Experiment 2

Aim: To design Flutter UI by including common widgets.

Theory:

Widgets: Each element on a screen of the Flutter app is a widget. The view of the screen completely depends upon the choice and sequence of the widgets used to build the apps. And the structure of the code of an apps is a tree of widgets.

Category of Widgets:

There are mainly 14 categories in which the flutter widgets are divided. They are mainly segregated on the basis of the functionality they provide in a flutter application.

1. Accessibility: These are the set of widgets that make a flutter app more easily accessible.
2. Animation and Motion: These widgets add animation to other widgets.
3. Assets, Images, and Icons: These widgets take charge of assets such as display images and show icons.
4. Async: These provide async functionality in the flutter application.
5. Basics: These are the bundle of widgets that are absolutely necessary for the development of any flutter application.
6. Cupertino: These are the iOS designed widgets.
7. Input: This set of widgets provides input functionality in a flutter application.
8. Interaction Models: These widgets are here to manage touch events and route users to different views in the application.
9. Layout: This bundle of widgets helps in placing the other widgets on the screen as needed.
10. Material Components: This is a set of widgets that mainly follow material design by Google.
11. Painting and effects: This is the set of widgets that apply visual changes to their child widgets without changing their layout or shape.
12. Scrolling: This provides scrollability of to a set of other widgets that are not scrollable by default.
13. Styling: This deals with the theme, responsiveness, and sizing of the app.
14. Text: This displays text.

Description of few of the widgets are as follows:

- Scaffold – Implements the basic material design visual layout structure.
- App-Bar – To create a bar at the top of the screen.
- Text - To write anything on the screen.
- Container – To contain any widget.
- Center – To provide center alignment to other widgets.

The code in main.dart:

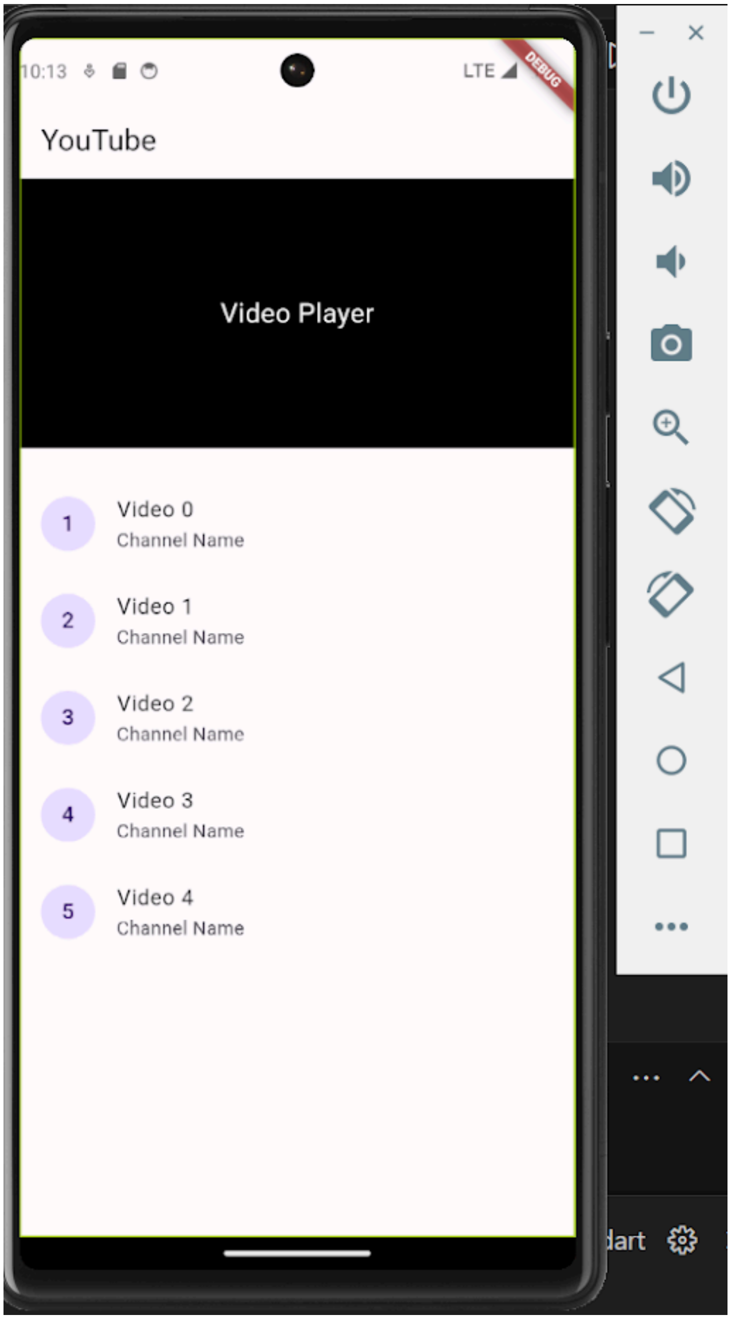
```
import 'package:flutter/material.dart';

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

class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
   return MaterialApp(
     title: 'YouTube Page',
     theme: ThemeData(
       primarySwatch: Colors.red,
       visualDensity: VisualDensity.adaptivePlatformDensity,
     ),
     home: YouTubePage(),
   );
 }
}

class YouTubePage extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
   return Scaffold(
     appBar: AppBar(
       title: Text('YouTube'),
     ),
     body: Column(
       children: [
         Container(
           height: 200,
           color: Colors.black,
           alignment: Alignment.center,
           child: Text(
             'Video Player',
             style: TextStyle(color: Colors.white, fontSize: 20),
           ),
         ),
         SizedBox(height: 20),
         Expanded(
           child: ListView.builder(
             itemCount: 5,
             itemBuilder: (context, index) {
               return ListTile(
                 leading: CircleAvatar(
                   child: Text((index + 1).toString()),
                 ),
                 title: Text('Video $index'),
                 subtitle: Text('Channel Name'),
                 onTap: () {
                   // Navigate to video details page
                 },
               );
             },
           ),
         ),
       ],
     ),
   );
 }
}
```

Output:



YouTube

Video Player

1 Video 0
Channel Name

2 Video 1
Channel Name

3 Video 2
Channel Name

4 Video 3
Channel Name

5 Video 4
Channel Name