Experiment 2: Exploring Flutter Widgets

Aim:To design Flutter UI by including common widgets.

Theory:

In Flutter, widgets are the building blocks of the UI. Widgets can be either stateless or stateful and can be combined to create complex UIs.

● Flutter Scaffold: Scaffold is a basic structure of the visual interface of an app. It provides functionality like an app bar, a drawer, and a bottom navigation bar.

● Flutter Container: Container is a box model that allows you to create a box with specific dimensions, padding, margin, and decoration.

● Flutter Row & Column: Row and Column are flex widgets used for arranging child widgets horizontally (Row) or vertically (Column).

● Flutter Text: Text widget is used for displaying a short piece of text. It supports styling and formatting.

● Flutter TextField: TextField is an input widget that allows users to enter text.

● Flutter Buttons: Flutter provides various button widgets like ElevatedButton, TextButton, and OutlinedButton for user interaction.

● Flutter Stack: Stack is a widget that allows you to overlay widgets on top of each other.

● Flutter Forms: Flutter forms are used to collect user input. They often involve using TextFields and validation.

● Flutter AlertDialog: AlertDialog is a popup dialog that displays important information or asks the user for input.

● Flutter Icons: Flutter comes with a set of customizable icons that can be used in your app.

● Flutter Images: Flutter supports the loading and display of images using the Image widget.

● Flutter Card: Card is a material design card. It's a container with rounded corners and elevation.

● Flutter Tabbar: TabBar is a widget that displays a horizontal row of tabs.

● Flutter Drawer: Drawer is a slide-in menu that is typically used for navigation.

● Flutter Lists: Lists are used to display a scrolling list of widgets. ● Flutter GridView: GridView is a scrollable grid of widgets.

● Flutter Toast: Toast is a temporary notification that appears at the bottom of the screen.

● Flutter Checkbox: Checkbox is a UI element that allows users to toggle between two states.
● Flutter Radio Button: Radio buttons allow users to select one option from a set.
● Flutter Progress Bar: Progress Bar indicates the progress of an ongoing task.
● Flutter Snackbar: Snackbar provides lightweight feedback about an operation.
● Flutter Tooltip: Tooltip provides additional information when the user hovers over a widget.
● Flutter Slider: Slider allows users to select a value from a range.
● Flutter Switch: Switch is a UI element for toggling between two states.
● Flutter Charts: Flutter supports various charting libraries for visualizing data.
● Bottom Navigation Bar: A bar at the bottom of the screen for navigation.
● Flutter Themes: Themes define the colors, fonts, and styles used in an app.
● Flutter Table: Table is used to create a two-dimensional array of widgets.
● Flutter Calendar: Calendar widgets help in displaying and selecting dates.
● Flutter Animation: Flutter supports various animation widgets and APIs for creating smooth animations.

Code:
Make a folder theme and add a file pallet.dart

```dart
import 'package:flutter/material.dart';

class Pallete {
  static const blackColor = Color.fromRGBO(1, 1, 1, 1);
  static const whiteColor = Colors.white;
  static const greyColor = Colors.black54;
  static const whiteColoor = Colors.white38;
  static const blueColor = Colors.blue;
  static const transparent = Colors.transparent;
  //Themes
  static var darkModeAppTheme = ThemeData.dark().copyWith(
    scaffoldBackgroundColor: blackColor,
  );
  static var lightModeAppTheme = ThemeData.light().copyWith(
    scaffoldBackgroundColor: whiteColor,
  );
}
```

In main.dart

```dart
import 'package:flutter/material.dart';
// import 'package:thread_clone_flutter/screens/home.dart';
```

```dart
import 'themes/pallet.dart';

void main() {
  runApp(const MyApp());
}
class MyApp extends StatelessWidget {
  const MyApp({super.key});
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Threads Clone',
      theme: Pallete.darkModeAppTheme,
       home: const SplashScreen()
      // home: const LoginScreen(),
    );
  }

}
```

Output: