PRN: 121A3014

Experiment No.: 2

Aim: To design Flutter UI by including common widgets. (Text ,Buttons, Container, Scaffold, AppBar,Row, Column)

Theory:

What is Flutter?

Flutter was introduced by Google as an open-source technology for coding and creating native apps for Android and iOS. Flutter is relatively new as it was officially presented in December 2018 as the first stable version 1.0 at the Flutter Live event.

Types of Widgets:

There are two types of Widgets:

A. **Visible widget (Output and Input) :**

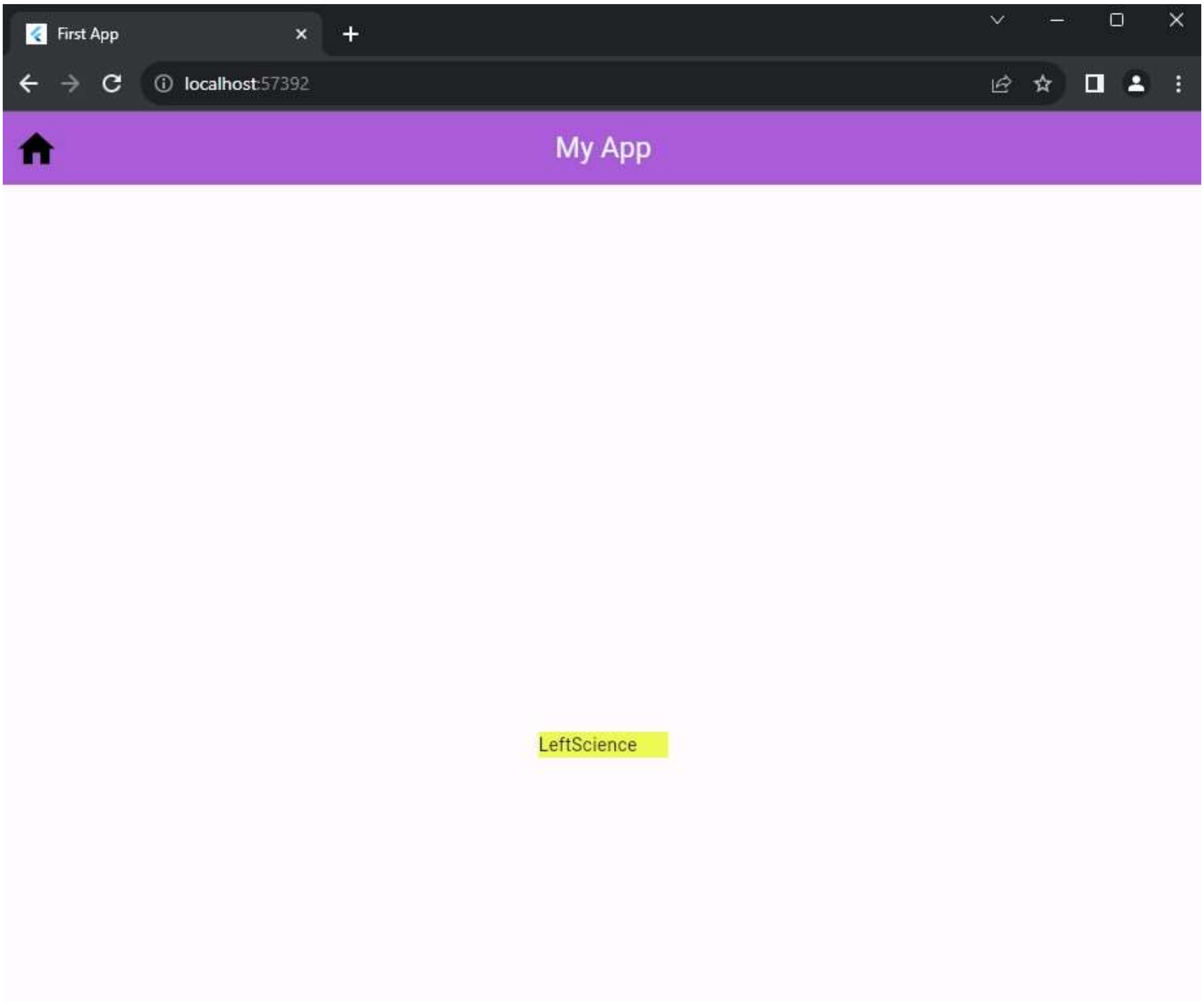
The visible widgets are related to the user input and output data. Some of theimportanttypes of this widget are:

