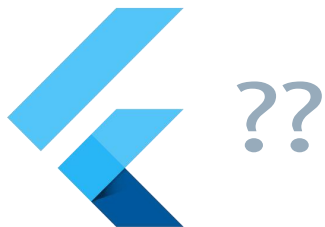


# Hello





UI toolkit for building beautiful,  
natively compiled applications for [mobile](#), [web](#), and [desktop](#)  
from a single codebase,  
written in Dart.

An open-source toolkit, made by 

# Beautiful

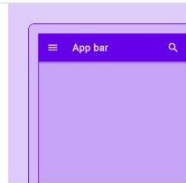
No compromises  
for UI design



- ✓ Control every pixel
- ✓ Never say no to your “designer”
- ✓ Rich palette of tools

# The power of widgets

Everything is a widget!!



AppBar

A Material Design app bar. An app bar consists of a toolbar and potentially other widgets, such as a TabBar and a FlexibleSpaceBar.

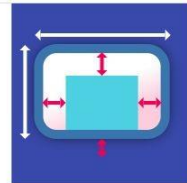
[Documentation](#)



Column

Layout a list of child widgets in the vertical direction.

[Documentation](#)



Container

A convenience widget that combines common painting, positioning, and sizing widgets.

[Documentation](#)



FlutterLogo

The Flutter logo, in widget form. This widget respects the IconTheme.

[Documentation](#)



Icon

A Material Design icon.

[Documentation](#)



Image

A widget that displays an image.

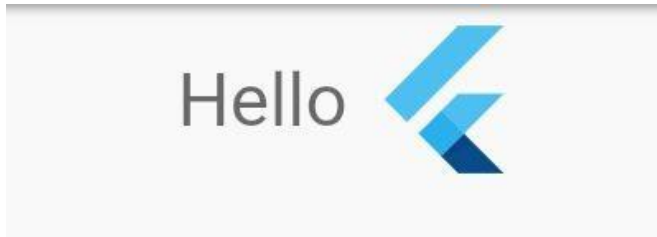
[Documentation](#)

# Declarative layout

Create layout by  
declaring intent

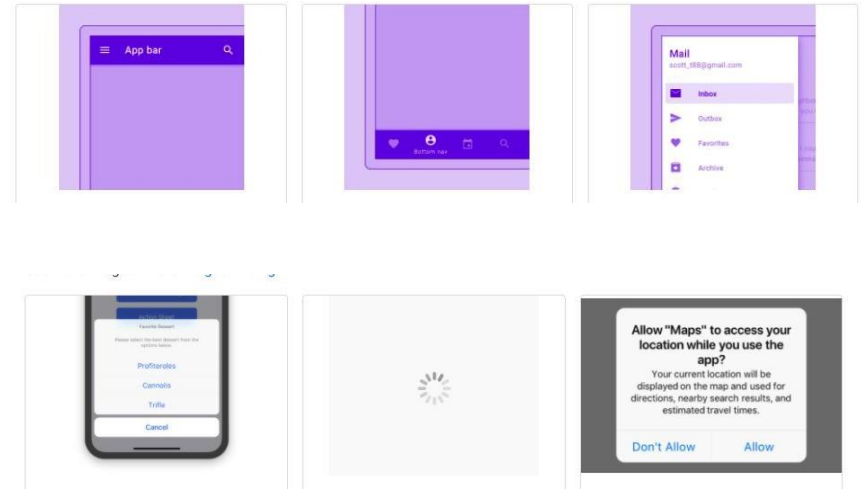
Flutter figures out  
the “how”

```
Row(  
  mainAxisAlignment: MainAxisAlignment.center,  
  crossAxisAlignment: CrossAxisAlignment.center,  
  children: [  
    Text(  
      'Hello',  
      style: Theme.of(context).textTheme.headline3),  
    Padding(  
      padding: const EdgeInsets.all(8.0),  
      child: FlutterLogo(size: 100),  
    ),  
  ],  
)
```



# Beautiful widgets

- Material Components widgets
- Cupertino (iOS-style) widgets



## Layout - related Widgets

- Single-child layout widgets  
(Container, Center, Expanded, Padding)
- Multi-child layout widgets  
(Column, Row, ListView, Stack)

## Structural Widgets

(Buttons, Icons, Text, Image)

## Animation & motion Widgets:

## Styling widgets

## Scrolling widgets

.....

## Material component Widgets

## Cupertino widgets

### Accessibility

Make your app accessible.

[Visit](#)

### Animation and Motion

Bring animations to your app.

[Visit](#)

### Assets, Images, and Icons

Manage assets, display images, and show icons.

[Visit](#)

### Async

Async patterns to your Flutter application.

[Visit](#)

### Basics

Widgets you absolutely need to know before building your first Flutter app.

[Visit](#)

### Cupertino (iOS-style widgets)

Beautiful and high-fidelity widgets for current iOS design language.

[Visit](#)

### Input

Take user input in addition to input widgets in Material Components and Cupertino.

[Visit](#)

### Interaction Models

Respond to touch events and route users to different views.

[Visit](#)

### Layout

Arrange other widgets columns, rows, grids, and many other layouts.

[Visit](#)

### Material Components

Visual, behavioral, and motion-rich widgets implementing the [Material Design](#) guidelines.

[Visit](#)

### Painting and effects

These widgets apply visual effects to the children without changing their layout, size, or position.

[Visit](#)

### Scrolling

Scroll multiple widgets as children of the parent.

[Visit](#)

### Styling

Manage the theme of your app, makes your app responsive to screen sizes, or add padding.

[Visit](#)

### Text

Display and style text.

