checkbox"/> Intel x86 Emulator Accelerator (HAXM installer)	7.6.5	Installed
<input type="checkbox"/> Layout Inspector image server for API 29-30	6	Not installed
<input type="checkbox"/> Layout Inspector image server for APIs	3	Not installed

 Hide Obsolete Packages Show Package Details[Cancel](#)[Apply](#)[OK](#)

```
$ flutter doctor --android-licenses
```

```
All SDK package licenses accepted.=====] 100% Computing updates...
```

```
$ flutter doctor
```

```
Doctor summary (to see all details, run flutter doctor -v):
```

```
[✓] Flutter (Channel stable, 2.2.3, on macOS 11.4 20F71 darwin-x64, locale en-GB)
[✓] Android toolchain - develop for Android devices (Android SDK version 30.0.3)
[!] Xcode - develop for iOS and macOS
```

! CocoaPods 1.9.1 out of date (1.10.0 is recommended).

CocoaPods is used to retrieve the iOS and macOS platform side's plugin code that responds to your plugin usage on the Dart side.

Without CocoaPods, plugins will not work on iOS or macOS.

For more info, see <https://flutter.dev/platform-plugins>

To upgrade see

<https://guides.cocoapods.org/using/getting-started.html#installation> for instructions.

```
[✓] Chrome - develop for the web
[✓] Android Studio (version 4.2)
[✓] VS Code (version 1.57.1)
[✓] Connected device (1 available)
```

! Doctor found issues in 1 category.

< Installation

CocoaPods is built with Ruby and it will be installable with the default Ruby available on macOS. You can use a Ruby Version manager, however we recommend that you use the standard Ruby available on macOS unless you know what you're doing.

Using the default Ruby install will require you to use `sudo` when installing gems. (This is only an issue for the duration of the gem installation, though.)

```
$ sudo gem install cocoapods
```

If you encounter any problems during installation, please visit [this](#) guide.

< Sudo-less installation

If you do *not* want to grant RubyGems admin privileges for this process, you can tell RubyGems to install into your user directory by passing either the `--user-install` flag to `gem install` or by configuring the RubyGems environment. The latter is in our opinion the best solution. To do this open up terminal and create or edit your `.bash_profile` with your preferred editor. Then enter these lines into the file:

```
$ sudo gem install cocoapods
```

Password:

```
Ignoring ffi-1.12.2 because its extensions are not built. Try: gem pristine ffi --version 1.12.2
```

```
Fetching activesupport-5.2.6.gem
```

```
Fetching public_suffix-4.0.6.gem
```

```
Fetching cocoapods-downloader-1.4.0.gem
```

```
Fetching addressable-2.8.0.gem
```

```
Fetching cocoapods-core-1.10.1.gem
```

```
Fetching nanaimo-0.3.0.gem
```

```
Fetching xcodeproj-1.19.0.gem
```

```
Fetching cocoapods-1.10.1.gem
```

```
Successfully installed activesupport-5.2.6
```

```
$ flutter doctor
```

```
Doctor summary (to see all details, run flutter doctor -v):
```

```
[✓] Flutter (Channel stable, 2.2.3, on macOS 11.4 20F71 darwin-x64, locale en-GB)
[✓] Android toolchain - develop for Android devices (Android SDK version 30.0.3)
[✓] Xcode - develop for iOS and macOS
[✓] Chrome - develop for the web
[✓] Android Studio (version 4.2)
[✓] VS Code (version 1.57.1)
[✓] Connected device (1 available)
```

- No issues found!

```
$
```

```
I
```

MACOS DEVELOPMENT ENVIRONMENT

```
playground $ flutter create first_app
```



Name is up to you but should NOT contain any
blanks (whitespace), dashes or special characters!

```
first_app/android/first_app_android.iml (created)
first_app/pubspec.yaml (created)
first_app/README.md (created)
first_app/ios/Runner/Runner-Bridging-Header.h (created)
first_app/ios/Runner/AppDelegate.swift (created)
first_app/ios/Runner.xcodeproj/project.pbxproj (created)
first_app/lib/main.dart (created)
first_app/.idea/runConfigurations/main_dart.xml (created)
first_app/.idea/libraries/Dart_SDK.xml (created)
first_app/.idea/libraries/KotlinJavaRuntime.xml (created)
first_app/.idea/modules.xml (created)
first_app/.idea/workspace.xml (created)
```

Running "flutter pub get" in first_app...

1,258ms

Wrote 78 files.

All done!

In order to run your application, type:

```
$ cd first_app
$ flutter run
```

Your application code is in first_app/lib/main.dart.

playground \$ █

macOS install | Flutter Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin... +

https://code.visualstudio.com

Visual Studio Code

Docs Updates Blog API Extensions FAQ Learn

Search Docs Download

Version 1.62 is now available! Read about the new features and fixes from October.

Code editing. Redefined.

Free. Built on open source. Runs everywhere.

Download for Windows

Stable Build

		Stable	Insiders
macOS	Universal	Download	Download
Windows x64	User Installer	Download	Download
Linux x64	.deb	Download	Download
	.rpm	Download	Download

Other downloads

The screenshot shows the Visual Studio Code application window. The code editor on the right displays a file named 'serviceWorker.js' with code related to service workers. The left sidebar shows the 'EXTENSIONS: MARKETPLACE' tab with a list of installed extensions, including Python, GitLens, C/C++, ESLint, Debugger for Chrome, Language Support, vscode-icons, Vetur, and C#. The bottom status bar shows the file path 'serviceWorker.js - create-react-app - Visual Studio Code - In...', the current branch 'master', and the terminal command '1: node'.

You can now view `create-react-app` in the browser.

Local: <http://localhost:3000/>
On Your Network: <http://10.211.55.3:3000/>

Note that the development build is not optimized.

Ln 43, Col 19 Spaces: 2 UTF-8 LF JavaScript 🔍 📡

EXTENSIONS: MARKETPLACE

flutter

- Flutter** Flutter support and debugger for Visual S... Dart Code 
- Awesome Flutter Snippets** Awesome Flutter Snippets is a collection s... Neevash Ramdial  [Install](#)
- Flutter Widget Snippets** A set of helpful widget snippets for day to... Alexis Villegas Torres  [Install](#)
- [FF] Flutter Files** Quickly scaffold flutter bloc file templates Igor Kravchenko  [Install](#)
- Flutter Tree** Extension for Flutter to build basic widget... Marcelo Velasquez  [Install](#)
- Flutter Intl** Flutter localization binding from .arb files ... Localizely  [Install](#)
- Flutter Helpers** Helper utilities for flutter projects Akshar Patel  [Install](#)
- flutter-stylizer** Flutter Stylizer organizes your Flutter class... gmlewis-vscode  [Install](#)
- Flutter Color** This plugin help you to easily visualize the... circle code solution  [Install](#)
- flutter_mobx** A extension for using MobX in Flutter Flutterando  [Install](#)
- vscode-flutter-i18n-json** VS Code extension to create a binding be... esskar  [Install](#)



Flutter

v3.28.0

Dart Code | ⚡ 3,495,681 | ★★★★★(50)

Flutter support and debugger for Visual Studio Code.

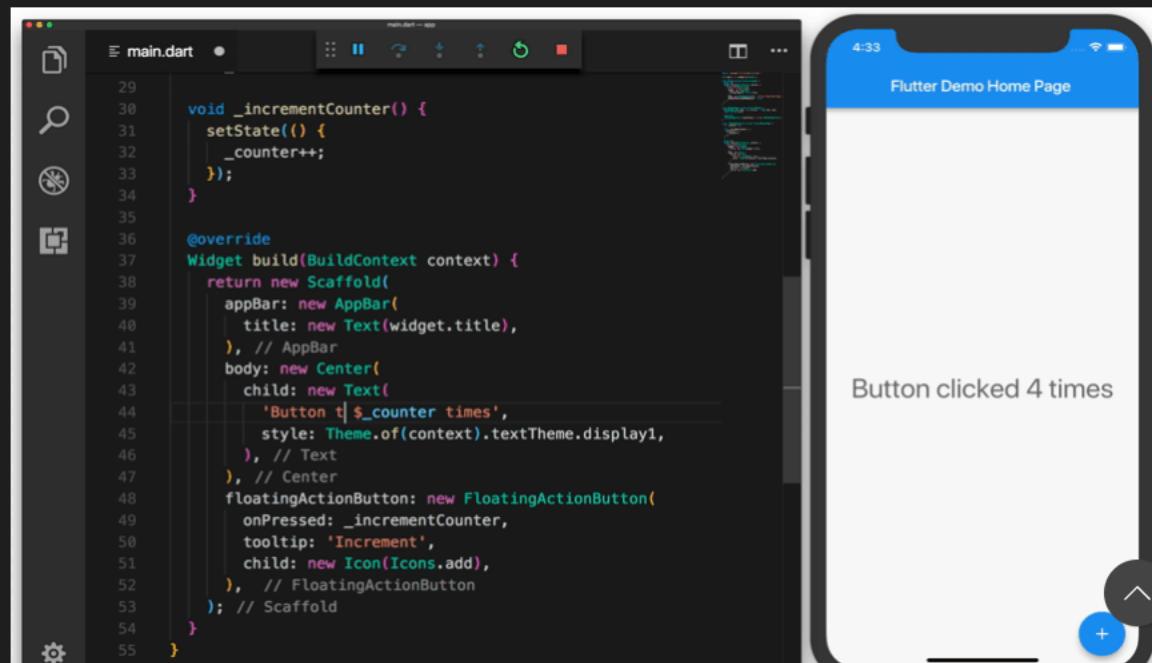
[Disable](#)[Uninstall](#)

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Dependencies](#) [Runtime Status](#)

Introduction

This [VS Code](#) extension adds support for effectively editing, refactoring, running, and reloading [Flutter](#) mobile apps, as well as support for the [Dart](#) programming language.



Categories

Programming Languages Snippets
Linters Formatters Debuggers

Resources

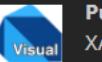
[Marketplace](#)
[Repository](#)
[License](#)

More Info

Released on 4/18/2018, 23:49:03
Last updated 11/1/2021, 23:30:15
Identifier dart-code.flutter

EXTENSIONS: MARKETPLACE

dart

- Dart** Dart language support and debugger for ... Dart Code 
- dart-import** Fix Dart/Flutter's imports Luan  
- Dart Data Class Generator** Create dart data classes easily, fast and wi... BendixMa  
- Dart (Syntax Highlighting ...** Syntax highlighting for Dart and nothing ... oscars  
- Json to Dart Model** Extension convert Json to Dart Model class hirantha  
- Flutter & Dart Utilities** Official package of Academia do Flutter (B... Rodrigo Rahman  
- Puzzle Dart** XAML for Dart Language Support bigmagician  
- Dart Getters And Setters** Dart Generate Getters And Setters Peter Haddad  
- Dart Built Value Snippets** VS Code Snippets for the Dart built_value ... YongZhen Low  
- Dart Exports** Generates and maintains an index.dart file... nolancorcoran  
- Dart Extensions Snippets** Snippets to create extensions in Dart Jordan ALCARAZ  

Dart

v3.28.0

Dart Code | ⚡ 3,749,822 | ★★★★★(50)

Dart language support and debugger for Visual Studio Code.

[Disable](#)[Uninstall](#)

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)[chat](#) [discord](#) [chat](#) [gitter](#) [twitter](#) [dartcode](#) [help](#) [contribute](#)

Introduction

Dart Code extends [VS Code](#) with support for the [Dart](#) programming language, and provides tools for effectively editing, refactoring, running, and reloading [Flutter](#) mobile apps, and [AngularDart](#) web apps.

Installation

Dart Code can be [installed from the Visual Studio Code Marketplace](#) or by [searching within VS Code](#).