1. **Text**: A Text widget holds some text to display on the screen. We can align the textwidget by using *textAlign* property, and style property allow the

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

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

class MyApp extends StatelessWidget {
  const MyApp({super.key});
```

```
 @override
 Widget build(BuildContext context) {
   return MaterialApp(
     title: 'First App',
     debugShowCheckedModeBanner: false,
     theme: ThemeData(primarySwatch: Colors.blue),
     home: Scaffold(
       appBar: AppBar(
         title: Text('My App'),
         backgroundColor: Color.fromARGB(255, 161, 101, 220),
         leading: Icon(Icons.home),
         centerTitle: true,
         foregroundColor: Color.fromARGB(255, 232, 246, 249),
         iconTheme: IconThemeData(color: Colors.black, size: 35.0),
       ),
       body: Center(
           child: Container(
         width: 100.0,
         height: 20.0,
         color: Color.fromARGB(255, 246, 246, 55),
         child: const RandomWord(),
         // child: Text('Hello World'),
       )),
     ),
   );
 }
}

class RandomWord extends StatefulWidget {
 const RandomWord({super.key});

 @override
 State<RandomWord> createState() => _RandomWordState();
}
```

```
class _RandomWordState extends State<RandomWord> {
  @override
  Widget build(BuildContext context) {
    final wordpair = WordPair.random();
    return Text(wordpair.asPascalCase);
  }
}
```



LeftScience

2. Stateful and Stateless:

Stateless Widget:

Stateless widgets are the widgets that don’t change i.e. they are immutable. Its appearance andproperties remain unchanged throughout the lifetime of the widget. In simple words, Stateless widgets cannot change their state during the runtime of the app, which means the widgets cannot be redrawn while the app is in action. *Examples:* Icon, IconButton, and Text are examples of stateless widgets.

Stateful Widget:

Stateful Widgets are the ones that change its properties during run-time. They are dynamic i.e.,they are mutable and can be drawn multiple times within its lifetime. It can change its appearance in response to events triggered by user interactions or when it receives data.

Examples : Checkbox, Radio Button, *Slider, InkWell*, Form, and *TextField* are examples ofStateful widgets.

To create a Stateful widget, we have to override the *createState*() method, which returns thestate of the widget.

Basic Widgets in Flutter:

AppBar: It is usually the topmost component of the app (or sometimes the bottom-most), it contains the toolbar and some other common action buttons.As all the components in a flutter application is a widget or a combination ofwidgets. So AppBar is also a built-in class or widget in flutter which gives the functionality of the AppBar out of the box. The AppBar widget is based on Material Design and much of the information is already provided by other classes like MediaQuery, Scaffold as to where the content of the AppBar should be placed.

```
appBar: new AppBar( title: new
Text('Appbar'),
         backgroundColor: Colors.green,
         elevation: 30.0,
         centerTitle: true,
         titleSpacing: 70.0,
         toolbarOpacity: 0.5,
       ),
```

Icons: Flutter provides an Icon Widget to create icons in our applications. We can create icons in Flutter, either using inbuilt icons or with the customicons. Flutter provides the list of all icons in the Icons class. The following code snippet demonstrates Flutter icons in the application.

```
appBar: new AppBar(
      title: new Text('Appbar'),
     leading: new Icon(Icons.menu),
     actions: <Widget>[
     new IconButton(icon: new Icon(Icons.arrow_forward),onPressed: (){print('you pressed
     Arrow Button');},),
     new IconButton(icon: new Icon(Icons.add),onPressed: (){print('you pressed Addition
     Button');}),
     new IconButton(icon: new Icon(Icons.close),onPressed: (){print('you pressed Close
     Button');}),
      ],
    ),
```

- **Row:** This widget arranges its children in a horizontal direction on the screen. In other words, it will expect child widgets in a horizontal array. If the child widgets need to fill the available horizontal space, we must wrap the children widgets in an Expanded widget. A row widget does not appear scrollable because it displays the widgets within the visible view. So, it is considered wrong if we have more children in a row which will not fit in theavailable space. If we want to make a scrollable list of row widgets, we need to use the ListView widget.

- **Column:** This widget arranges its children in a vertical direction on the screen. In other words, it will expect a vertical array of children widgets. If the child widgets need to fill the available vertical space, we must wrap the children widgets in an Expanded widget. A column widget does not appear scrollable because it displays the widgets within the visible view. So, it is considered wrong if we have more children in a column which will not fit inthe available space. If we want to make a scrollable list of column widgets, we need to use the ListView Widget.needs. Generally, it is similar to a box for storing contents. It allows many attributes to the user for decorating its child widgets, such as using margin,which separates the container with other contents.

- **Buttons:** They are the Flutter widgets, which are a part of the material design library. It allows a user to trigger an event; such as taking actions, making choices, searching things, and many more. They can be placed anywhere in our UI like dialogs, forms, cards, toolbars, etc. Flutter providesseveral types of buttons that have different shapes, styles, and features.

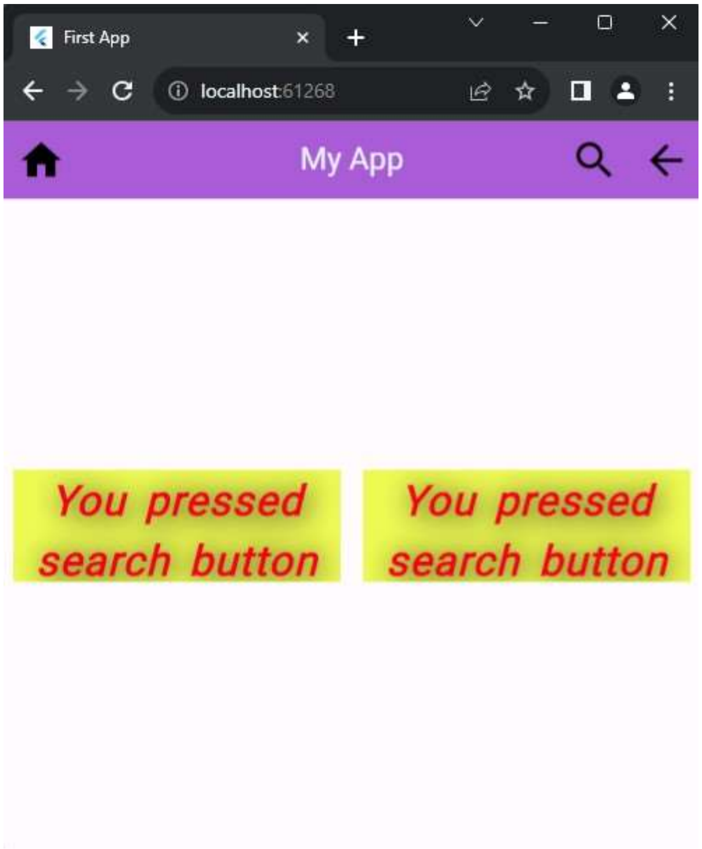
- **Text:** A Text is a widget in Flutter that allows us to display a string of text with a single line in our application. Depending on the layout constraints, we can break the string across multiple lines or might all be displayed on the same line. If we do not specify any styling to the text widget, it will use the closest DefaultTextStyle class style. This class does not have any explicit style.The following are the essential properties of the Text widget: TextAlign, TextDirection, Overflow, TextScaleFactor, SoftWrap, MaxLines, TextWidthBasis, TextHeightBehavior, Style.

- **Center:** This widget comes built-in with flutter; it aligns its child widget to the center of the available space on the screen. The size of this widget will beas big as possible if the widthFactor and heightFactor properties are set to null and the dimensions are constrained. And in case the dimensions are not constrained and the widthFactor and HeightFactor are set to null then the Center widget takes the size of its child widget.

Program:

Row,column and container :

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

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

class MyApp extends StatefulWidget {
  const MyApp({super.key});

  @override
  State<MyApp> createState() => _MyAppState();
}

class _MyAppState extends State<MyApp> {
  String msg = '';
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'First App',
      debugShowCheckedModeBanner: false,
      theme: ThemeData(primarySwatch: Colors.blue),
      home: Scaffold(
        appBar: AppBar(
          actions: <Widget>[
            IconButton(
                onPressed: () {
                  setState(() {
                    msg = 'You pressed search button';
                  });