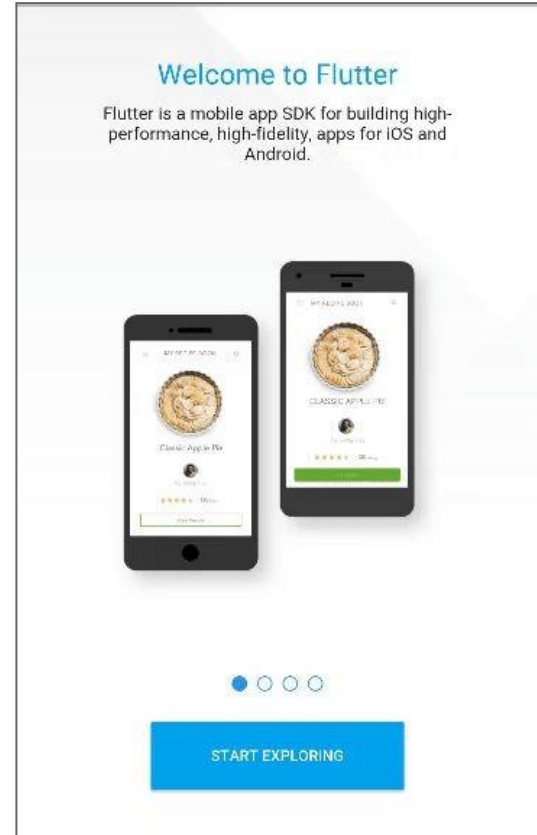
[Visit](#)



# Fast

High velocity  
development

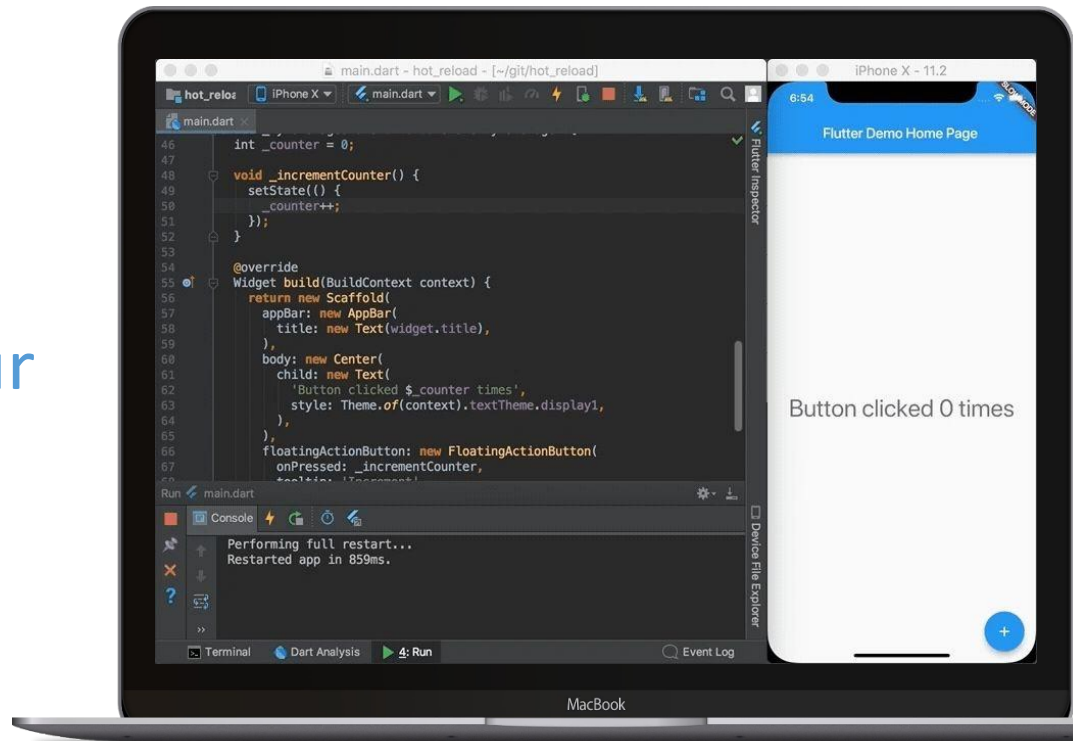
+ Native  
performance



- ✓ Graphics engine Skia 2D
- ✓ 60fps, GPU accelerated
- ✓ Compiled to native machine code

Productive

Develop while your  
app is running



- ✓ Paint your app to life
- ✓ Hot Reload & Hot Restart
- ✓ Revolutionary capabilities for developers and designers

# One language

**Dart Since  
2011**

made by **Google**





# Dart

- ✓ Client -optimized OOP language for fast apps in any platform
- ✓ Dart js compiler
- ✓ Just in time and Ahead of time compilation