Features

- Edit and Debug Flutter mobile apps (launch using F5 or the [Debug menu](#))
- Edit and Debug Dart command line apps (launch using F5 or the [Debug menu](#))
- Automatic hot reloads for Flutter
- Refactorings and Code fixes ([lightbulb](#))
- Quickly switch between devices for Flutter
- Flutter Doctor command
- Flutter Get Packages command
- Flutter Upgrade Packages command
- Automatically gets packages when pubspec.yaml is saved
- Automatically finds SDKs from PATH
- Notification of new stable Dart SDK releases

Categories

[Programming Languages](#) [Snippets](#)
[Linters](#) [Formatters](#) [Debuggers](#)

Resources

[Marketplace](#)
[Repository](#)
[License](#)

More Info

Released on 8/5/2016, 21:51:42
Last updated 11/1/2021, 23:30:05
Identifier dart-code.dart-code



EXTENSIONS: MARKETPLACE

material icon

- Material Icon Theme** 42ms
Material Design Icons for Visual Studio Code
Philipp Kief
- Angular Material 2, Flex Ia...** 228K ★ 5
Provides snippets Angular Material 2, An...
1Ton Technologies [Install](#)
- Material Theme Icons** 593K ★ 5
Material Theme Icons, the most epic icons...
Equinusocio [Install](#)
- Icon Fonts** 291K ★ 5
Snippets for popular icon fonts such as Fo...
idleberg [Install](#)
- Gruvbox Material Icon Theme** 4K ★ 5
Gruvbox Material Icons
JonathanHarty [Install](#)
- Community Material The...** 620K ★ 4.5
The official community maintained Materi...
Equinusocio [Install](#)
- Material Theme** 536K ★ 5
The most epic theme now for Visual Studi...
Equinusocio [Install](#)
- Unofficial Material Icon Theme** 2K
Unofficial Material Design Icons for Visual...
artalatarta [Install](#)
- Unofficial Material Icon Theme** 788
Unofficial Material Design Icons for Visual...
artalatarta [Install](#)
- Angular material icon picker** 416 ★ 5
Search angular material icons directly fro...
niccolofanton [Install](#)
- Sublime Material Theme** 468K ★ 4.5
Port of the Material Theme for Sublime Te...
Jarvis Prestidge [Install](#)



Material Icon Theme v4.10.0

Philipp Kief | 9,880,814 | ★★★★★(224)

Material Design Icons for Visual Studio Code

[Set File Icon Theme](#) [Disable](#) [Uninstall](#) [⚙️](#)

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)

File icons

Icon	Name	Icon	Name	Icon	Name	Icon	Name	Icon	Name	Categories
3d	3d	Css-map	Css-map	Image	Image	Npm	Npm	Silverstripe	Silverstripe	Themes
Abc	Abc	Cucumber	Cucumber	Imba	Imba	Nrwl	Nrwl	Sketch	Sketch	Resources
Actionscript	Actionscript	Cuda	Cuda	Ionic	Ionic	Nuget	Nuget	Slim	Slim	Marketplace
Ada	Ada	D	D	Istanbul	Istanbul	Nunjucks	Nunjucks	Slug	Slug	Repository
Adonis	Adonis	Dart	Dart	Jar	Jar	Nuxt	Nuxt	Smarty	Smarty	License
Advpl_include	Advpl_include	Database	Database	Java	Java	Ocaml	Ocaml	Sml	Sml	More Info
Advpl_prw	Advpl_prw	Denizscript	Denizscript	Javaclass	Javaclass	Odin	Odin	Snowpack	Snowpack	Released on
Advpl_ptm	Advpl_ptm	Dhall	Dhall	Javascript	Javascript	Opa	Opa	Snyk	Snyk	Last updated
Advpl_tpp	Advpl_tpp	Diff	Diff	Javascript-map	Javascript-map	Opam	Opam	Solidity	Solidity	Identifier
Android	Android	Dinophp	Dinophp	Jenkins	Jenkins	Pascal	Pascal	Stencil	Stencil	
Angular	Angular	Disc	Disc	Jest	Jest	Pawn	Pawn	Stitches	Stitches	
Angular-component	Angular-component	Django	Django	Jinja	Jinja	Pdf	Pdf	Storybook	Storybook	
Angular-directive	Angular-directive	Docker	Docker	Jsconfig	Jsconfig	Percy	Percy	Stryker	Stryker	
Angular-guard	Angular-guard	Document	Document	Json	Json	Perl	Perl	Stylelint	Stylelint	



-  New File Ctrl+N
-  New Window Ctrl+Shift+N
-  New File... Ctrl+Alt+Windows+N
-
-  Open File... Ctrl+O
-  **Open Folder...** Ctrl+K Ctrl+O
-  Open Workspace from File...
-  Open Recent >
-
-  Add Folder to Workspace...
-  Save Workspace As...
-  Duplicate Workspace
-
-  Save Ctrl+S
-  Save As... Ctrl+Shift+S
-  Save All Ctrl+K S
-
-  Auto Save
-  Preferences >
-
-  Revert File
-  Close Editor Ctrl+F4
-  Close Window Alt+F4
-
-  Exit

Show All Commands `Ctrl + Shift + P`Open File `Ctrl + O`Open Folder `Ctrl + K Ctrl + O`Open Recent `Ctrl + R`

EXPLORER



You might be prompted whether you "Trust" this folder and you should agree there.

- > ios
- > lib
- > test
- > web
 - ↳ .gitignore
 - ↳ .metadata
 - ↳ .packages
 - ↳ first_app.iml
 - ↳ pubspec.lock
 - ↳ pubspec.yaml
 - ↳ README.md

> DEPENDENCIES

EXPLORER

...

FIRST_APP



- > .dart_tool
- > .idea
- > android
- > ios
- > lib
- > test
- > web
- > .gitignore
- > .metadata
- > .packages
- > first_app.iml
- > pubspec.lock
- > pubspec.yaml
- > README.md



> DEPENDENCIES

EXPLORER

...

pubspec.yaml X

Download Refresh Sync Close More

FIRST_APP



- > .dart_tool
- > .idea
- > android
- > ios
- > lib
- > test
- > web
 - .gitignore
 - .metadata
 - .packages
 - first_app.iml
 - pubspec.lock
 - pubspec.yaml
- README.md

> DEPENDENCIES

```
1 name: first_app
2 description: A new Flutter project.
3
4 # The following line prevents the package from being accidentally publi
5 # pub.dev using `pub publish`. This is preferred for private packages.
6 publish_to: 'none' # Remove this line if you wish to publish to pub.dev
7
8 # The following defines the version and build number for your applicati
9 # A version number is three numbers separated by dots, like 1.2.43
10 # followed by an optional build number separated by a +.
11 # Both the version and the builder number may be overridden in flutter
12 # build by specifying --build-name and --build-number, respectively.
13 # In Android, build-name is used as versionName while build-number used
14 # Read more about Android versioning at https://developer.android.com/s
15 # In iOS, build-name is used as CFBundleShortVersionString while build-
16 # Read more about iOS versioning at
17 # https://developer.apple.com/library/archive/documentation/General/Ref
18 version: 1.0.0+1
19
20 environment:
21   sdk: ">=2.11.0 <3.0.0"
22
23 dependencies:
24   flutter:
25     sdk: flutter
```

EXPLORER



And this integrated terminal also already navigated into your project folder. So any commands you run there are executed from inside the project folder!



```
> ios ios  
└ lib  
    └ main.dart  
> test  
> web  
.gitignore  
.metadata  
.packages  
first_app.iml  
pubspec.lock  
pubspec.yaml  
README.md
```

main.dart

```
12  
13     theme: ThemeData(  
14         // This is the theme of your application.  
15         //  
16         // Try running your application with "flutter run". You'll see  
17         // application has a blue toolbar. Then, without quitting the a  
18         // changing the primarySwatch below to Colors.green and then in  
19         // "hot reload" (press "r" in the console where you ran "flutte  
20         // or simply save your changes to "hot reload" in a Flutter IDE  
21         // Notice that the counter didn't reset back to zero; the appli  
22         // is not restarted.
```

TERMINAL

PROBLEMS

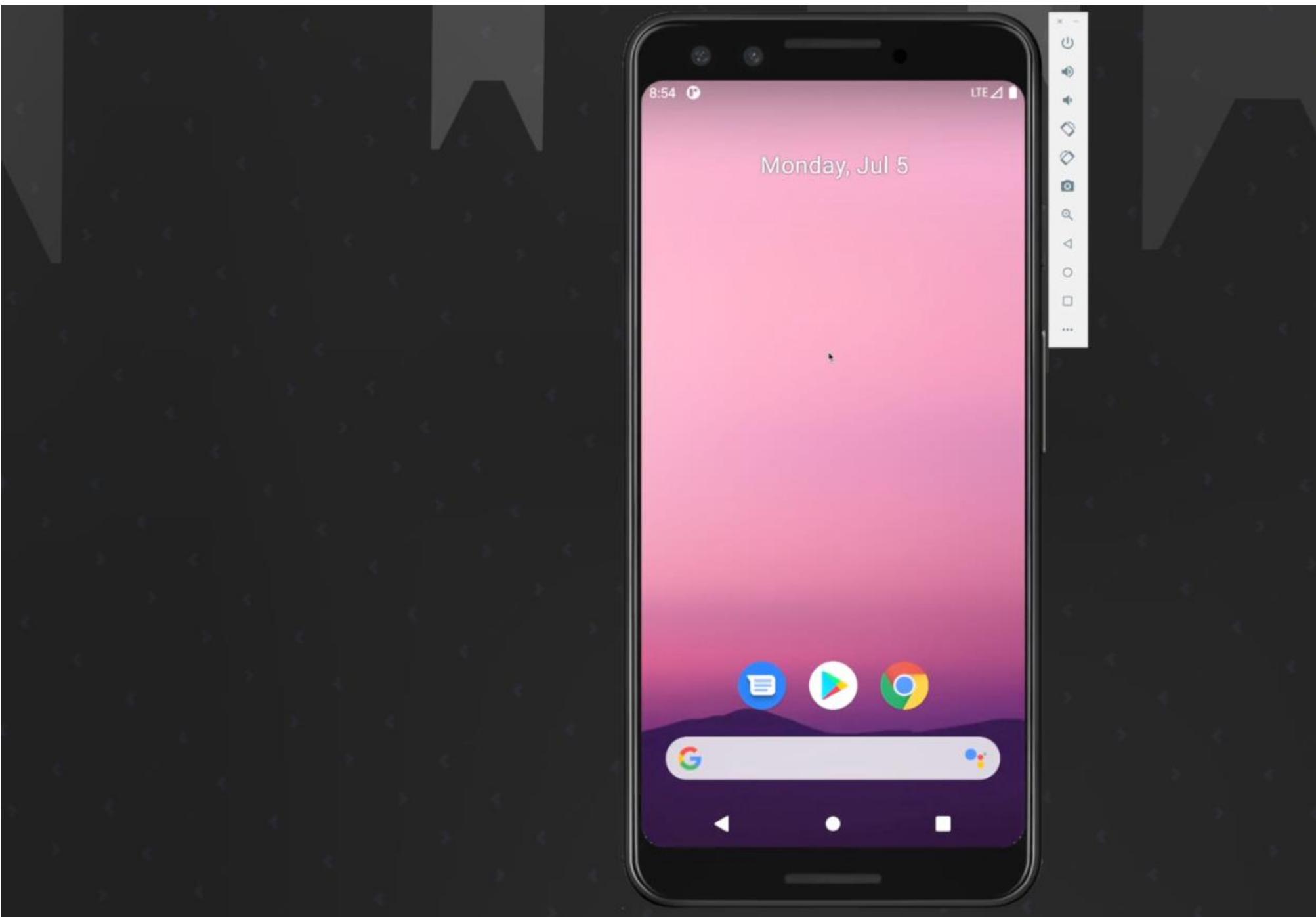
OUTPUT

DEBUG CONSOLE

 zsh + ^ X

first_app \$

> DEPENDENCIES



EXPLORER

...

main.dart X

D □ ...

FIRST_APP

- > .dart_tool
- > .idea
- > android
- > build
- > ios
- < lib
 - main.dart
- > test
- > web
- .gitignore
- .metadata
- .packages
- first_app.iml
- pubspec.lock
- pubspec.yaml
- README.md

```
8 // This widget is the root of your application.
9
10 @override
11 Widget build(BuildContext context) {
12   return MaterialApp(
13     title: 'Flutter Demo',
14     theme: ThemeData(
15       // This is the theme of your application.
16       //
17       // Try running your application with "flutter run". You'll see
18       // application has a blue toolbar. Then, without quitting the a
19       // changing the primarySwatch below to Colors.green and then in
// "hot reload" (press "r" in the console where you ran "flutte
```

TERMINAL

PROBLEMS

OUTPUT

DEBUG CONSOLE

bash + ^ ×

```
first_app $ flutter run
Using hardware rendering with device sdk gphone x86. If you notice graphics
artifacts, consider enabling software rendering with
"--enable-software-rendering".
Launching lib/main.dart on sdk gphone x86 in debug mode...
Running Gradle task 'assembleDebug'...
Running Gradle task 'assembleDebug'... Done
✓ Built build/app/outputs/flutter-apk/app-debug.apk.
Installing build/app/outputs/flutter-apk/app.apk...
98.3s
7.0s
```

EXPLORER

...

main.dart X

D □ ...

