Flutter Introduction

by Radhen Soni

What is Flutter?

Flutter is an open-source UI software development kit created by Google. It is used to develop cross platform applications from a single codebase for any web browser, Fuchsia, Android, iOS, Linux, macOS, and Windows. Flutter apps are written in the Dart language.

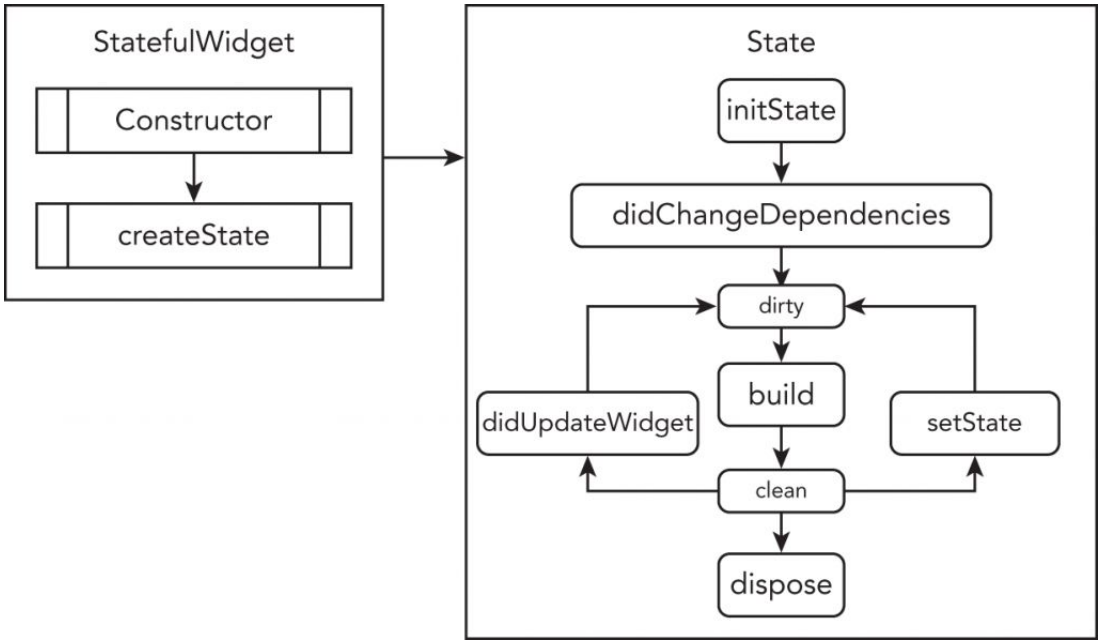
What is the advantage of Flutter?

1. Cross-platform Operations: Apps made with flutter can be operated on both the platform (iOS and Android). There is no need for reconfiguration and redesigning.

2. Time Constraint: The time required to launch the app into the market, also reduces as only a single app has to be made, which would work independently of the platform.

3. Powerful Design: Flutter mobile framework is the latest in the market, and this helps to create a very powerful app design with minimum efforts.

Flutter life cycle



Stateless Widgets

Stateless Widgets are those widgets that don’t need to deal with the State as they don’t change powerfully at runtime. It becomes permanent like on variables, buttons, symbols, and so forth, or any express that can’t be changed on the application to recover information.

Stateful Widgets

Stateful Widgets are those widgets that hold the State and the UI being portrayed can change progressively at runtime. It is a changeable widget, so it is attracted numerous times during its lifetime.

Appbar widget

- Flutter AppBar class is used to display a material design app bar with optional actions (IconButtons).
- AppBar is typically displayed as a fixed-height widget at the top of the screen.
- Flutter AppBar widget has properties like actions, backgroundColor, primary,title, etc., that help us to enhance the look or functionality of the AppBar.

Container widget

- Flutter Container widget center aligns its child widget within itself, along vertical and horizontal axes.

- It have helpful properties like, shadow, border, border radius, margin, padding & color.

Align widget

- Flutter To align a child widget within its parent you use the Align widget. If you know how to use the Center widget then you are the right track because Center is just a special case of Align.
- Wrap the widget you wish to align with the Align widget and set its alignment property. Below are the properties of align.

Alignment.topLeft, Alignment.topCenter, Alignment.topRight,
Alignment.centerLeft, Alignment.center, Alignment.centerRight,
Alignment.bottomLeft, Alignment.bottomCenter, Alignment.bottomRight

Column widget

Column widget is used to display its widgets in a vertical widget array.

We shall display a Column widget with four children widgets of which two are Text widgets and two are Icon widgets. We shall wrap the Column widget in a Center widget to display the column at the centre of the screen along horizontal axis.

Row widget

Row widget is used to display its widgets in a horizontal array.

We shall display a Row widget with four children widgets of which two are Text widgets and two are Icon widgets.

Button widget

Flutter TextButton class is used to display a material design Text Button that has no borders by default. TextButton is generally used on toolbars, dialogs, inline with text or other content, etc.

By default, when a TextButton button is pressed, ripple effect is shown.

Elevated button has also same feature as text button.

Icon widget

Icons can be used as a representative symbol for a quick understanding of the functionality, or path of the navigation, etc.

You can set color, size of the icon.

Along with that we need to discuss about IconButton

Flutter IconButton acts just like a button, but with an icon instead of an usual button. We can set icon with clickable button.

Radio widget

Flutter Radio widget is used to select single option of give all. We display a material design radio button.

TextField widget

TextField is used to get text input from user. The default behavior of TextField is that, when you press on it, it gets focus and a keyboard slides from the bottom of the screen. When you enter text using keyboard, the string is displayed in the TextField.

You can access the value entered in the TextField by attaching a TextEditingController to the TextField controller. Or you may also access the value using onChanged() function of TextField.

Image widget

Flutter Image widget displays an image in our Flutter Application. We can display image in 3 ways.

Assets - by file

Network - by url

Memory - by uint8List

DataTable widget

A DataTable is a material design used to display data on a table or in rows and columns. A Data table is used to show the data which have columns and rows as child, a Column is used to set the name of the column, and a Row is used to set the values of the columns.

GridView widget

GridView displays its children widgets as a grid like phone’s gallery.

The default main axis is vertical axis, and the cross axis is horizontal axis. Using gridDelegate property, we have set the number of widgets to be arranged in the cross axis to 3.

ListView widget

Flutter ListView widget displays a set of items in a scrollable list. Pass the list of items to the children property of ListView widget, and you have a ListView in your Flutter application.

You can make the ListView scroll in either of the directions: horizontal, vertical. using scrollDirection property. Or you can also make it never scroll using physics property set to NeverScrollableScrollPhysics() object.

Slider widget

A slider in Flutter is a material design widget used for selecting a range of values. It is an input widget where we can set a range of values by dragging or pressing on the desired position.

RangeSlider widget

Flutter RangeSlider widget is used to select a range from a range of values.

In the following example, the values, and onChanged properties are set. onChanged property takes a function. The function receives the current selected range of the slider as argument.

WebView widget

Flutter WebView widget displays a browser like space to show the webpage specified by URL. So, you can display a webpage just like another widget in your mobile application.

TabBar widget

We will learn how to display a horizontal row of tabs and display a widget that corresponds to the currently selected tab.

To display a horizontal row of tabs, we can use TabBar widget.

To display a widget that corresponds to the currently selected tab, we can use TabBarView page view.When we use TabBar and TabBarView.

Animation widget

We will learn how to animate color of a widget, i.e., transitioning from a starting color to ending color.You can animate color of a widget using ColorTween, & Tween animation through which we can animate size of widget.