- ✓ Stateful hot reload
- ✓ Syntax simplicity
- ✓ 'to -native' compilation

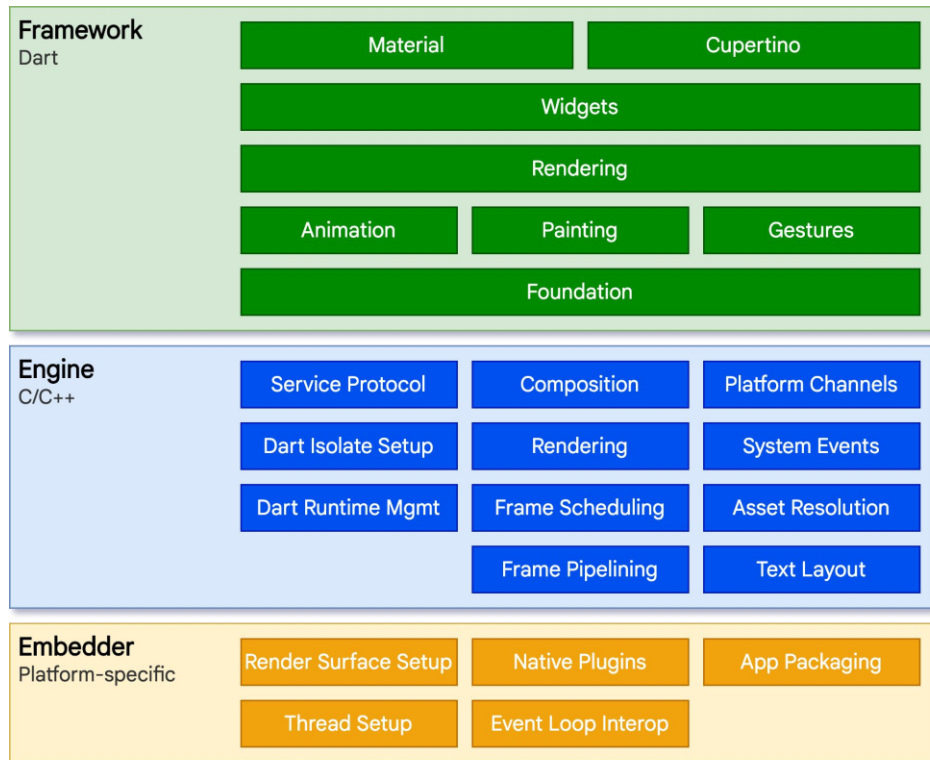
Why

is Unique?