FIRST_APP

- > .dart_tool
- > .idea
- > android
- > build
- > ios
- > lib
 - main.dart
- > test
- > web

```
8 // This widget is the root of your application.
9
10 @override
11 Widget build(BuildContext context) {
12   return MaterialApp(
13     title: 'Flutter Demo',
14     theme: ThemeData(
15       //
16       // Try running your application with "flutter run". You'll see
```

TERMINAL

PROBLEMS

OUTPUT

DEBUG CONSOLE

bash + ^ x

Installing build/app/outputs/flutter-apk/app.apk...
Syncing files to device sdk gphone x86...

7.0s
128ms

Flutter run key commands.

r Hot reload. 🔥🔥🔥

R Hot restart.

h Repeat this help message.

d Detach (terminate "flutter run" but leave application running).

c Clear the screen

q Quit (terminate the application on the device).

Running with unsound null safety

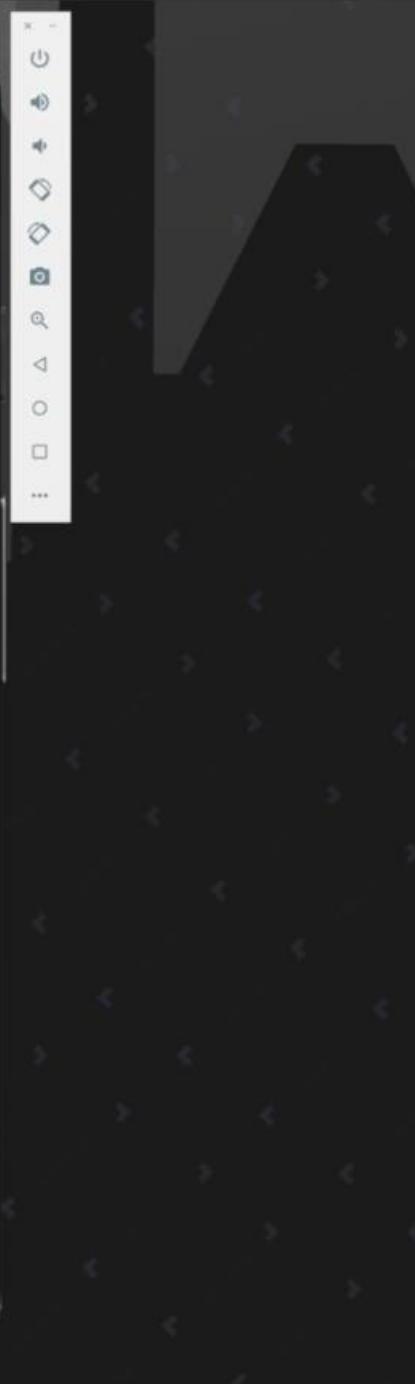
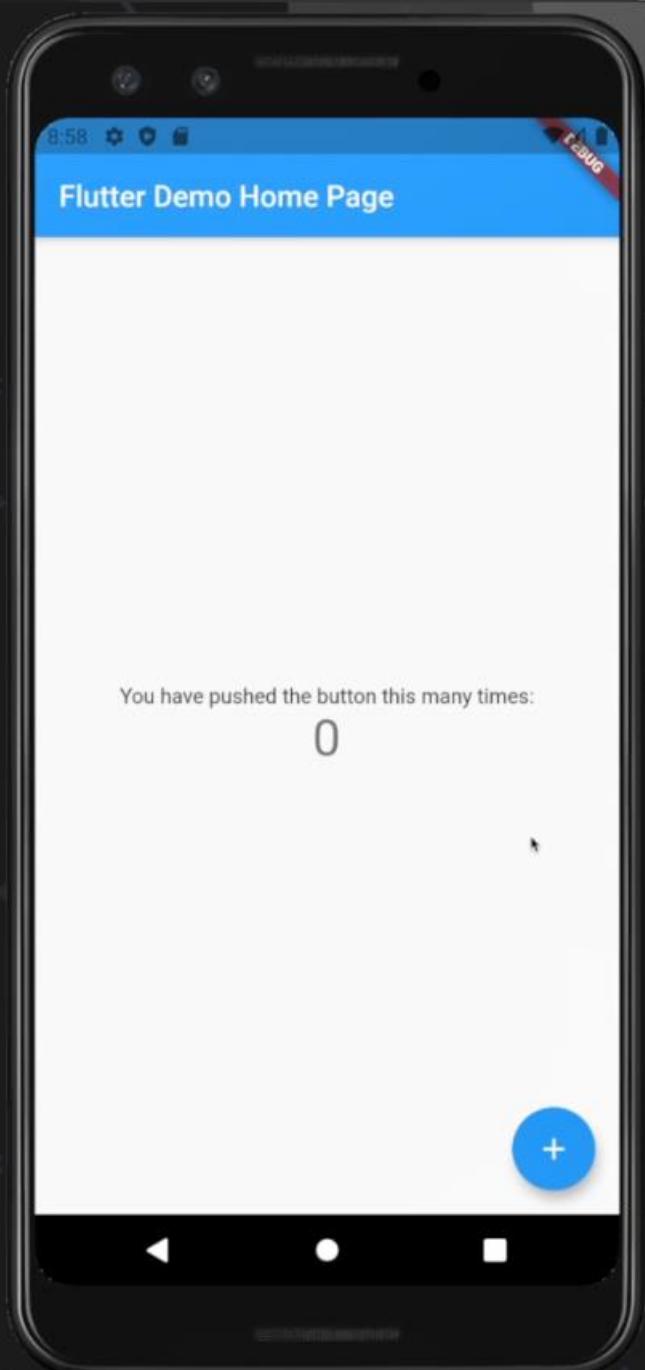
For more information see <https://dart.dev/null-safety/unsound-null-safety>

An Observatory debugger and profiler on sdk gphone x86 is available at:

<http://127.0.0.1:64370/BwxlBMVxppE=/>

Activating Dart DevTools...

> DEPENDENCIES



CAPTURE

- Start Debugging
- Run Without Debugging**
- Stop Debugging
- Restart Debugging
- Open Configurations
- Add Configuration...

- Step Over
- Step Into
- Step Out
- Continue

- Toggle Breakpoint
- New Breakpoint
- >

- Enable All Breakpoints
- Disable All Breakpoints
- Remove All Breakpoints
- Install Additional Debuggers...



main.dart

main.dart - first_app

```
47  
48     class _MyHomePageState extends State<MyHomePage> {  
49         int _counter = 0;  
50  
51         void _incrementCounter() {  
52             setState(() {  
53                 // This call to setState tells the Flutter framework that something  
54                 // changed in this State, which causes it to rerun the build method  
55                 // so that the display can reflect the updated values. If we chan  
56                 // _counter without calling setState(), then the build method wou  
57                 // be called again, and so nothing would appear to happen.  
58                 _counter = _counter + 2;  
59             });  
60         }  
61  
62         @override  
63         Widget build(BuildContext context) {  
64             // This method is rerun every time setState is called, for instance  
65             // by the _incrementCounter method above.  
66             //  
67             // The Flutter framework has been optimized to make rerunning build  
68             // fast, so that you can just rebuild anything that needs updating  
69             // than having to individually change instances of widgets.  
70             return Scaffold(  
71                 appBar: AppBar(
```

EXPLORER

...



main.dart X

D □ ...

FIRST_APP

> .dart_tool

> .idea

> android

> build

> ios

> lib

| main.dart

> test

> web

.gitignore

.metadata

.packages

first_app.iml

pubspec.lock

pubspec.yaml

README.md

```
47  
48 class _MyHomePageState extends State<MyHomePage> {  
49     int _counter = 0;  
50  
51     void _incrementCounter() {  
52         setState(() {  
53             // This call to setState tells the Flutter framework that something  
54             // changed in this State, which causes it to rerun the build method  
55             // so that the display can reflect the updated values. If we chan-  
56             // _counter without calling setState(), then the build method wou-  
57             // be called again, and so nothing would appear to happen.  
58             _counter = _counter + 1;  
59         });  
60     }  
61  
62     @override  
63     Widget build(BuildContext context) {
```

DEBUG CONSOLE

...

Filter (e.g. text, !exclude)

≡ ^ X

```
Launching lib/main.dart on sdk gphone x86 in debug mode... lib/main.dart:1  
✓ Built build/app/outputs/flutter-apk/app-debug.apk.  
Connecting to VM Service at ws://127.0.0.1:64803/mN9f1o9sEfE=/ws  
Reloaded 1 of 553 libraries in 1,420ms.
```

> DEPENDENCIES

>

FLUTTER WINDOWS SETUP

Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin... +

https://docs.flutter.dev/get-started/install/windows

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

1. Install

2. Set up an editor

3. Test drive

4. Write your first app

5. Learn more

From another platform?

Flutter for Android devs

Flutter for iOS devs

Flutter for React Native devs

Flutter for web devs

Flutter for Xamarin.Forms devs

Introduction to declarative UI

Dart language overview ↗

Building a web app

Samples & tutorials

Development

User interface

Set up an editor ↗

Windows install

Get started > Install > Windows

System requirements

To install and run Flutter, your development environment must meet these minimum requirements:

- **Operating Systems:** Windows 7 SP1 or later (64-bit), x86-64 based.
- **Disk Space:** 1.64 GB (does not include disk space for IDE/tools).
- **Tools:** Flutter depends on these tools being available in your environment.
 - [Windows PowerShell 5.0](#) or newer (this is pre-installed with Windows 10)
 - [Git for Windows](#) 2.x, with the **Use Git from the Windows Command Prompt** option.

If Git for Windows is already installed, make sure you can run `git` commands from the command prompt or PowerShell.

Get the Flutter SDK

1. Download the following installation bundle to get the latest stable release of the Flutter SDK:

[flutter windows 2.5.3-stable.zip](#)

Zoom

Contents

System requirements

Get the Flutter SDK

Update your path

Run flutter doctor

Android setup

Install Android Studio

Set up your Android device

Set up the Android emulator

Agree to Android Licenses

Windows setup

Additional Windows requirements

Enable desktop support

Web setup

Next step

Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin... +

https://docs.flutter.dev/get-started/install/windows

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started
1. Install
2. Set up an editor
3. Test drive
4. Write your first app
5. Learn more
From another platform?
Flutter for Android devs
Flutter for iOS devs
Flutter for React Native devs
Flutter for web devs
Flutter for Xamarin.Forms devs
Introduction to declarative UI
Dart language overview ↗
Building a web app
Samples & tutorials
Development
User interface

Get the Flutter SDK

1. Download the following installation bundle to get the latest stable release of the Flutter SDK:

[flutter_windows_2.5.3-stable.zip](#)

For other release channels, and older builds, see the [SDK releases](#) page.

2. Extract the zip file and place the contained `flutter` in the desired installation location for the Flutter SDK (for example, `C:\Users\<your-user-name>\Documents`).

⚠ Warning: Do not install Flutter in a directory like `C:\Program Files\` that requires elevated privileges.

If you don't want to install a fixed version of the installation bundle, you can skip steps 1 and 2. Instead, get the source code from the [Flutter repo](#) on GitHub, and change branches or tags as needed. For example:

```
C:\src>git clone https://github.com/flutter/flutter.git -b stable
```

You are now ready to run Flutter commands in the Flutter Console.

Update your path

If you wish to run Flutter commands in the regular Windows console, take these steps to add Flutter to the `PATH` environment variable:

Contents

[System requirements](#)

[Get the Flutter SDK](#)

[Update your path](#)

[Run flutter doctor](#)

[Android setup](#)

[Install Android Studio](#)

[Set up your Android device](#)

[Set up the Android emulator](#)

[Agree to Android Licenses](#)

[Windows setup](#)

[Additional Windows requirements](#)

[Enable desktop support](#)

[Web setup](#)

[Next step](#)

Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin... +

https://docs.flutter.dev/get-started/install/windows

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

- 1. Install
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

User interface

Update your path

If you wish to run Flutter commands in the regular Windows console, take these steps to add Flutter to the [PATH](#) environment variable:

- From the Start search bar, enter 'env' and select **Edit environment variables for your account**.
- Under **User variables** check if there is an entry called **Path**:
 - If the entry exists, append the full path to `flutter\bin` using ; as a separator from existing values.
 - If the entry doesn't exist, create a new user variable named **Path** with the full path to `flutter\bin` as its value.

You have to close and reopen any existing console windows for these changes to take effect.

