Anket Kadam MAD PWA Lab
D15B/26

Experiment 2

AIM: To design Flutter UI by including common widgets.

THEORY:

Introduction to Flutter Widgets:

- Define what Flutter widgets are and their significance in building user interfaces.
- Mention how Flutter widgets are the basic building blocks of Flutter applications and how they contribute to creating interactive UI components.

Overview of Common Widgets:

- Provide an overview of the most commonly used widgets in Flutter.
- Categorize them based on their functionality such as layout, text, input, interaction, etc.

Layout Widgets:

- Explain layout widgets like Container, Row, Column, Stack, and ListView.
- Discuss how each layout widget works and its typical use cases.
- Demonstrate how to use them to create different UI layouts.

Text Widgets:

- Introduce text widgets like Text, RichText, and TextField.
- Explain how to style text, handle text overflow, and implement text editing functionalities.

Input Widgets:

- Discuss input widgets such as TextField, Checkbox, Radio, and Switch.
- Describe how to handle user input events and validate input data.

Interaction Widgets:

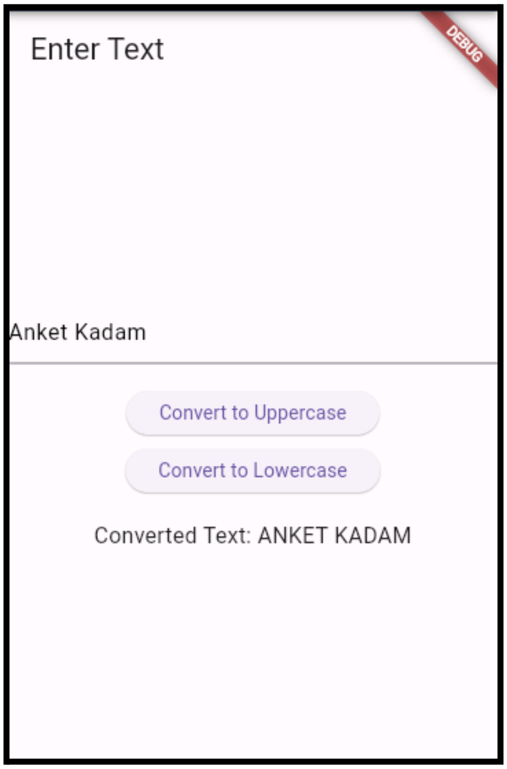
- Explore interaction widgets like GestureDetector, InkWell, Draggable, and Slider.
- Illustrate how to handle gestures and implement custom touch interactions.

Material Design Widgets:

- Introduce material design widgets like AppBar, BottomNavigationBar, FloatingActionButton, and SnackBar.
- Explain how to implement common material design components in Flutter apps.

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Text Display App',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: TextDisplayScreen(),
    );
  }
}

class TextDisplayScreen extends StatefulWidget {
  @override
   _TextDisplayScreenState createState() => _TextDisplayScreenState();
}

class _TextDisplayScreenState extends State<TextDisplayScreen> {
  String _displayText = '';

  void _updateDisplayText(String newText) {
    setState(() {
      _displayText = newText;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text('Text Display App'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            TextField(
              decoration: const InputDecoration(
                hintText: 'Enter your text here',
              ),
              onChanged: (text) {
                _updateDisplayText(text);
              },
            ),
            const SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                // Action to be performed when button is pressed
                // For now, we'll just print the entered text
                print(_displayText);
              },
              child: const Text('Display Text'),
            ),
            const SizedBox(height: 20),
            Text(
              'Displayed Text: $_displayText',
              style: const TextStyle(fontSize: 20),
            ),
          ],
        ),
      ),
    );
  }
}
```

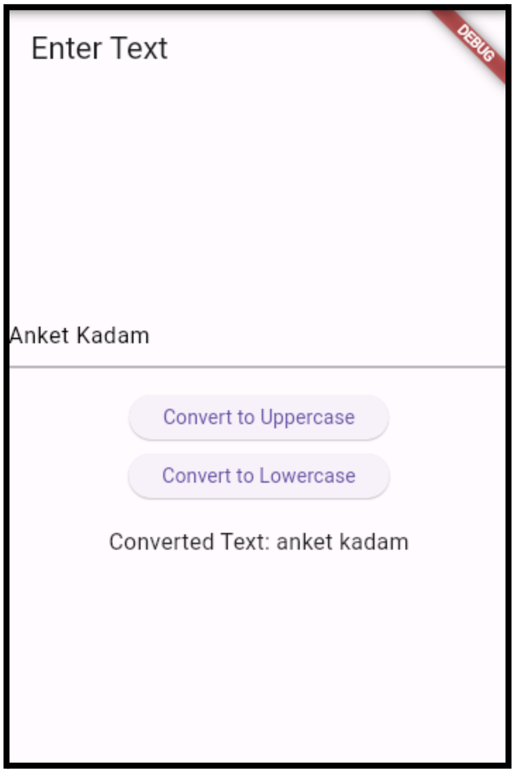
OUTPUT:



Enter Text

Anket Kadam

Converted Text: ANKET KADAM



Enter Text

Anket Kadam

Converted Text: anket kadam

Conclusion: we understood and studied common and essential flutter widgets.