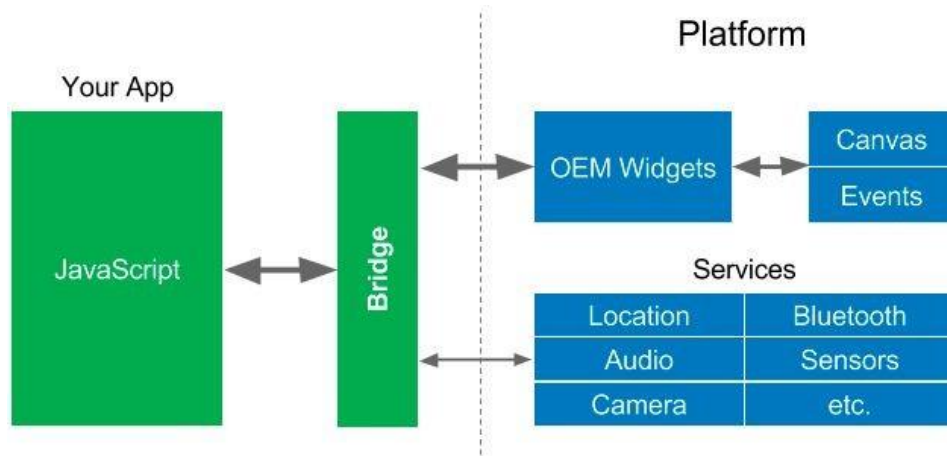


# Architecture overview

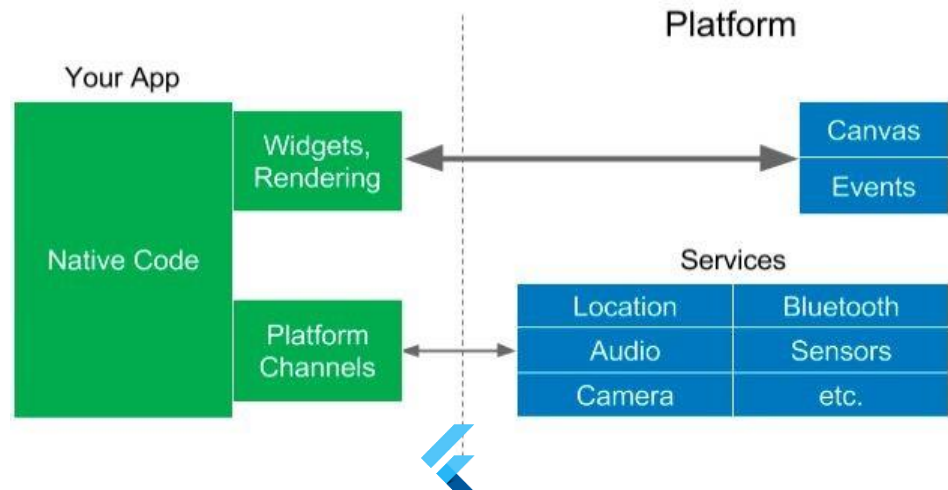


# Web

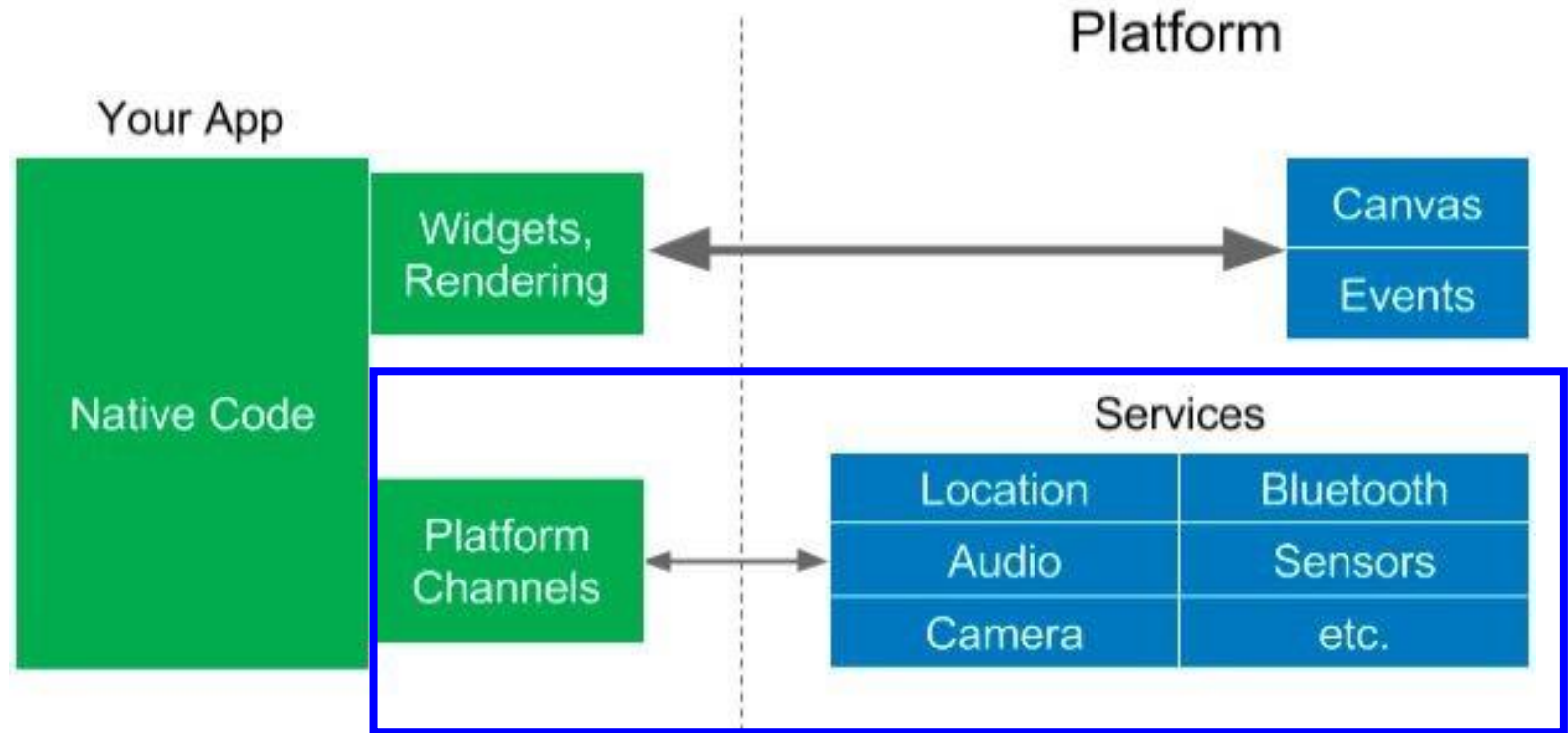




VS



# Plugins/Packages





# pub.dev

🔍 Search packages

Find and use packages to build [Dart](#) and [Flutter](#) apps.

## Flutter Favorites

Packages that demonstrate the [highest levels of quality](#), selected by the Flutter Ecosystem Committee

### [url\\_launcher](#)

Flutter plugin for launching a URL on Android and iOS. Supports web, phone, SMS, and

### [built\\_value\\_generator](#)

Value types with builders, Dart classes as enums, and serialization. This library is the

### [bloc](#)

A predictable state management library that helps implement the BLoC (Business

### [font\\_awesome\\_flutter](#)

The Font Awesome Icon pack available as Flutter Icons. Provides 1500 additional icons

Why

is Awesome?




# Set up


Set up an editor >


Install


Docs > Get started > Install

Select the operating system on which you are installing Flutter:

  
Windows

  
macOS

  
Linux

  
Chrome OS

# IDE Support



[flutter.dev/docs/get-started/editor](https://flutter.dev/docs/get-started/editor)



# dartpad.dev

The screenshot displays the DartPad web interface, which is used for writing and running Dart code. The interface is split into two main sections: a code editor on the left and a preview window on the right.

**Code Editor (Left):**

- Toolbar:** Includes icons for "New Pad", "Reset", "Format", and "Install SDK".
- Code:** The code is for a Flutter counter app. It includes a `setState` function to increment a counter, an `@override` `build` method that returns a `Scaffold` with an `AppBar` and a `body` containing a `Text` widget and a `FloatingActionButton` that calls `_incrementCounter`.
- Buttons:** A blue "RUN" button is located to the right of the code editor.
- Footer:** Links for "Privacy notice" and "Send feedback" are visible.

**Preview Window (Right):**

- Title:** "Counter app in Flutter".
- Header:** "Flutter Demo Home Page".
- Content:** The preview shows the rendered app. It displays the text "You have pushed the button this many times:" followed by a large number "0".
- Interaction:** A blue circular button with a white "+" icon is visible in the bottom right corner of the preview area, representing the floating action button from the code.
- Status Bar:** At the bottom right, it says "no issues Based on Dart SDK 2.8.4".

# codepen.io

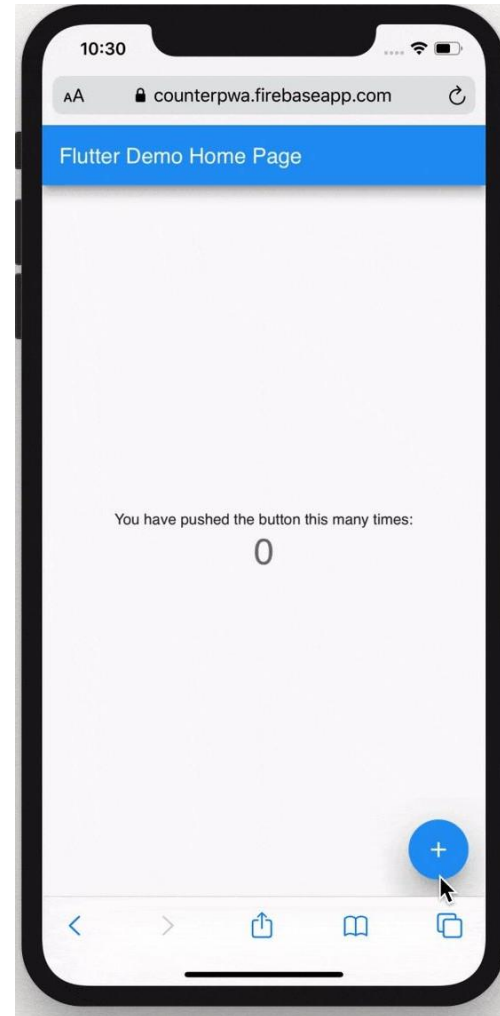
The screenshot shows the CodePen editor interface. The top bar includes the file name 'Untitled' with a small icon, and buttons for 'Save', 'Settings', 'Change View', and a user profile icon. The left sidebar shows a file explorer with a folder named 'Flutter'. The main editor area contains the following Dart code:

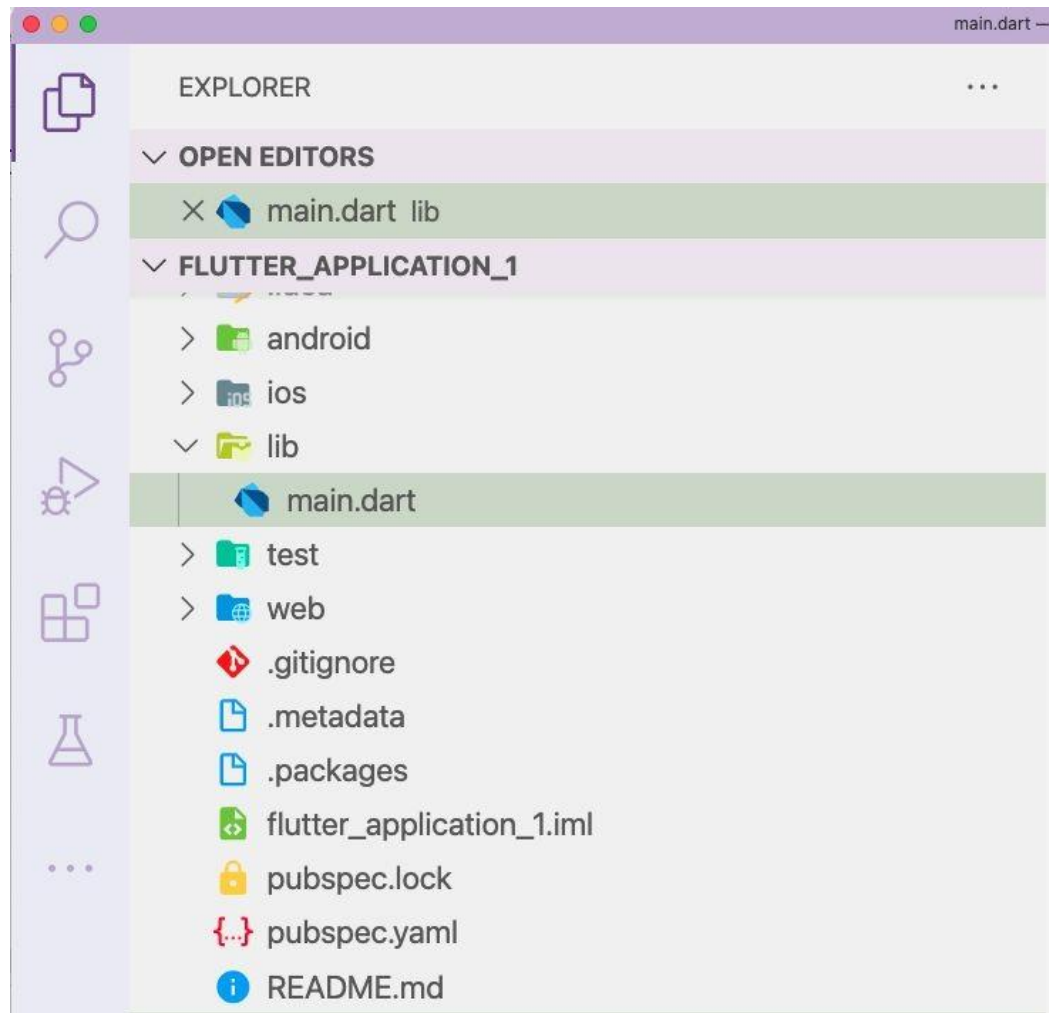
```
1 import
  'package:flutter/material.dart';
2
3 void main() {
4   runApp(
5     MaterialApp(
6       home: MyWidget(),
7     ),
8   );
9 }
10
11 class MyWidget extends
  StatelessWidget {
12   @override
13   Widget build(BuildContext
    context) {
14     return Scaffold(
15       body: Center(
16         child: Text(
17           'Build some widgets!',
18           style:
19             Theme.of(context).textTheme.headlin
20             e4,
21         ),
22       ),
23     );
24   }
25 }
```

The right side of the editor is a large, empty white space with the text 'Build some widgets!' centered in a large, light gray font. A red 'DEBUG' button is visible in the top right corner of the editor area. At the bottom, there is a 'Console' tab and a 'Assets' tab. A small 'PRO' badge is visible in the bottom left corner, and a banner at the bottom center reads 'CodePen: Unlock all of CodePen'.

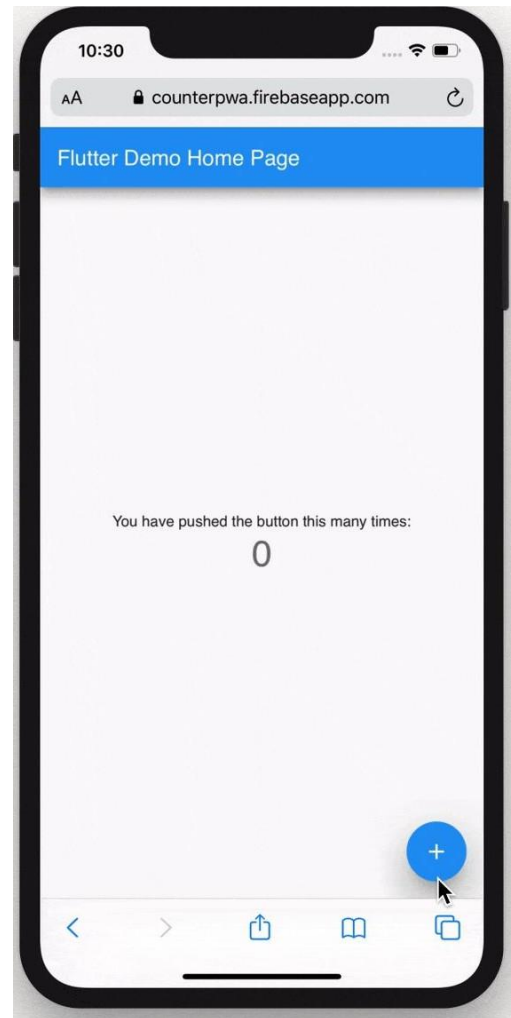


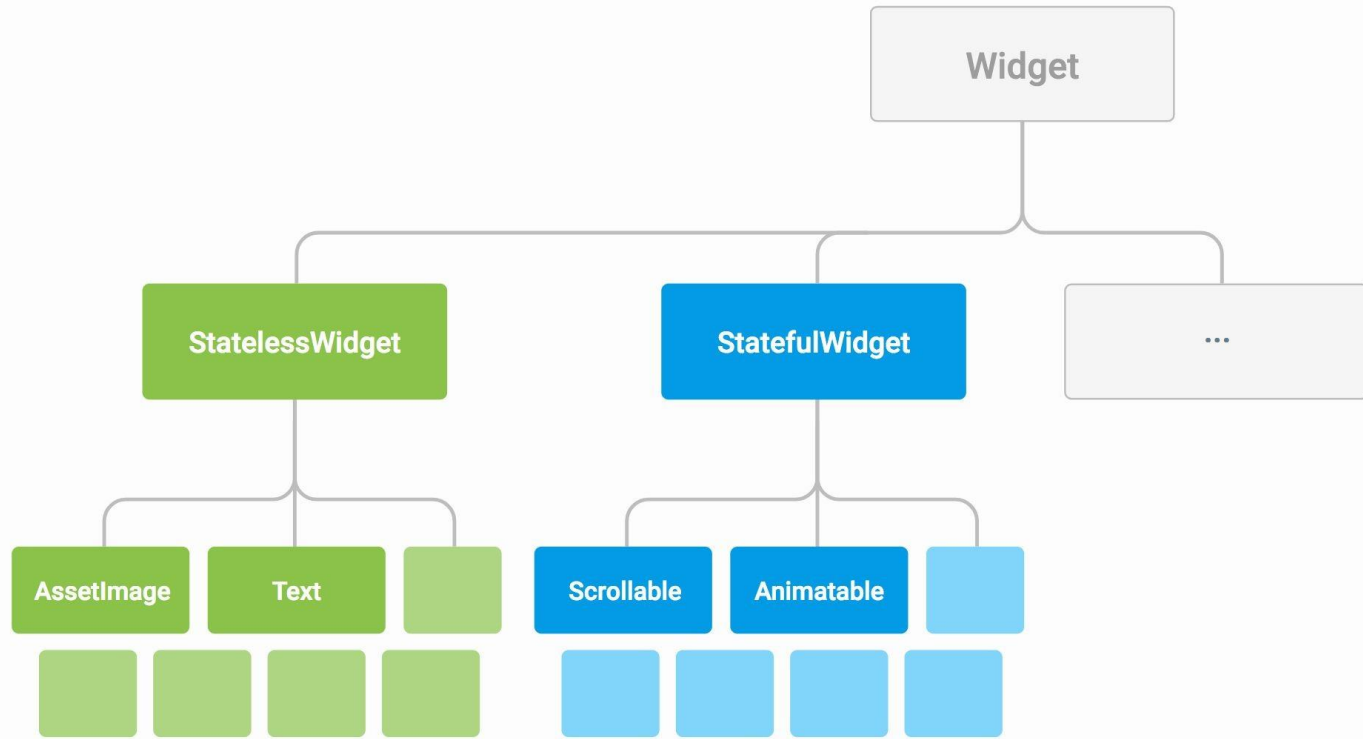
app





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





# Stateless widget

Input Data

Data can change  
(externally)

Widget

Renders UI

Gets (re)- rendered  
when  
Input Data changes

```
class MyApp extends StatelessWidget {  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Demo',  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
      ),  
      home: MyHomePage(title: 'Flutter Demo Home Page'),  
    );  
  }  
}
```



# Stateful widget

Input Data

Data can change  
(externally)

