Name: Atif Ansari Roll no: 04 Class: D15B

MPL Experiment - 2

* **Aim:** To Design Flutter UI by including common widgets.

* **Theory:** Flutter provides a variety of pre-build UI widgets that help in creating rich and interactive user interfaces for mobile applications. These widgets are categorized into Structural Widgets, Interactive Widgets, Layout Widgets, and Styling Widgets.

Types of Widgets used in Flutter:

1. **Structural Widgets:**
   - **Scaffold:** Provides a basic layout structure with an app bar, body, and floating buttons.
   - **AppBar:** Displays a toolbar with a title and actions.
   - **SafeArea:** Ensures content does not overlap with system UI elements.
   - **SingleChildScrollView:** Allows scrolling when content exceeds screen size.

2. **Interactive Widgets:**
   - **ElevatedButton:** A material-styled button that elevates when pressed.
   - **IconButton:** A clickable icon.

3. **Layout Widgets:**
   - **Column:** Arranges child widgets in a vertical direction.

- Padding: Adds spacing around widgets.
- Align: Positions a widget within its parent.

4. Styling Widgets:
- Text: Displays styled text.
- Icon: Displays material icons.
- Container: Used for decoration and styling.

* Conclusion: This experiment focused on building a Flutter UI using common widgets. By using Scaffold, AppBar, Text, Column, ElevatedButton, Icon, and Padding, we successfully created a structured Settings UI with different sections. The Scaffold provided the screen layout, AppBar created a navigation bar, and Column was used to align elements vertically. Each section header was styled using Text, and interactive buttons allowed user actions.

We also made use of SafeArea and ~~Single~~ SingleChildScrollView to ensure the UI was responsive and ~~adaptable~~ adaptable across devices. The use of ElevatedButton.icon() showcased how we could combine icons with text in buttons to improve user experience.

Overall, this experiment demonstrated the power of flutter's widget-based approach to UI development, allowing for flexibility, reusability, and responsive designs.

Code:

```
import 'package:flutter/material.dart';
void main() {
 runApp(const AccountApp());
}
class AccountApp extends StatelessWidget {
 const AccountApp({Key? key}) : super(key: key);
 @override
 Widget build(BuildContext context) {
  return MaterialApp(
   home: const Account(),
  );
 }
}
class Account extends StatelessWidget {
 const Account({Key? key}) : super(key: key);
 @override
 Widget build(BuildContext context) {
  return Scaffold(
   backgroundColor: Colors.white24,
   appBar: AppBar(
    centerTitle: true,
    backgroundColor: Colors.white24,
    leading: IconButton(
     onPressed: () {
      Navigator.pop(context);
     },
     icon: const Icon(Icons.arrow_back),
    ),
    title: const Text(
     "Settings",
     style: TextStyle(color: Colors.white),
    ),
   ),
   body: SafeArea(
    child: Padding(
     padding: const EdgeInsets.fromLTRB(10, 20, 10, 0),
     child: SingleChildScrollView(
      child: Column(
       crossAxisAlignment: CrossAxisAlignment.start,
       children: [
        _buildSectionHeader("PROFILE"),
        const SizedBox(height: 2),
        _buildProfileButton(
         label: "VESIT\nVESIT",
         onPressed: () {
          print("You pressed Profile Button");
         },
```

```
),
const SizedBox(height: 10),
_buildSectionHeader("FEATURES"),
const SizedBox(height: 10),
_buildFeatureButton(
  label: "Memories",
  icon: Icons.calendar_today,
  onPressed: () {
    print("You pressed Memories Button");
  },
),
_buildFeatureButton(
  label: "Blocked Profile",
  icon: Icons.block,
  onPressed: () {
    print("You pressed Blocked Profile Button");
  },
),
const SizedBox(height: 10),
_buildSectionHeader("SETTINGS"),
_buildFeatureButton(
  label: "Notifications",
  icon: Icons.notifications,
  onPressed: () {
    print("You pressed Notifications Button");
  },
),
_buildFeatureButton(
  label: "Time Zone",
  icon: Icons.access_time,
  onPressed: () {
    print("You pressed Time Zone Button");
  },
),
_buildFeatureButton(
  label: "Others",
  icon: Icons.settings_suggest,
  onPressed: () {
    print("You pressed Others Button");
  },
),
const SizedBox(height: 10),
_buildSectionHeader("ABOUT"),
_buildFeatureButton(
  label: "Share BeReal",
  icon: Icons.share,
  onPressed: () {
    print("You pressed Share BeReal Button");
  },
```

```
            ),
            _buildFeatureButton(
              label: "Rate",
              icon: Icons.star_outline,
              onPressed: () {
                print("You pressed Rate Button");
              },
            ),
            _buildFeatureButton(
              label: "Help",
              icon: Icons.help_outline,
              onPressed: () {
                print("You pressed Help Button");
              },
            ),
            _buildFeatureButton(
              label: "About",
              icon: Icons.info,
              onPressed: () {
                print("You pressed About Button");
              },
            ),
          ],
        ),
      ),
    ),
  );
}
Widget _buildSectionHeader(String title) {
  return Row(
    children: [
      Text(
        title,
        style: const TextStyle(fontSize: 22, fontWeight: FontWeight.bold),
      ),
      const VerticalDivider(
        color: Colors.transparent,
        thickness: 1,
        width: 10,
      ),
    ],
  );
}
Widget _buildProfileButton({required String label, required VoidCallback onPressed}) {
  return ElevatedButton.icon(
    onPressed: onPressed,
    icon: const Icon(Icons.account_circle, color: Colors.white),
    label: Text(
```

```
      label,
      style: const TextStyle(color: Colors.white, fontSize: 16),
    ),
    style: ElevatedButton.styleFrom(backgroundColor: Colors.deepOrangeAccent),
  );
 }
 Widget _buildFeatureButton({required String label, required IconData icon, required VoidCallback onPressed})
{
  return ElevatedButton.icon(
    onPressed: onPressed,
    icon: Icon(icon, color: Colors.white),
    label: Text(label, style: const TextStyle(color: Colors.white, fontSize: 18)),
    style: ElevatedButton.styleFrom(backgroundColor: Colors.deepOrangeAccent),
  );
 }
}
```

Output: