**Week 3: Widgets & Layouts**

In this week I have learned about different widgets and layouts in flutter.

**Widgets**

Widgets are classes used to build UIs. Widgets are used for both layouts and UI elemnts.

1. Container Widget
2. Row Widget
3. Stack Widget
4. Expanded Widget
5. Circular Widget
6. Divider Widgets
7. Text Field
8. List Tile Widget
9. **Container Widget**

The Container widget lets you create a rectangular visual element. A container can be decorated with a BoxDecoration, such as a background, a border, or a shadow. A Container can also have margins, padding, and constraints applied to its size. In addition, a Container can be transformed in three dimensional space using a matrix.

1. **Row Widget**

Row widget displays its children in a horizontal array.

To cause a child to expand to fill the available horizontal space, wrap the child in an [Expanded](https://api.flutter.dev/flutter/widgets/Expanded-class.html) widget.

The [Row](https://api.flutter.dev/flutter/widgets/Row-class.html) widget does not scroll (and in general it is considered an error to have more children in a [Row](https://api.flutter.dev/flutter/widgets/Row-class.html) than will fit in the available room). If you have a line of widgets and want them to be able to scroll if there is insufficient room, consider using a [ListView](https://api.flutter.dev/flutter/widgets/ListView-class.html).

1. **Stack Widget**

Instead of being linearly oriented (either horizontally or vertically), a Stack widget lets you place widgets on top of each other in paint order. You can then use the [Positioned](https://api.flutter.dev/flutter/widgets/Positioned-class.html) widget on children of a Stack to position them relative to the top, right, bottom, or left edge of the stack. Stacks are based on the web’s absolute positioning layout model.

1. **Expanded Widget**

Expanded widget expands a child of a [Row](https://api.flutter.dev/flutter/widgets/Row-class.html), [Column](https://api.flutter.dev/flutter/widgets/Column-class.html), or [Flex](https://api.flutter.dev/flutter/widgets/Flex-class.html) so that the child fills the available space.

Using an [Expanded](https://api.flutter.dev/flutter/widgets/Expanded-class.html) widget makes a child of a [Row](https://api.flutter.dev/flutter/widgets/Row-class.html), [Column](https://api.flutter.dev/flutter/widgets/Column-class.html), or [Flex](https://api.flutter.dev/flutter/widgets/Flex-class.html) expand to fill the available space along the main axis (e.g., horizontally for a [Row](https://api.flutter.dev/flutter/widgets/Row-class.html) or vertically for a [Column](https://api.flutter.dev/flutter/widgets/Column-class.html)). If multiple children are expanded, the available space is divided among them according to the [flex](https://api.flutter.dev/flutter/widgets/Flexible/flex.html) factor.

An [Expanded](https://api.flutter.dev/flutter/widgets/Expanded-class.html) widget must be a descendant of a [Row](https://api.flutter.dev/flutter/widgets/Row-class.html), [Column](https://api.flutter.dev/flutter/widgets/Column-class.html), or [Flex](https://api.flutter.dev/flutter/widgets/Flex-class.html), and the path from the [Expanded](https://api.flutter.dev/flutter/widgets/Expanded-class.html) widget to its enclosing [Row](https://api.flutter.dev/flutter/widgets/Row-class.html), [Column](https://api.flutter.dev/flutter/widgets/Column-class.html), or [Flex](https://api.flutter.dev/flutter/widgets/Flex-class.html) must contain only [StatelessWidget](https://api.flutter.dev/flutter/widgets/StatelessWidget-class.html)s or [StatefulWidget](https://api.flutter.dev/flutter/widgets/StatefulWidget-class.html)s (not other kinds of widgets, like [RenderObjectWidget](https://api.flutter.dev/flutter/widgets/RenderObjectWidget-class.html)s).

1. **Circular Avatar Widget**

Circular avatar widget is used to create a Circular Image in Flutter. CircleAvatar is simply a widget used to display a user profile picture. In the absence of a user's profile picture, Circle Avatar can display the user's initials.

1. **Divider Widget**

A thin horizontal line, with padding on either side. In the Material Design language, this represents a divider. Dividers can be used in lists, [Drawer](https://api.flutter.dev/flutter/material/Drawer-class.html)s, and elsewhere to separate content.

1. **Text Form Field Widget**

TextField in Flutter is the most commonly used text input widget that allows users to collect inputs from the keyboard into an app. We can use the TextField widget in building forms, sending messages, creating search experiences, and many more. By default, Flutter decorated the TextField with an underline.

1. **List Tile Widget**

It is mainly used to populate the scrollable views such as ListView, Column, and Row. For example, you can use the ListTile to show a list of to-do items, emails, navigation options, and more.

**Layouts**

In Flutter, a layout refers to the way widgets are arranged on the screen. Layouts are used to define the structure of a user interface by specifying the position and size of each widget in relation to other widgets.

Flutter provides a wide range of layout widgets that developers can use to create responsive and dynamic user interfaces. These widgets include:

1. Container: A widget that allows developers to specify a specific width and height for a child widget.
2. Row and Column: Widgets that display their children in a horizontal or vertical line.
3. Stack: A widget that overlays its children on top of each other.
4. Expanded: A widget that expands its child to fill the available space.
5. GridView: A widget that arranges its children in a grid.
6. ListView: A widget that displays a scrollable list of children.
7. Wrap: A widget that displays its children in a flow layout.
8. Flex: A widget that allows developers to create flexible layouts that can adapt to different screen sizes and orientations.

Using these layout widgets, developers can create complex user interfaces with ease, while ensuring that the UI is responsive and adapts to different screen sizes and orientations.