Widget

Internal State

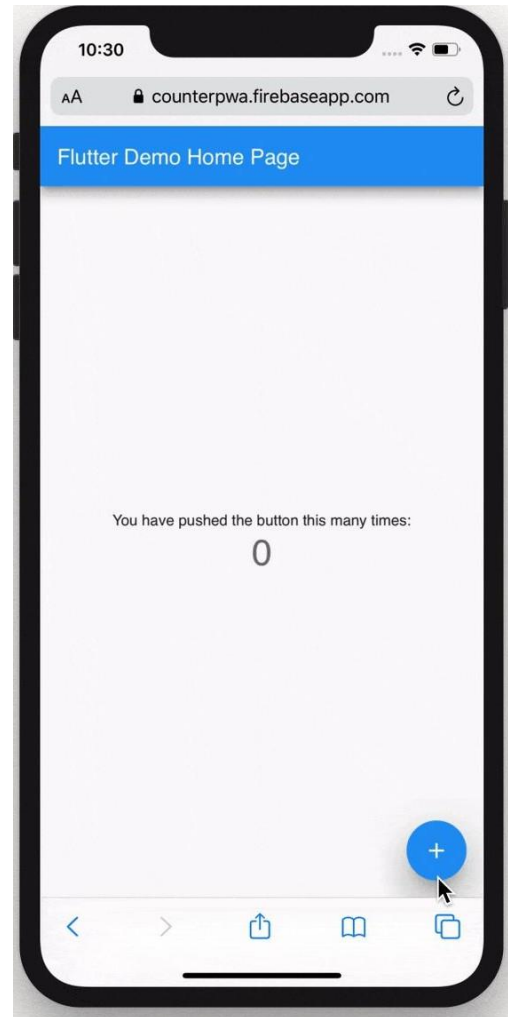
Renders UI

Gets (re)- rendered  
when  
Input Data  
or local State  
changes

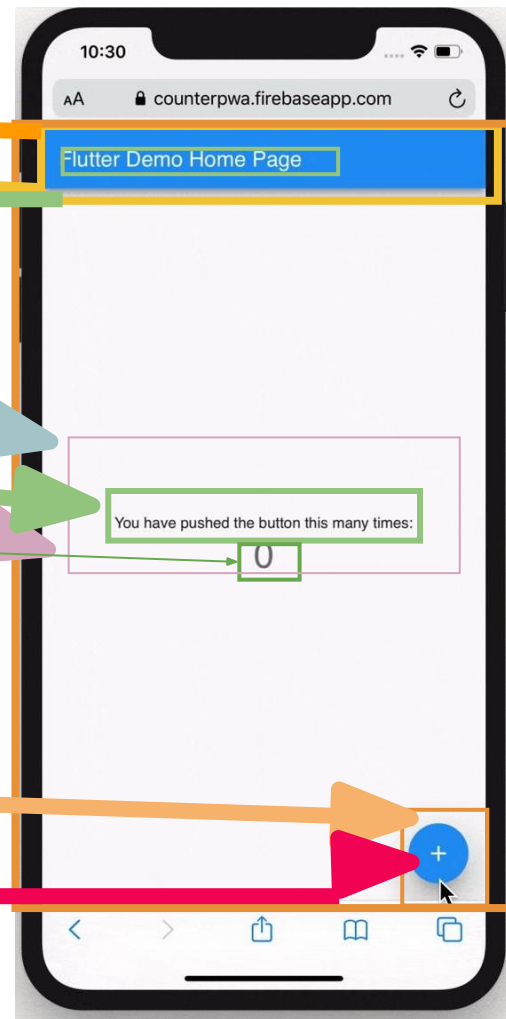
```
class MyHomePage extends StatefulWidget {  
    MyHomePage({Key key, this.title}) : super(key: key);  
  
    final String title;  
  
    @override  
    _MyHomePageState createState() => _MyHomePageState();  
}
```

```
class _MyHomePageState extends State<MyHomePage>{  
  int _counter = 0;  
  
  void _incrementCounter() {  
    setState(() {  
      _counter++;  
    });  
  }  
}
```

•  
•  
•



```
Widget build(BuildContext context) {  
  return Scaffold(  
    appBar: AppBar(title: Text(widget.title)),  
    body: Center(  
      child: Column(  
        mainAxisAlignment: MainAxisAlignment.center,  
        children: <Widget>[  
          Text(  
            'You have pushed the button this many times:',  
          ),  
          Text( '$_counter',  
            style: Theme.of(context).textTheme.headline4,  
          ),  
        ],  
      )),  
    floatingActionButton: FloatingActionButton(  
      onPressed: _incrementCounter,  
      tooltip: 'Increment',  
      child: Icon(Icons.add),  
    ), );  
}
```





# Demo

**BETWEEN FLUTTER AND DART, WHICH IS THE PROGRAMMING  
LANGUAGE AND WHICH IS THE FRAMEWORK?**



# **WHAT IS A WIDGET IN FLUTTER?**



**HOW MANY TYPES OF WIDGETS ARE THERE IN FLUTTER?**






# **WHAT ARE THE ADVANTAGES OF FLUTTER?**



# **WHAT IS HOT RELOAD IN FLUTTER?**



**DURING A TYPICAL DEVELOPMENT CYCLE, YOU TEST AN APP USING FLUTTER RUN AT THE COMMAND LINE, OR BY USING THE RUN AND DEBUG OPTIONS IN YOUR IDE. BY DEFAULT, FLUTTER BUILDS A DEBUG VERSION OF YOUR APP. QUESTION, WHAT ARE THE DIFFERENT BUILD MODES IN FLUTTER?**