                },
                icon: Icon(Icons.search)),
            IconButton(
                onPressed: () {
                  setState(() {
                    msg = 'You pressed back button';
                  });
                },
                icon: Icon(Icons.arrow_back))
          ],
          title: Text('My App'),
          backgroundColor: Color.fromARGB(255, 161, 101, 220),
          leading: Icon(Icons.home),
          centerTitle: true,
          foregroundColor: Color.fromARGB(255, 232, 246, 249),
          iconTheme: IconThemeData(color: Colors.black, size: 35.0),
        ),
        body: Row(
          children: [
            Expanded(
              child: Center(
                  child: Padding(
                padding: const EdgeInsets.all(8.0),
                child: Container(
                  width: 800.0,
                  height: 80.0,
                  color: Color.fromARGB(255, 246, 246, 55),
                  // child: const RandomWord(),
                  child: Text(
                    msg,
                    textAlign: TextAlign.center,
                    style: const TextStyle(
                      color: Color.fromARGB(255, 242, 19, 19),
                      fontSize: 30,
                      fontStyle: FontStyle.italic,
                      fontWeight: FontWeight.bold,
                      letterSpacing: 1,
                      wordSpacing: 5,
                      shadows: [
                        Shadow(
                            color: Color.fromARGB(255, 13, 13, 14),
                            offset: Offset(2, 2),
                            blurRadius: 10)
                      ],
                    ),
                  ),
                ),
              )),
            ),
            Expanded(
              child: Center(
                  child: Padding(
                padding: const EdgeInsets.all(8.0),
                child: Container(
                  width: 800.0,
                  height: 80.0,
                  color: Color.fromARGB(255, 246, 246, 55),
                  // child: const RandomWord(),
                  child: Text(
                    msg,
                    textAlign: TextAlign.center,
                    style: const TextStyle(
                      color: Color.fromARGB(255, 242, 19, 19),
                      fontSize: 30,
                      fontStyle: FontStyle.italic,
                      fontWeight: FontWeight.bold,
                      letterSpacing: 1,
                      wordSpacing: 5,
                      shadows: [
                        Shadow(
                            color: Color.fromARGB(255, 13, 13, 14),
                            offset: Offset(2, 2),
                            blurRadius: 10)
                      ],
                    ),
                  ),
                ),
              )),
            ),
          ],
        ),
      ),
    );
  }
}

class RandomWord extends StatefulWidget {
  const RandomWord({super.key});

  @override
  State<RandomWord> createState() => _RandomWordState();
}

class _RandomWordState extends State<RandomWord> {
  @override
  Widget build(BuildContext context) {
    final wordpair = WordPair.random();
    return Text(wordpair.asPascalCase);
  }
}
```



You pressed search button

You pressed search button

Conclusion:

Hence, we have successfully designed a Flutter UI by including common widgets. And we have learned that Flutter is one of the best solutions to develop apps for Android and iOS, without having to write in a different codebase for each platform. The smartphone versions of these apps function as true, native apps on Apple and Android devices and are compiled for the respective platform before publication. They do not need a runtime module or a browser. Using the same codebase, it is also possible to create web apps for browsers as well as native programs for Windows, Linux and macOS. The popularity of Flutter app development is constantly growing.