Note: As of Flutter's 1.19.0 dev release, the Flutter SDK contains the `dart` command alongside the `flutter` command so that you can more easily run Dart command-line programs. Downloading the Flutter SDK also downloads the compatible version of Dart, but if you've downloaded the Dart SDK separately, make sure that the Flutter version of `dart` is first in your path, as the two versions might not be compatible. The following command tells you whether the `flutter` and `dart` commands originate from the same `bin` directory and are therefore compatible.

```
C:\>where flutter dart
C:\path-to-flutter-sdk\bin\flutter
C:\path-to-flutter-sdk\bin\flutter.bat
C:\path-to-dart-sdk\bin\dart.exe      :: this should go after `C:\path-to-flutter-sdk\bin\` comm
C:\path-to-flutter-sdk\bin\dart
C:\path-to-flutter-sdk\bin\dart.bat
```

As shown above, the command `dart` from the Flutter SDK doesn't come first. Update your path to use commands from `C:\path-to-flutter-sdk\bin\` before commands from `C:\path-to-dart-sdk\bin\` (in this case). After restarting your shell for the changes to take effect, running the `where` command again should show that the `flutter` and `dart` commands

Contents

System requirements

Get the Flutter SDK

Update your path

Run `flutter doctor`

Android setup

Install Android Studio

Set up your Android device

Set up the Android emulator

Agree to Android Licenses

Windows setup

Additional Windows requirements

Enable desktop support

Web setup

Next step

Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin

https://git-scm.com/download/win

Search entire site...

git --distributed-is-the-new-centralized

About

Documentation

Downloads

GUI Clients

Logos

Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloading Git

Your download is starting...

You are downloading the latest (**2.34.0**) **64-bit** version of **Git for Windows**. This is the most recent [maintained build](#). It was released **2 days ago**, on 2021-11-15.

[Click here to download manually](#), if your download hasn't started.

Other Git for Windows downloads

[Git for Windows Setup](#)

[32-bit Git for Windows Setup](#).

[64-bit Git for Windows Setup](#).

[Git for Windows Portable \("thumbdrive edition"\)](#)

[32-bit Git for Windows Portable](#).

[64-bit Git for Windows Portable](#).

The current source code release is version **2.34.0**. If you want the newer version, you can build it from [the source code](#).

Now What?

Now that you have downloaded Git, it's time to start using it.

github.com/git-for-windows/git/releases/download/.../PortableGit-2.34.0-64-bit.7z.exe

Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin... +

https://docs.flutter.dev/get-started/install/windows

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Get started

1. Install

1. Set up an editor
3. Test drive
4. Write your first app
5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI
- Dart language overview ↗
- Building a web app

Samples & tutorials

Development

User interface

Android setup

Note: Flutter relies on a full installation of Android Studio to supply its Android platform dependencies. However, you can write your Flutter apps in a number of editors; a later step discusses that.

Install Android Studio

1. Download and install [Android Studio](#).
2. Start Android Studio, and go through the 'Android Studio Setup Wizard'. This installs the latest Android SDK, Android SDK Command-line Tools, and Android SDK Build-Tools, which are required by Flutter when developing for Android.
3. Run `flutter doctor` to confirm that Flutter has located your installation of Android Studio. If Flutter cannot locate it, run `flutter config --android-studio-dir <directory>` to set the directory that Android Studio is installed to.

Set up your Android device

To prepare to run and test your Flutter app on an Android device, you need an Android device running Android 4.1 (API level 16) or higher.

1. Enable **Developer options** and **USB debugging** on your device. Detailed instructions are available in the [Android documentation](#).
2. Windows-only: Install the [Google USB Driver](#).
3. Using a USB cable, plug your phone into your computer. If prompted on your device, authorize your computer to access your device.
4. In the terminal run the `flutter devices` command to verify that Flutter recognizes your connected Android device. By

Contents

System requirements

Get the Flutter SDK

Update your path

Run flutter doctor

Android setup

Install Android Studio

Set up your Android device

Set up the Android emulator

Agree to Android Licenses

Windows setup

Additional Windows requirements

Enable desktop support

Web setup

Next step

Windows install | Flutter Git - Downloading Package Download Android Studio and S... Visual Studio Code - Code Editin... +

https://docs.flutter.dev/get-started/install/windows

Flutter

Multi-Platform ▾ Development ▾ Ecosystem ▾ Showcase Docs ▾ Get started

Set up your Android device

To prepare to run and test your Flutter app on an Android device, you need an Android device running Android 4.1 (API level 16) or higher.

1. Enable **Developer options** and **USB debugging** on your device. Detailed instructions are available in the [Android documentation](#).
2. Windows-only: Install the [Google USB Driver](#).
3. Using a USB cable, plug your phone into your computer. If prompted on your device, authorize your computer to access your device.
4. In the terminal, run the `flutter devices` command to verify that Flutter recognizes your connected Android device. By default, Flutter uses the version of the Android SDK where your `adb` tool is based. If you want Flutter to use a different installation of the Android SDK, you must set the `ANDROID_SDK_ROOT` environment variable to that installation directory.

Set up the Android emulator

To prepare to run and test your Flutter app on the Android emulator, follow these steps:

1. Enable [VM acceleration](#) on your machine.
2. Launch **Android Studio**, click the **AVD Manager** icon, and select **Create Virtual Device...**
 - In older versions of Android Studio, you should instead launch **Android Studio > Tools > Android > AVD Manager** and select **Create Virtual Device...**. (The **Android** submenu is only present when inside an Android project.)
 - If you do not have a project open, you can choose **Configure > AVD Manager** and select **Create Virtual Device...**
3. Choose a device definition and select **Next**.
4. Select one or more system images for the Android versions you want to emulate, and select **Next**. An `x86` or `x86_64` image is recommended.
5. Under Emulated Performance, select **Hardware - GLES 2.0** to enable [hardware acceleration](#).
6. Verify the AVD configuration is correct, and select **Finish**.

Get started

- 1. [Install](#)
- 2. Set up an editor
- 3. Test drive
- 4. Write your first app
- 5. Learn more

From another platform?

- Flutter for Android devs
- Flutter for iOS devs
- Flutter for React Native devs
- Flutter for web devs
- Flutter for Xamarin.Forms devs
- Introduction to declarative UI

Dart language overview ↗

Building a web app

Samples & tutorials

Development

User interface

Contents

System requirements

Get the Flutter SDK

Update your path

Run flutter doctor

Android setup

Install Android Studio

Set up your Android device

Set up the Android emulator

Agree to Android Licenses

Windows setup

Additional Windows requirements

Enable desktop support

Web setup

Next step

A screenshot of a web browser window showing the Android Studio download page. The address bar contains the URL <https://developer.android.com/studio>. The page header includes tabs for "macOS install | Flutter", "Windows install | Flutter", "Git - Downloading Package", and "Download Android Studio and S...". The main navigation menu has items like "developers" (with an Android icon), "Platform", "Android Studio" (underlined in green), "Google Play", "Jetpack", "Kotlin", "More", "Search" (with a magnifying glass icon), "English" (with a dropdown arrow), and "Sign in".

ANDROID STUDIO

[Download](#) [What's new](#) [User guide](#) [Preview](#)



Android Studio provides the fastest tools for building apps on every type of Android device.

[Download Android Studio](#)

2020.3.1 for Windows 64-bit (914 MiB)

[Download options](#)

[Release notes](#)

**Appearance & Behavior**

- Appearance
- Menus and Toolbars
- System Settings
 - HTTP Proxy
 - Data Sharing
 - Date Formats
 - Updates
 - Process Elevation
 - Passwords

Android SDK

- Memory Settings
- Notifications
- Quick Lists
- Path Variables

Keymap

- > Editor
- > Build, Execution, Deployment
- > Languages & Frameworks
- > Tools

Appearance & Behavior > System Settings > Android SDK

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: Q:\myprograms\Android\AndroidSDK

[Edit](#) [Optimize disk space](#)[SDK Platforms](#) [SDK Tools](#) [SDK Update Sites](#)

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, the IDE will automatically check for updates. Check "show package details" to display individual SDK components.

	Name	API Level	Revision	Status
	<input type="checkbox"/> Android Sv2 Preview	Sv2	1	Not installed
	<input type="checkbox"/> Android 12.0 (S)	31	1	Not installed
	<input checked="" type="checkbox"/> Android 11.0 (R)	30	3	Installed
	<input type="checkbox"/> Android 10.0 (Q)	29	5	Not installed
	<input type="checkbox"/> Android 9.0 (Pie)	28	6	Not installed
	<input type="checkbox"/> Android 8.1 (Oreo)	27	3	Not installed
	<input type="checkbox"/> Android 8.0 (Oreo)	26	2	Not installed
	<input type="checkbox"/> Android 7.1.1 (Nougat)	25	3	Not installed
	<input type="checkbox"/> Android 7.0 (Nougat)	24	2	Not installed
	<input type="checkbox"/> Android 6.0 (Marshmallow)	23	3	Not installed
	<input type="checkbox"/> Android 5.1 (Lollipop)	22	2	Not installed
	<input type="checkbox"/> Android 5.0 (Lollipop)	21	2	Not installed
	<input type="checkbox"/> Android 4.4W (KitKat Wear)	20	2	Not installed
	<input type="checkbox"/> Android 4.4 (KitKat)	19	4	Not installed
	<input type="checkbox"/> Android 4.3 (Jelly Bean)	18	3	Not installed
	<input type="checkbox"/> Android 4.2 (Jelly Bean)	17	3	Not installed
	<input type="checkbox"/> Android 4.1 (Jelly Bean)	16	5	Not installed
	<input type="checkbox"/> Android 4.0.3 (IceCreamSandwich)	15	5	Not installed

 Hide Obsolete Packages Show Package Details**OK****Cancel****Apply**

**Appearance & Behavior**

- Appearance
- Menus and Toolbars
- System Settings
 - HTTP Proxy
 - Data Sharing
 - Date Formats
 - Updates
 - Process Elevation
 - Passwords

Android SDK

- Memory Settings
- Notifications
- Quick Lists
- Path Variables

Keymap

- > Editor
- > Build, Execution, Deployment
- > Languages & Frameworks
- > Tools

Appearance & Behavior > System Settings > Android SDK

Manager for the Android SDK and Tools used by the IDE

Android SDK Location: Q:\myprograms\Android\AndroidSDK

[Edit](#) [Optimize disk space](#)[SDK Platforms](#) [SDK Tools](#) [SDK Update Sites](#)

Below are the available SDK developer tools. Once installed, the IDE will automatically check for updates. Check "show package details" to display available versions of an SDK Tool.

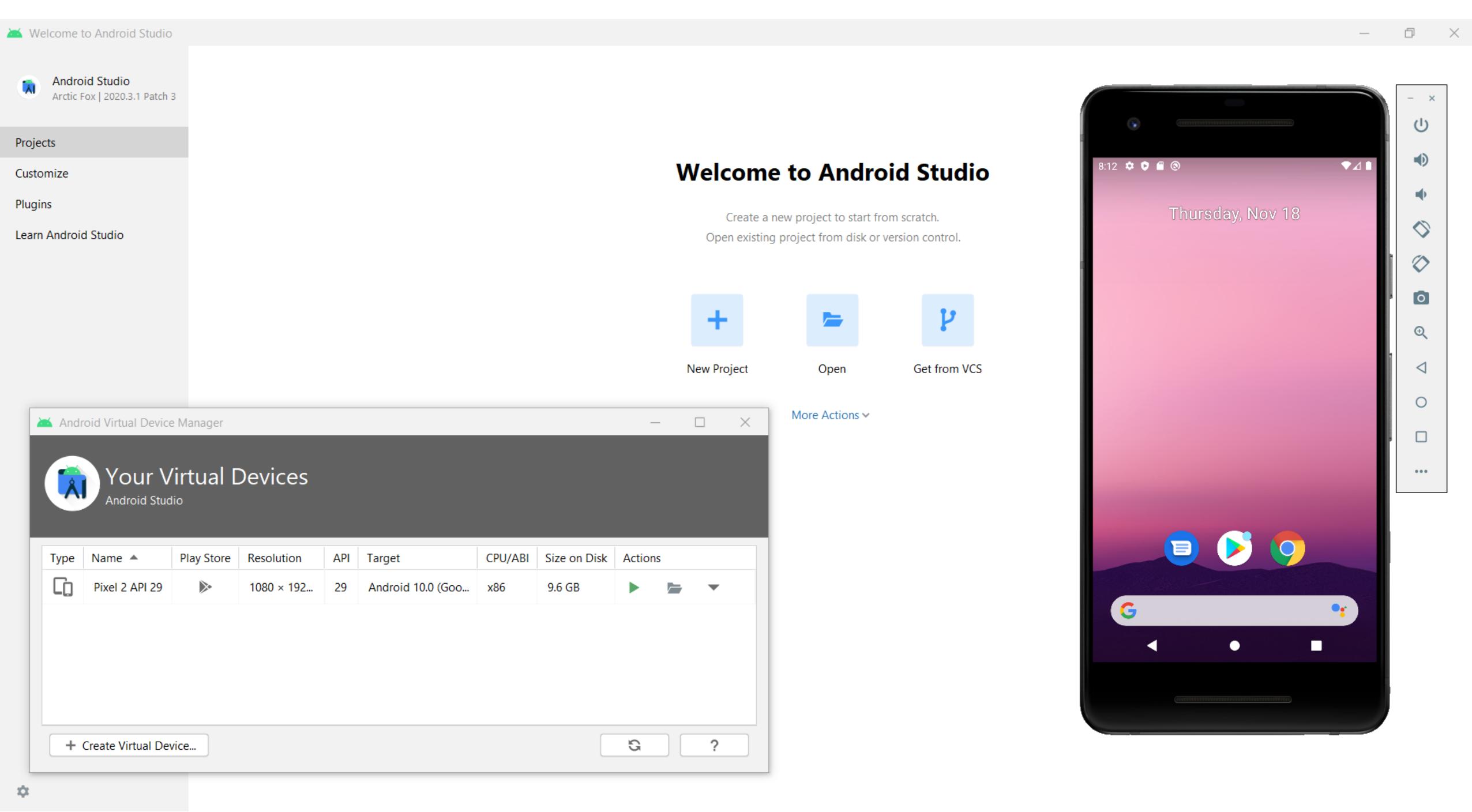
Name	Version	Status
<input checked="" type="checkbox"/> Android SDK Build-Tools 32-rc1		Installed
<input type="checkbox"/> NDK (Side by side)		Not Installed
<input checked="" type="checkbox"/> Android SDK Command-line Tools (latest)		Installed
<input type="checkbox"/> CMake		Not Installed
<input type="checkbox"/> Android Auto API Simulators	1	Not installed
<input type="checkbox"/> Android Auto Desktop Head Unit Emulator	1.1	Not installed
<input checked="" type="checkbox"/> Android Emulator	30.9.5	Installed
<input type="checkbox"/> Android Emulator Hypervisor Driver for AMD Processors (installer)	1.7.0	Not installed
<input checked="" type="checkbox"/> Android SDK Platform-Tools	31.0.3	Installed
<input type="checkbox"/> Google Play APK Expansion library	1	Not installed
<input type="checkbox"/> Google Play Instant Development SDK	1.9.0	Not installed
<input type="checkbox"/> Google Play Licensing Library	1	Not installed
<input type="checkbox"/> Google Play services	49	Not installed
<input type="checkbox"/> Google USB Driver	13	Not installed
<input type="checkbox"/> Google Web Driver	2	Not installed
<input checked="" type="checkbox"/> Intel x86 Emulator Accelerator (HAXM installer)	7.6.5	Installed
<input type="checkbox"/> Layout Inspector image server for API 29-30	6	Not installed
<input type="checkbox"/> Layout Inspector image server for API S	3	Not installed

 Hide Obsolete Packages Show Package Details

OK

Cancel

Apply



```
C:\Users\Rizqi>where flutter dart
Q:\myprograms\flutter\bin\flutter
Q:\myprograms\flutter\bin\flutter.bat
Q:\myprograms\flutter\bin\dart
Q:\myprograms\flutter\bin\dart.bat
```

```
C:\Users\Rizqi>flutter
Manage your Flutter app development.
```

Common commands:

```
flutter create <output directory>
Create a new Flutter project in the specified directory.
```

```
flutter run [options]
Run your Flutter application on an attached device or in an emulator.
```

```
Usage: flutter <command> [arguments]
```

Global options:

-h, --help	Print this usage information.
-v, --verbose	Noisy logging, including all shell commands executed. If used with "--help", shows hidden options. If used with "flutter doctor", shows additional diagnostic information. (Use "-vv" to force verbose logging in those cases.)
-d, --device-id	Target device id or name (prefixes allowed).
--version	Reports the version of this tool.
--suppress-analytics	Suppress analytics reporting when this command runs.

```
C:\Users\Rizqi>flutter doctor
Running "flutter pub get" in flutter_tools...                                10.7s
Doctor summary (to see all details, run flutter doctor -v):
[!] Flutter (Channel stable, 2.5.3, on Microsoft Windows [Version 10.0.19043.1288], locale en-ID)
[!] Android toolchain - develop for Android devices (Android SDK version 30.0.3)
  X cmdline-tools component is missing
    Run `path/to/sdkmanager --install "cmdline-tools;latest"`
    See https://developer.android.com/studio/command-line for more details.
  X Android license status unknown.
    Run `flutter doctor --android-licenses` to accept the SDK licenses.
    See https://flutter.dev/docs/get-started/install/windows#android-setup for more details.
[!] Chrome - develop for the web
[!] Android Studio (version 2020.3)
[!] Connected device (2 available)

! Doctor found issues in 1 category.

C:\Users\Rizqi>
```

```
Q:\>cd %ANDROID_SDK_ROOT%\tools\bin
```

```
Q:\myprograms\Android\AndroidSDK\tools\bin>sdkmanager --install "cmdline-tools;latest"
```

```
Warning: Mapping new ns http://schemas.android.com/repository/android/common/02 to old ns http://schemas.android.com/repository/android/common/01
```

```
Warning: Mapping new ns http://schemas.android.com/repository/android/generic/02 to old ns http://schemas.android.com/repository/android/generic/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/addon2/02 to old ns http://schemas.android.com/sdk/android/repo/addon2/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/repository2/02 to old ns http://schemas.android.com/sdk/android/repo/repository2/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/sys-img2/02 to old ns http://schemas.android.com/sdk/android/repo/sys-img2/01
```

```
Warning: Mapping new ns http://schemas.android.com/repository/android/common/02 to old ns http://schemas.android.com/repository/android/common/01
```

```
Warning: Mapping new ns http://schemas.android.com/repository/android/generic/02 to old ns http://schemas.android.com/repository/android/generic/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/addon2/02 to old ns http://schemas.android.com/sdk/android/repo/addon2/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/repository2/02 to old ns http://schemas.android.com/sdk/android/repo/repository2/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/sys-img2/02 to old ns http://schemas.android.com/sdk/android/repo/sys-img2/01
```

```
Warning: Mapping new ns http://schemas.android.com/repository/android/common/02 to old ns http://schemas.android.com/repository/android/common/01
```

```
Warning: Mapping new ns http://schemas.android.com/repository/android/generic/02 to old ns http://schemas.android.com/repository/android/generic/01
```

```
Warning: Mapping new ns http://schemas.android.com/sdk/android/repo/addon2/02 to old ns http://schemas.android.com/sdk/android/repo/addon2/01
```

```
Q:\myprograms\Android\AndroidSDK\tools\bin>flutter doctor --android-licenses
4 of 7 SDK package licenses not accepted. 100% Computing updates...
Review licenses that have not been accepted (y/N)? y
```

1/4: License android-googletv-license:

Terms and Conditions

This is the Google TV Add-on for the Android Software Development Kit License Agreement.

1. Introduction

1.1 The Google TV Add-on for the Android Software Development Kit (referred to in this License Agreement as the "Google TV Add-on" and specifically including the Android system files, packaged APIs, and Google APIs add-ons) is licensed to you subject to the terms of this License Agreement. This License Agreement forms a legally binding contract between you and Google in relation to your use of the Google TV Add-on.

1.2 "Google" means Google Inc., a Delaware corporation with principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States.

2. Accepting this License Agreement

2.1 In order to use the Google TV Add-on, you must first agree to this License Agreement. You may not use the Google TV Add-on if you do not accept this License Agreement.

2.2 You can accept this License Agreement by:

(A) clicking to accept or agree to this License Agreement, where this option is made available to you

```
Q:\myprograms\Android\AndroidSDK\tools\bin>flutter doctor
Doctor summary (to see all details, run flutter doctor -v):
[✓] Flutter (Channel stable, 2.5.3, on Microsoft Windows [Version 10.0.19043.1288], locale en-ID)
[✓] Android toolchain - develop for Android devices (Android SDK version 30.0.3)
[✓] Chrome - develop for the web
[✓] Android Studio (version 2020.3)
[✓] Connected device (2 available)

• No issues found!
```

```
Q:\myprograms\Android\AndroidSDK\tools\bin>
```

WINDOWS DEVELOPMENT ENVIRONMENT

```
Q:\myworkspaces\flutter_ws>flutter create first_app
Creating project first_app...
first_app\lib\main.dart (created)
first_app\pubspec.yaml (created)
first_app\README.md (created)
first_app\test\widget_test.dart (created)
first_app\.gitignore (created)
first_app\.idea\libraries\Dart_SDK.xml (created)
first_app\.idea\libraries\KotlinJavaRuntime.xml (created)
first_app\.idea\modules.xml (created)
first_app\.idea\runConfigurations\main_dart.xml (created)
first_app\.idea\workspace.xml (created)
first_app\.metadata (created)
first_app\analysis_options.yaml (created)
first_app\android\app\build.gradle (created)
first_app\android\app\src\main\kotlin\com\example\first_app\MainActivity.kt (created)
first_app\android\build.gradle (created)
first_app\android\first_app_android.iml (created)
first_app\android\.gitignore (created)
first_app\android\app\src\debug\AndroidManifest.xml (created)
first_app\android\app\src\main\AndroidManifest.xml (created)
first_app\android\app\src\main\res\drawable\launch_background.xml (created)
first_app\android\app\src\main\res\drawable-v21\launch_background.xml (created)
first_app\android\app\src\main\res\mipmap-hdpi\ic_launcher.png (created)
first_app\android\app\src\main\res\mipmap-mdpi\ic_launcher.png (created)
first_app\android\app\src\main\res\mipmap-xhdpi\ic_launcher.png (created)
first_app\android\app\src\main\res\mipmap-xxhdpi\ic_launcher.png (created)
first_app\android\app\src\main\res\mipmap-xxxhdpi\ic_launcher.png (created)
first_app\android\app\src\main\res\values\styles.xml (created)
```

```
first_app\ios\Runner\Info.plist (created)
first_app\ios\Runner.xcodeproj\project.xcworkspace\contents.xcworkspacedata (created)
first_app\ios\Runner.xcodeproj\project.xcworkspace\xcshareddata\IDEWorkspaceChecks.plist (created) first_
app\ios\Runner.xcodeproj\project.xcworkspace\xcshareddata\WorkspaceSettings.xcsettings
(created)
first_app\ios\Runner.xcworkspace\contents.xcworkspacedata (created)
first_app\ios\Runner.xcworkspace\xcshareddata\IDEWorkspaceChecks.plist (created)
first_app\ios\Runner.xcworkspace\xcshareddata\WorkspaceSettings.xcsettings (created)
first_app\first_app.iml (created)
first_app\web\favicon.png (created)
first_app\web\icons\Icon-192.png (created)
first_app\web\icons\Icon-512.png (created)
first_app\web\icons\Icon-maskable-192.png (created)
first_app\web\icons\Icon-maskable-512.png (created)
first_app\web\index.html (created)
first_app\web\manifest.json (created)
Running "flutter pub get" in first_app...                                6.2s
Wrote 81 files.
```

All done!

In order to run your application, type:

```
$ cd first_app
$ flutter run
```

Your application code is in first_app\lib\main.dart.

Q:\myworkspaces\flutter_ws>

The screenshot shows a browser window with multiple tabs open, including "Windows install | Flutter", "Git - Downloading Package", "Download Android Studio and Si", and "Visual Studio Code - Code Editin". The main content is the Visual Studio Code website at <https://code.visualstudio.com>.

The website features a large banner with the text "Code editing. Redefined." and "Free. Built on open source. Runs everywhere." Below this is a "Download for Windows" button for the "Stable Build". A note states: "By using VS Code, you agree to its license and privacy statement." The right side of the page displays the Visual Studio Code interface, showing the Extensions Marketplace sidebar with various extensions like Python, GitLens, C/C++, ESLint, Debugger for Chrome, Language Support, vscode-icons, Vetur, and C#. The main editor area shows a service worker file with code related to service workers and registration. The bottom status bar indicates the file is "master", has 0 errors and 0 warnings, and is in "JavaScript" mode.

Version 1.62 is now available! Read about the new features and fixes from October.

Code editing. Redefined.

Free. Built on open source. Runs everywhere.

Download for Windows
Stable Build

Other platforms and [Insiders Edition](#)

By using VS Code, you agree to its [license](#) and [privacy statement](#).

EXTENSIONS: MARKETPLACE

- Python 2019.6.24221 54.9M 4.5 Microsoft [Install](#)
- GitLens — Git sup... 9.8.5 23.1M 5 Eric Amodio [Install](#)
- C/C++ 0.24.0 23M 3.5 Microsoft [Install](#)
- ESLint 1.9.0 21.9M 4.5 Integrates ESLint JavaScript into VS ... Dirk Baeumer [Install](#)
- Debugger for Ch... 4.11.6 20.6M 4.5 Debug your JavaScript code in the C... Microsoft [Install](#)
- Language Supp... 0.47.0 18.7M 4.5 Java Linting, Intellisense, formatting, ... Red Hat [Install](#)
- vscode-icons 8.8.0 17.2M 5 Icons for Visual Studio Code VSCode Icons Team [Install](#)
- Vetur 0.21.1 17M 4.5 Vue tooling for VS Code Pine Wu [Install](#)
- C# 1.21.0 15.6M 4.4 C# for Visual Studio Code (powered ... Microsoft [Install](#)

File Edit Selection View Go Debug Terminal Help serviceWorker.js - create-react-app - Visual Studio Code - In... — □ X

JS App.js JS index.js JS serviceWorker.js

```
src > JS serviceWorker.js > register > window.addEventListener('load') callback
39   checkValidServiceWorker(swUrl, config);
40   // Add some additional logging to localhost, p...
41   // service worker/PWA documentation.
42   navigator.serviceWorker.ready.then(() => {
43     product
44     productSub
45     removeSiteSpecificTrackingException
46     removeWebWideTrackingException
47     requestMediaKeySystemAccess
48     sendBeacon
49     serviceWorker (property) Navigator.serviceWorke...
50     storage
51     storeSiteSpecificTrackingException
52     storeWebWideTrackingException
53   } userAgent
54   }
55 }
```

TERMINAL ... 1: node

```
You can now view create-react-app in the browser.
Local: http://localhost:3000/
On Your Network: http://10.211.55.3:3000/
Note that the development build is not optimized.
```

Ln 43, Col 19 Spaces: 2 UTF-8 LF JavaScript

EXTENSIONS: MARKETPLACE

flutter

- Flutter** Dart Code | 3,495,681 | ★★★★★(50)
Flutter support and debugger for Visual Studio Code.
- Awesome Flutter Snippets** 693K ★ 5
Awesome Flutter Snippets is a collection of snippets for Flutter.
- Flutter Widget Snippets** 405K ★ 5
A set of helpful widget snippets for day to day development.
- [FF] Flutter Files** 187K ★ 5
Quickly scaffold flutter bloc file templates.
- Flutter Tree** 125K ★ 5
Extension for Flutter to build basic widget trees.
- Flutter Intl** 101K ★ 5
Flutter localization binding from .arb files.
- Flutter Helpers** 79K ★ 5
Helper utilities for flutter projects.
- flutter-stylizer** 80K ★ 5
Flutter Stylizer organizes your Flutter class...
- Flutter Color** 83K ★ 4.5
This plugin helps you to easily visualize the...
- flutter_mobx** 30K ★ 5
A extension for using MobX in Flutter.
- vscode-flutter-i18n-json** 53K ★ 5
VS Code extension to create a binding between...

Flutter Command palette

0 22K 50

Extension: Flutter X



Flutter

v3.28.0

Dart Code | 3,495,681 | ★★★★★(50)

Flutter support and debugger for Visual Studio Code.

Disable

Uninstall

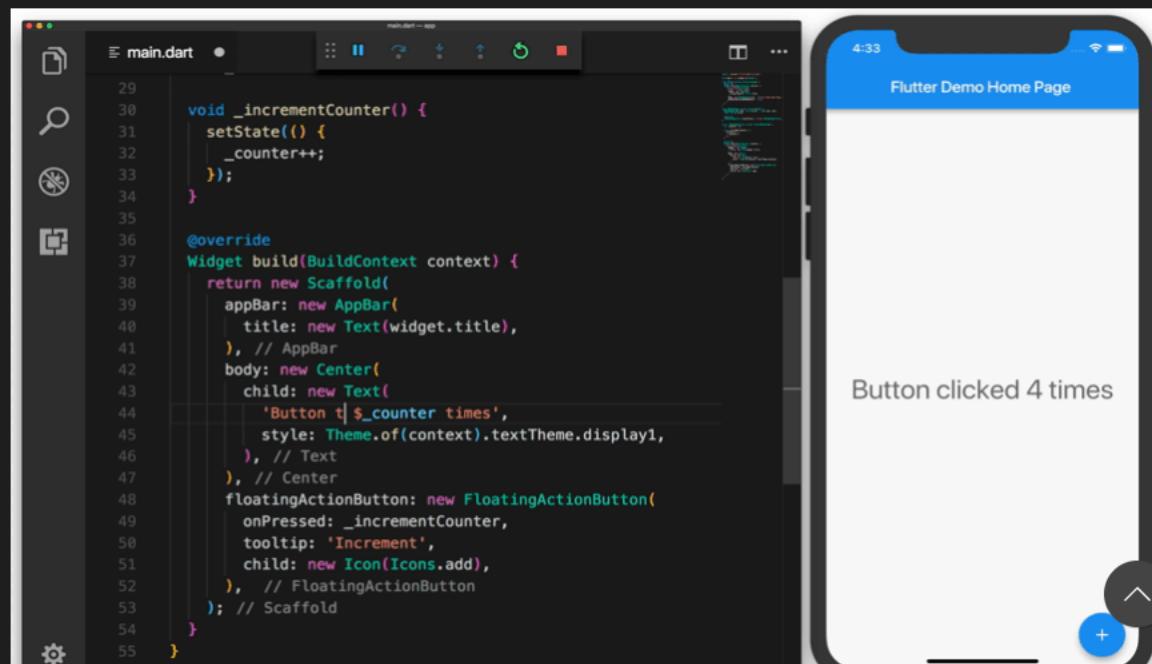


This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Dependencies](#) [Runtime Status](#)

Introduction

This [VS Code](#) extension adds support for effectively editing, refactoring, running, and reloading [Flutter](#) mobile apps, as well as support for the [Dart](#) programming language.



Categories

[Programming Languages](#) [Snippets](#)[Linters](#) [Formatters](#) [Debuggers](#)

Resources

[Marketplace](#)[Repository](#)[License](#)

More Info

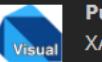
Released on 4/18/2018, 23:49:03

Last updated 11/1/2021, 23:30:15

Identifier dart-code.flutter

EXTENSIONS: MARKETPLACE

dart

- Dart** Dart language support and debugger for ... Dart Code 
- dart-import** Fix Dart/Flutter's imports Luan  
- Dart Data Class Generator** Create dart data classes easily, fast and wi... BendixMa  
- Dart (Syntax Highlighting ...** Syntax highlighting for Dart and nothing ... oscars  
- Json to Dart Model** Extension convert Json to Dart Model class hirantha  
- Flutter & Dart Utilities** Official package of Academia do Flutter (B... Rodrigo Rahman  
- Puzzle Dart** XAML for Dart Language Support bigmagician  
- Dart Getters And Setters** Dart Generate Getters And Setters Peter Haddad  
- Dart Built Value Snippets** VS Code Snippets for the Dart built_value ... YongZhen Low  
- Dart Exports** Generates and maintains an index.dart file... nolancorcoran  
- Dart Extensions Snippets** Snippets to create extensions in Dart Jordan ALCARAZ  

Dart

v3.28.0

Dart Code | ⚡ 3,749,822 | ★★★★★(50)

Dart language support and debugger for Visual Studio Code.



This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)[chat](#) [discord](#) [chat](#) [gitter](#) [twitter](#) [dartcode](#) [help](#) [contribute](#)

Introduction

Dart Code extends [VS Code](#) with support for the [Dart](#) programming language, and provides tools for effectively editing, refactoring, running, and reloading [Flutter](#) mobile apps, and [AngularDart](#) web apps.

Installation

Dart Code can be [installed from the Visual Studio Code Marketplace](#) or by [searching within VS Code](#).

Features

- Edit and Debug Flutter mobile apps (launch using F5 or the [Debug menu](#))
- Edit and Debug Dart command line apps (launch using F5 or the [Debug menu](#))
- Automatic hot reloads for Flutter
- Refactorings and Code fixes ([lightbulb](#))
- Quickly switch between devices for Flutter
- Flutter Doctor command
- Flutter Get Packages command
- Flutter Upgrade Packages command
- Automatically gets packages when pubspec.yaml is saved
- Automatically finds SDKs from PATH
- Notification of new stable Dart SDK releases

Categories

[Programming Languages](#) [Snippets](#)
[Linters](#) [Formatters](#) [Debuggers](#)

Resources

[Marketplace](#)
[Repository](#)
[License](#)

More Info

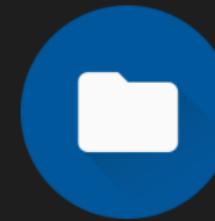
Released on 8/5/2016, 21:51:42
Last updated 11/1/2021, 23:30:05
Identifier dart-code.dart-code



EXTENSIONS: MARKETPLACE

material icon

- Material Icon Theme** 42ms
Material Design Icons for Visual Studio Code
Philipp Kief
- Angular Material 2, Flex Ia...** 228K ★ 5
Provides snippets Angular Material 2, An...
1Ton Technologies
- Material Theme Icons** 593K ★ 5
Material Theme Icons, the most epic icons...
Equinusocio
- Icon Fonts** 291K ★ 5
Snippets for popular icon fonts such as Fo...
idleberg
- Gruvbox Material Icon Theme** 4K ★ 5
Gruvbox Material Icons
JonathanHarty
- Community Material The...** 620K ★ 4.5
The official community maintained Materi...
Equinusocio
- Material Theme** 536K ★ 5
The most epic theme now for Visual Studi...
Equinusocio
- Unofficial Material Icon Theme** 2K
Unofficial Material Design Icons for Visual...
artalatarta
- Unofficial Material Icon Theme** 788
Unofficial Material Design Icons for Visual...
artalatarta
- Angular material icon picker** 416 ★ 5
Search angular material icons directly fro...
niccolofanton
- Sublime Material Theme** 468K ★ 4.5
Port of the Material Theme for Sublime Te...
Jarvis Prestidge



Material Icon Theme v4.10.0

Philipp Kief | 9,880,814 | ★★★★★(224)

Material Design Icons for Visual Studio Code

[Set File Icon Theme](#) [Disable](#) [Uninstall](#)

This extension is enabled globally.

[Details](#) [Feature Contributions](#) [Changelog](#) [Runtime Status](#)

File icons

Icon	Name	Icon	Name	Icon	Name	Icon	Name	Icon	Name	Categories
	3d		Css-map		Image		Npm		Silverstripe	Themes
	Abc		Cucumber		Imba		Nrwl		Sketch	Resources
	Actionscript		Cuda		Ionic		Nuget		Slim	Marketplace
	Ada		D		Istanbul		Nunjucks		Slug	Repository
	Adonis		Dart		Jar		Nuxt		Smarty	License
	Advpl_include		Database		Java		Ocaml		Sml	More Info
	Advpl_prw		Denizscript		Javaclass		Odin		Snowpack	Released on
	Advpl_ptm		Dhall		Javascript		Opa		Snyk	Last updated
	Advpl_tipp		Diff		Javascript-map		Opam		Solidity	Identifier
	Android		Dinophp		Jenkins		Pascal		Stencil	pkief.material-icon-theme
	Angular		Disc		Jest		Pawn		Stitches	
	Angular-component		Django		Jinja		Pdf		Storybook	
	Angular-directive		Docker		Jsconfig		Percy		Stryker	
	Angular-guard		Document		Json		Perl		Stylelint	



```
app\ios\Runner.xcodeproj\project.xcworkspace\xcshareddata\WorkspaceSettings.xcsettings  
(created)  
first_app\ios\Runner.xcworkspace\contents.xcworkspacedata (created)  
first_app\ios\Runner.xcworkspace\xcshareddata\IDEWorkspaceChecks.plist (created)  
first_app\ios\Runner.xcworkspace\xcshareddata\WorkspaceSettings.xcsettings (created)  
first_app\first_app.iml (created)  
first_app\web\favicon.png (created)  
first_app\web\icons\Icon-192.png (created)  
first_app\web\icons\Icon-512.png (created)  
first_app\web\icons\Icon-maskable-192.png (created)  
first_app\web\icons\Icon-maskable-512.png (created)  
first_app\web\index.html (created)  
first_app\web\manifest.json (created)  
Running "flutter pub get" in first_app... 6.2s  
Wrote 81 files.
```

All done!

In order to run your application, type:

```
$ cd first_app  
$ flutter run
```

Your application code is in first_app\lib\main.dart.

```
Q:\myworkspaces\flutter_ws>cd first_app
```

```
Q:\myworkspaces\flutter_ws\first_app>code .
```

```
Q:\myworkspaces\flutter_ws\first_app>
```

The screenshot shows the Visual Studio Code interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Title Bar:** pubspec.yaml - first_app - Visual Studio Code
- Left Sidebar (Explorer):**
 - OPEN EDITORS: pubspec.yaml
 - FIRST_APP:
 - .dart_tool
 - .idea
 - android
 - build
 - ios
 - lib
 - test
 - web
 - .gitignore
 - .metadata
 - .packages
 - analysis_options.yaml
 - first_app.iml
 - pubspec.lock
 - pubspec.yaml
 - README.md
- Central Area:** The pubspec.yaml file content is displayed.

```
1 name: first_app
2 description: A new Flutter project.
3
4 # The following line prevents the package from being accidentally published to
5 # pub.dev using `flutter pub publish`. This is preferred for private packages.
6 publish_to: 'none' # Remove this line if you wish to publish to pub.dev
7
8 # The following defines the version and build number for your application.
9 # A version number is three numbers separated by dots, like 1.2.43
10 # followed by an optional build number separated by a +.
11 # Both the version and the builder number may be overridden in flutter
12 # build by specifying --build-name and --build-number, respectively.
13 # In Android, build-name is used as versionName while build-number used as versi
14 # Read more about Android versioning at https://developer.android.com/studio/publish/app-signing/versioning
15 # In iOS, build-name is used as CFBundleShortVersionString while build-number us
16 # Read more about iOS versioning at
17 # https://developer.apple.com/library/archive/documentation/General/Reference/Infor
18 version: 1.0.0+1
19
20 environment:
21   sdk: ">=2.12.0 <3.0.0"
22
```
- Bottom Status Bar:** Ln 1, Col 1, Spaces: 2, UTF-8, CRLF, YAML, Dart DevTools, Flutter: 2.5.3, Pixel 2 API 29 (android-x86 emulator), Prettier

```
Q:\myworkspaces\flutter_ws\first_app>flutter run
Using hardware rendering with device Android SDK built for x86. If you notice graphics artifacts, consider enabling software rendering with "--enable-software-rendering".
Launching lib\main.dart on Android SDK built for x86 in debug mode...
Checking the license for package Android SDK Build-Tools 29.0.2 in Q:\myprograms\Android\AndroidSDK\...
License for package Android SDK Build-Tools 29.0.2 accepted.
Preparing "Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)"...
"Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)" ready.
Installing Android SDK Build-Tools 29.0.2 in Q:\myprograms\Android\AndroidSDK\build-tools\29.0.2
"Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)" complete.
"Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)" finished.
Running Gradle task 'assembleDebug'...                                332.2s
✓ Built build\app\outputs\flutter-apk\app-debug.apk.
Installing build\app\outputs\flutter-apk\app.apk...                  1,840ms
W/FlutterActivityAndFragmentDelegate( 5537): A splash screen was provided to Flutter, but this is deprecated. See flutter.dev/go/android-splash-migration for migration steps.
Syncing files to device Android SDK built for x86...                  262ms

Flutter run key commands.
r Hot reload.
R Hot restart.
h List all available interactive commands.
d Detach (terminate "flutter run" but leave application running).
c Clear the screen
q Quit (terminate the application on the device).
```

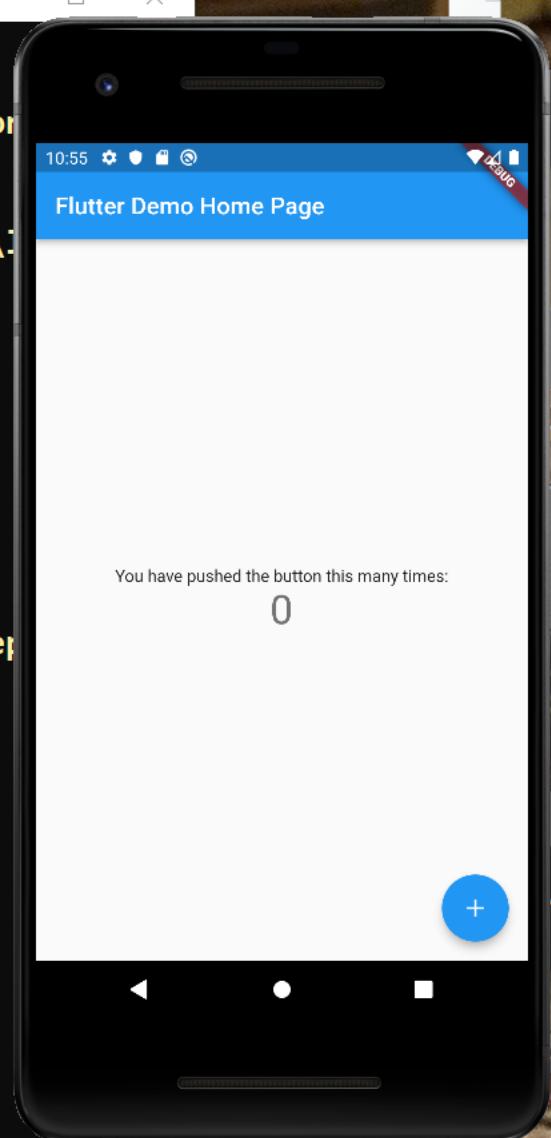
Running with sound null safety

An Observatory debugger and profiler on Android SDK built for x86 is available at:

<http://127.0.0.1:63882/m5Y708DiG98=/>

Activating Dart DevTools... 16.1s

The Flutter DevTools debugger and profiler on Android SDK built for x86 is available at:



MyCalendar...

```
Checking the license for package Android SDK Build-Tools 29.0.2 in Q:\myprograms\Android\AndroidSDK\l  
License for package Android SDK Build-Tools 29.0.2 accepted.  
Preparing "Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)".  
"Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)" ready.  
Installing Android SDK Build-Tools 29.0.2 in Q:\myprograms\Android\AndroidSDK\build-tools\29.0.2  
"Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)" complete.  
"Install Android SDK Build-Tools 29.0.2 (revision: 29.0.2)" finished.  
Running Gradle task 'assembleDebug'... 332.2s  
✓ Built build\app\outputs\flutter-apk\app-debug.apk.  
Installing build\app\outputs\flutter-apk\app.apk... 1,840ms  
W/FlutterActivityAndFragmentDelegate( 5537): A splash screen was provided to Flutter, but this is de  
. See flutter.dev/go/android-splash-migration for migration steps.  
Syncing files to device Android SDK built for x86... 262ms
```

Flutter run key commands.

- r Hot reload.
- R Hot restart.
- h List all available interactive commands.
- d Detach (terminate "flutter run" but leave application running).
- c Clear the screen
- q Quit (terminate the application on the device).

Running with sound null safety

An Observatory debugger and profiler on Android SDK built for x86 is available at:

<http://127.0.0.1:63882/m5Y708DiG98=/>

Activating Dart DevTools... 16.1s

The Flutter DevTools debugger and profiler on Android SDK built for x86 is available at:

<http://127.0.0.1:9101?uri=http://127.0.0.1:63882/m5Y708DiG98=/>

Application finished.

Press q



The screenshot shows the Visual Studio Code interface with the following details:

- File** Edit Selection View Go Run Terminal Help
- pubspec.yaml - first_app - Visual Studio Code
- EXPLORER
- OPEN EDITORS: pubspec.yaml
- FIRST_APP: .dart_tool, .idea, android, build, ios, lib, test, web, .gitignore, .metadata, .packages, analysis_options.yaml, first_app.iml, pubspec.lock, pubspec.yaml, README.md
- Run Without Debugging Ctrl+F5 (highlighted)
- Start Debugging F5
- Stop Debugging Shift+F5
- Restart Debugging Ctrl+Shift+F5
- Open Configurations
- Add Configuration...
- Step Over F10
- Step Into F11
- Step Out Shift+F11
- Continue F5
- Toggle Breakpoint F9
- New Breakpoint >
- Enable All Breakpoints
- Disable All Breakpoints
- Remove All Breakpoints
- Install Additional Debuggers...

pubspec.yaml content:

```
first_app
  description: A new Flutter project.

  # The following line prevents the package from being accidentally published to
  # pub.dev using `flutter pub publish`. This is preferred for private packages.
  publish_to: 'none' # Remove this line if you wish to publish to pub.dev

  # The following defines the version and build number for your application.
  # The version number is three numbers separated by dots, like 1.2.43
  # followed by an optional build number separated by a +.
  # Both the version and the builder number may be overridden in flutter
  # build by specifying --build-name and --build-number, respectively.
  # Android, build-name is used as versionName while build-number used as versionCode
  # and more about Android versioning at https://developer.android.com/studio/publish/versioning
  version: 1.0.0+1

  # In iOS, build-name is used as CFBundleShortVersionString while build-number is used as CFBundleVersion
  # Read more about iOS versioning at https://developer.apple.com/library/archive/documentation/General/Reference/InfoPlistKeyReference/Articles/AboutInformationPlistKeys.html#//apple\_ref/doc/uid/TP40009251-CH21-SW1
  environment:
    sdk: ">=2.12.0 <3.0.0"
```

Bottom status bar: Ln 1, Col 1 Spaces: 2 UTF-8 CRLF YAML Dart DevTools Flutter: 2.5.3 Pixel 2 API 29 (android-x86 emulator) ✓ Prettier

File Edit Selection View Go Run Terminal Help

main.dart - first_app - Visual Studio Code

EXPLORER

OPEN EDITORS

main.dart lib

FIRST_APP

.dart_tool .idea android build ios lib main.dart test web .gitignore .metadata .packages analysis_options.yaml first_app.iml pubspec.lock pubspec.yaml README.md

main.dart

lib > main.dart > ...

1 import 'package:flutter/material.dart';
2
Run | Debug | Profile
3 void main() {
4 runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8 const MyApp({Key? key}) : super(key: key);
9
// This widget is the root of your application.
11 @override
12 Widget build(BuildContext context) {

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

Launching lib\main.dart on Android SDK built for x86 in debug mode

10 // This widget is the root of your application.

Wednesday, Nov 17

10:59

Google Play Store Google Chrome

Running Gradle task 'assembleDebug'...

Ln 1, Col 1 Spaces: 2 CRLF Dart Dart DevTools Flutter: 2.5.3 Pixel 2 API 29 (android-x86 emulator) Prettier

File Edit Selection View Go Run Terminal Help main.dart - first_app - Visual Studio Code

EXPLORER OPEN EDITORS main.dart lib FIRST_APP .dart_tool .idea android build ios lib main.dart test web .gitignore .metadata .packages analysis_options.yaml first_app.iml pubspec.lock pubspec.yaml README.md

main.dart x lib > main.dart > ...

```
1 import 'package:flutter/material.dart';
2
3 void main() {
4   runApp(const MyApp());
5 }
6
7 class MyApp extends StatelessWidget {
8   const MyApp({Key? key}) : super(key: key);
9
10 // This widget is the root of your application.
11 @override
12 Widget build(BuildContext context) {
```