**WHAT ARE THE BENEFITS OF FLUTTER?**

# **WHAT ARE PACKAGES IN FLUTTER?**

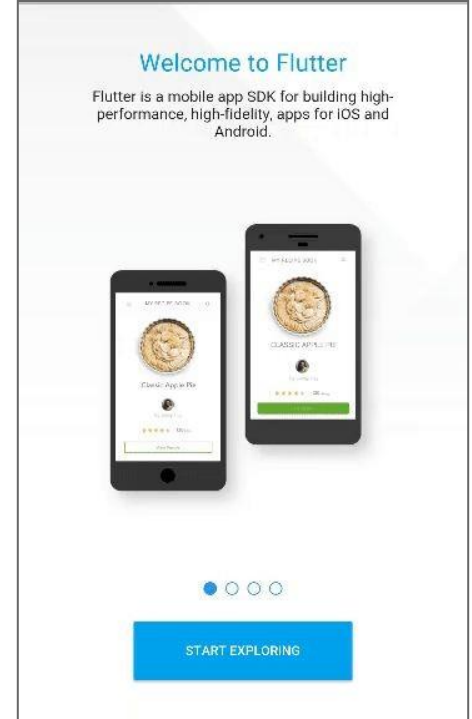


# Recap





# Great looking and fast Apps





One codebase  
on every screen







`flutter build android [--release]`



`flutter build ios [--release]`



`flutter build linux [--release]`



`flutter build macos [--release]`



`flutter build windows [--release]`



`flutter build web [--release]`

Getting  
started with



▼ 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

[flutter.dev/docs/get-started/flutter-for/android-devs](https://flutter.dev/docs/get-started/flutter-for/android-devs)

# Newbie

- [Checkout examples](#)
- [Watch Flutter videos/courses](#)
- [Read Flutter documentations](#)
- [Play and practice](#)
- Find a friend to learn with

- Dart code lab
- Tour of dart
- Dart video tutorials



# Flutter docs / blogs

- [flutter.dev](https://flutter.dev)
- [flutter.faq](https://flutter.faq)
- [flutter.community](https://flutter.community)
- [flutter.medium](https://flutter.medium)



- [flutter.showcases](https://flutter.showcases)
- [codepen.io/flutter](https://codepen.io/flutter)
- [itsallwidgets.com](https://itsallwidgets.com)
- [latest codelabs](https://latest.codelabs.dev)
- [fluttergems.dev](https://fluttergems.dev)



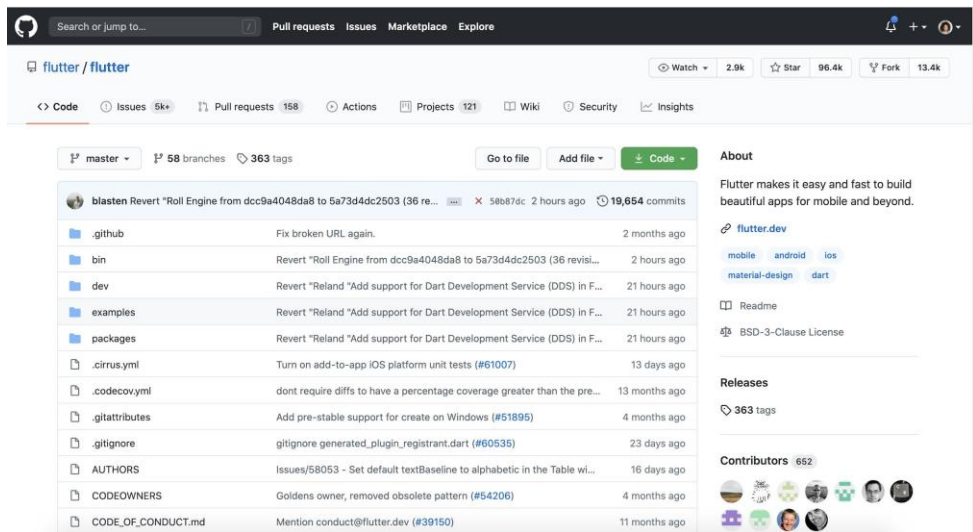
Coding GIF by Memecandy from giphy.com

- [The Flutter boring show](#)
- [Widget of the week](#)
- [Flutter in focus](#)





Open  
Everything is free  
100% community  
contribution



github.com/flutter/

# Welcome to the Flutter community

Below you'll find ways to get involved in the Flutter developer community as well as links to resources that can help answer your questions. If you're in China, read [Using Flutter in China](#).

## GitHub

Tell us what you want to see

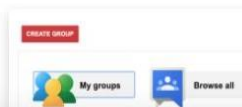
An easy way to send feedback is to "thumbs up" issues important to you on GitHub

[See the issue tracker](#)



Google Groups

Google Groups is a great way to stay up to date



flutter.dev/community



**World-wide community  
of people whose gender  
identity is either female  
or non-binary**

[flutteristas.org](https://flutteristas.org)

**IS FLUTTER LIMITED TO THE MOBILE WORLD ONLY?**

**(FACT)**

**FLUTTER IS AN OPEN-SOURCE USER INTERFACE SOFTWARE DEVELOPMENT KIT WHICH MEANS THERE ARE SOME ADVANTAGES TO UTILIZING AN OPEN SOURCE SDK LIKE TRANSPARENCY – AN OPEN SOURCE CODE MEANS THE CODE IS FULLY AVAILABLE FOR THE WORLD TO SEE, AND (IN MOST CASES) FOR ANYONE TO CONTRIBUTE. EVERY LIBRARY HAS OWNERS WHO CONTROL FINAL EDITS, BUT ANYONE IN THE DEVELOPER COMMUNITY CAN MAKE SUGGESTIONS AND TAKE PART.**



**(FACT)**

**SOME COMPANIES THAT USE FLUTTER: GOOGLE, EBAY, TENCENT, THE NEW YORK TIMES, BMW, ETC.**

## **[TRIVIA]**

**DO YOU KNOW THAT IT TOOK ALMOST 9 YEARS BEFORE GOOGLE RELEASED DART'S FIRST STABLE RELEASE? DART WAS FIRST APPEARED IN OCTOBER 10, 2011 AND THE FIRST STABLE RELEASE WAS RELEASED ON MARCH 23, 2020.**

# Any questions?