Run | Debug | Profile

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
Launching lib/main.dart on Android SDK built for x86 in debug mode...
✓ Built build/app/outputs/flutter-apk/app-debug.apk.
W/FlutterActivityAndFragmentDelegate( 5691): A splash screen was
recated. See flutter.dev/go/android-splash-migration for migration
Connecting to VM Service at ws://127.0.0.1:64414/SQCQpZbE_4I=/ws
D/eglCodecCommon( 5691): setVertexArrayObject: set vao to 0 (0) 1
D/skia    ( 5691): Shader compilation error
D/skia    ( 5691): -----
```

main.dart:1

Flutter Demo Home Page

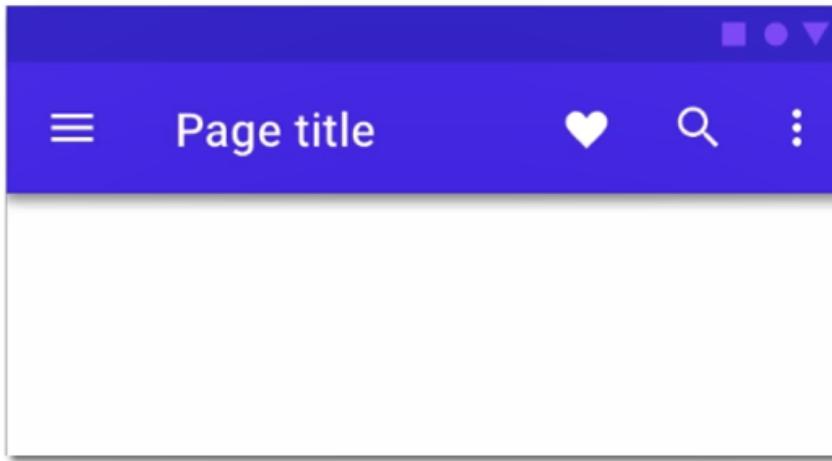
11:00 ⚡ 🌐 ⚡

You have pushed the button this many times:
0

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF Dart Dart DevTools Flutter: 2.5.3 Pixel 2 API 29 (android-x86 emulator) Prettier

FLUTTER AND MATERIAL DESIGN

Material Design Everywhere



Material is a **Design System** created (and heavily used) by Google

It's NOT Google's Style for Everyone! It is indeed **highly customizable** (and works on iOS devices, too)

Material Design is **built into Flutter** but you also find Apple-styled (Cupertino) widgets

FLUTTER ALTERNATIVES

Flutter vs React Native vs Ionic

Flutter

React Native

Ionic

Flutter vs React Native vs Ionic

Flutter

React Native

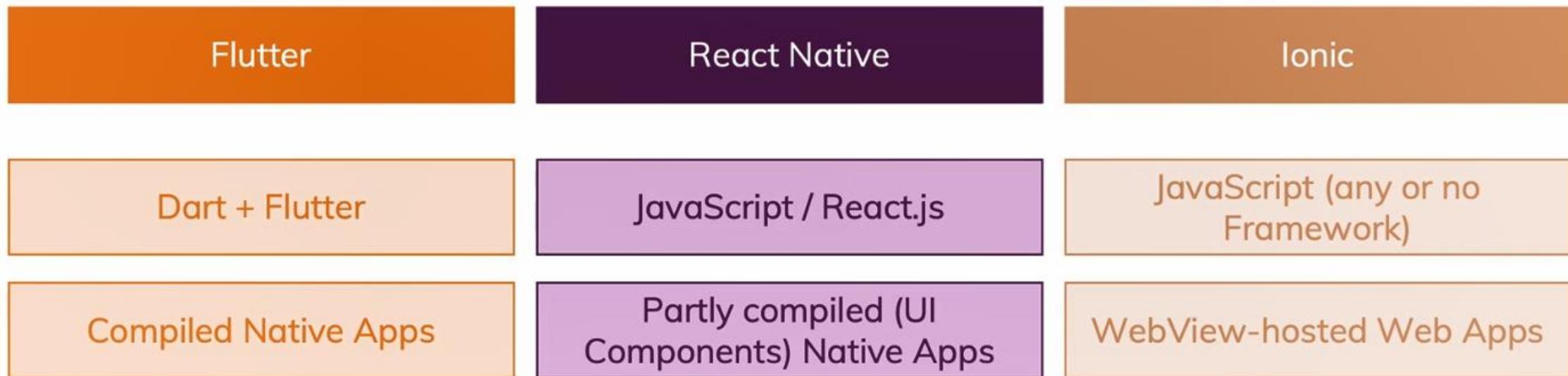
Ionic

Dart + Flutter

JavaScript / React.js

JavaScript (any or no
Framework)

Flutter vs React Native vs Ionic



Flutter vs React Native vs Ionic

Flutter	React Native	Ionic
Dart + Flutter	JavaScript / React.js	JavaScript (any or no Framework)
Compiled Native Apps	Partly compiled (UI Components) Native Apps	WebView-hosted Web Apps
Does NOT compile to iOS / Android UI Components	Does compile to iOS / Android UI Components	Does NOT compile to iOS / Android UI Components

Flutter vs React Native vs Ionic

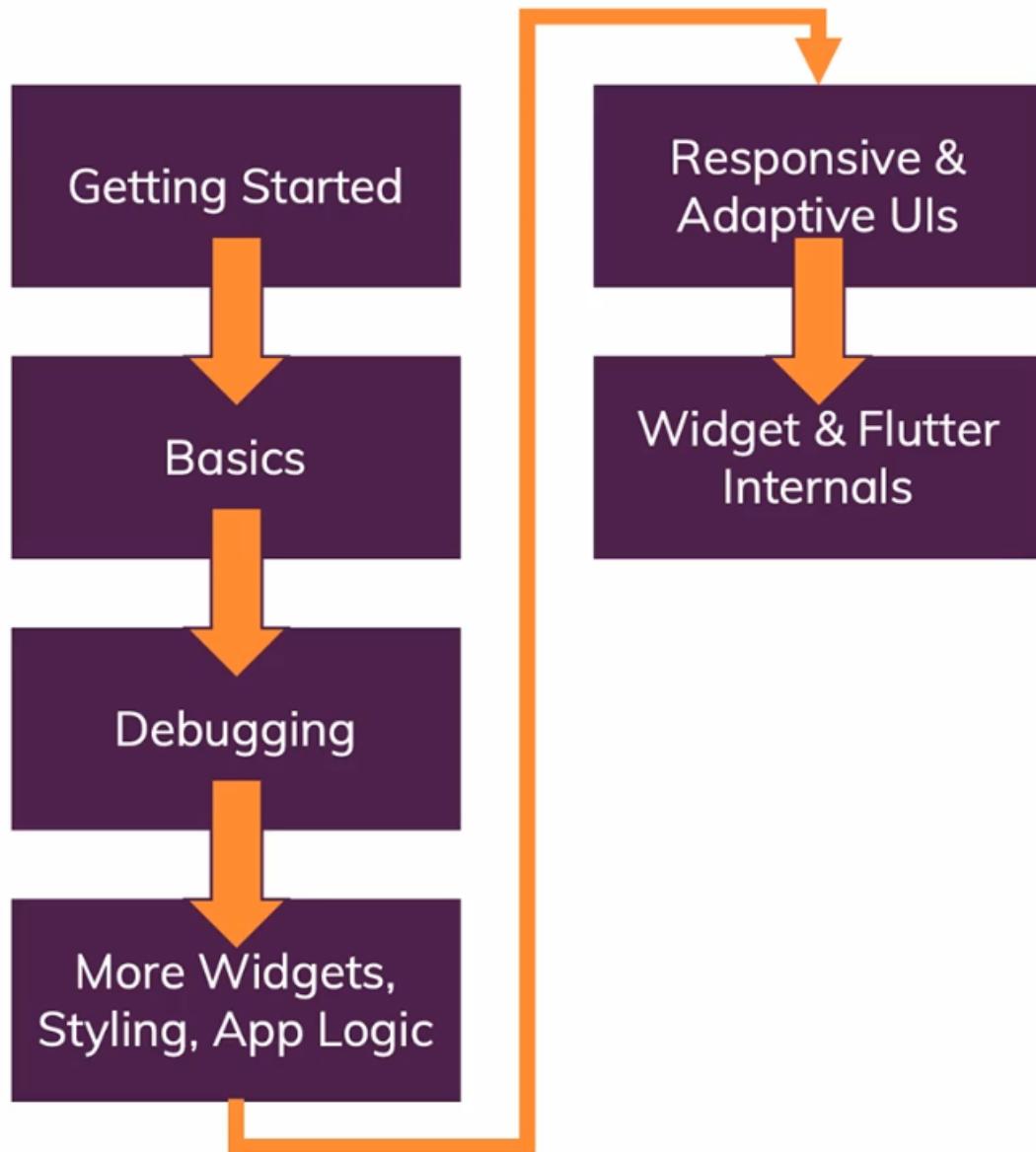
Flutter	React Native	Ionic
Dart + Flutter	JavaScript / React.js	JavaScript (any or no Framework)
Compiled Native Apps	Partly compiled (UI Components) Native Apps	WebView-hosted Web Apps
Does NOT compile to iOS / Android UI Components	Does compile to iOS / Android UI Components	Does NOT compile to iOS / Android UI Components
Cross-platform (mobile apps, web apps, desktop apps)	Mostly mobile apps (+ React Native Web)	Cross-platform (mobile apps, web apps, desktop apps)

Flutter vs React Native vs Ionic

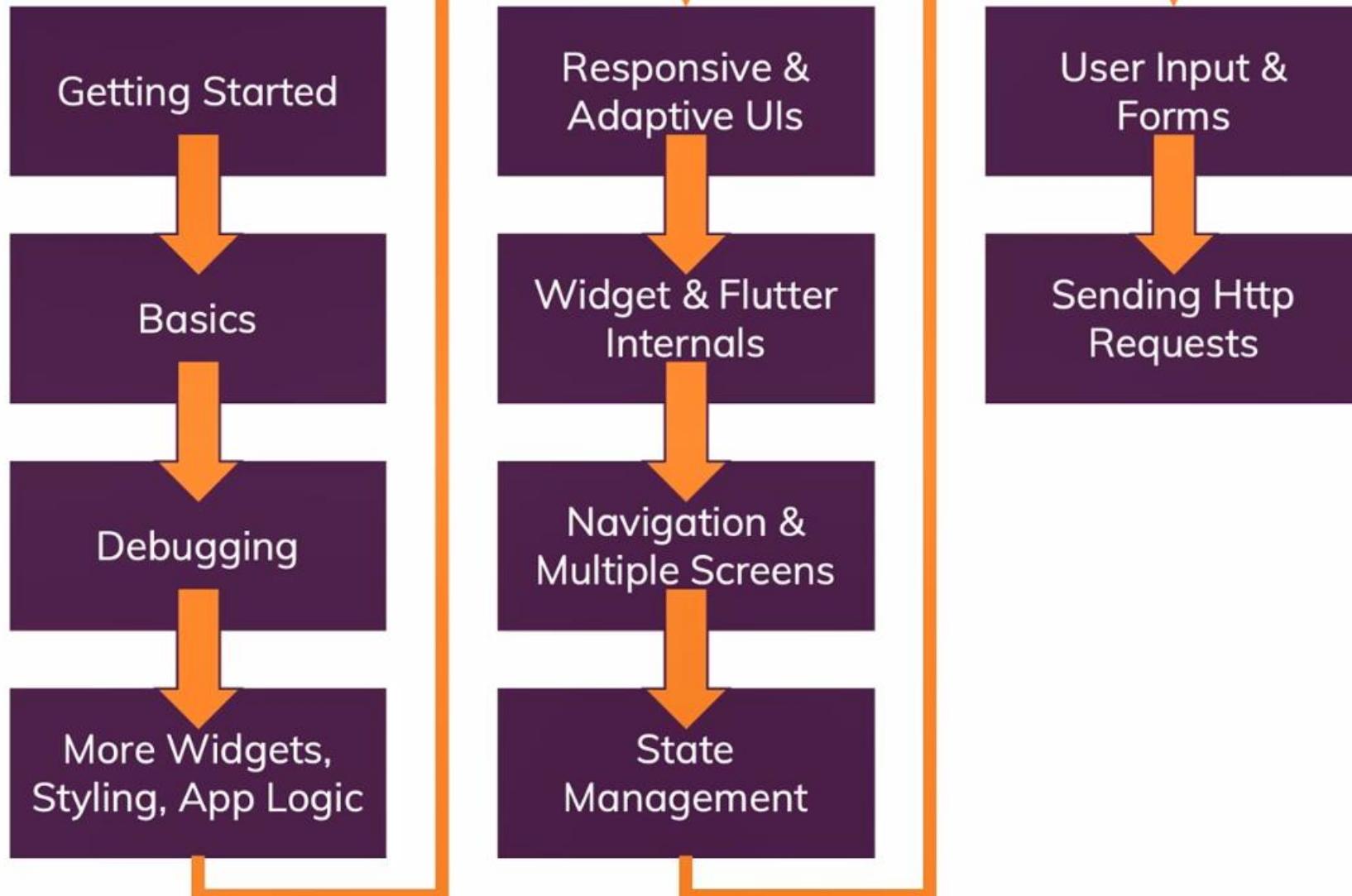
Flutter	React Native	Ionic
Dart + Flutter	JavaScript / React.js	JavaScript (any or no Framework)
Compiled Native Apps	Partly compiled (UI Components) Native Apps	WebView-hosted Web Apps
Does NOT compile to iOS / Android UI Components	Does compile to iOS / Android UI Components	Does NOT compile to iOS / Android UI Components
Cross-platform (mobile apps, web apps, desktop apps)	Mostly mobile apps (+ React Native Web)	Cross-platform (mobile apps, web apps, desktop apps)
Developed by Google	Developed by Facebook	Developed by Ionic

COURSE OUTLINE

Course Structure



Course Structure



Course Structure

