

Aim: To design flutter UI using commonly used widgets

Theory:

### 1. Container:

- Purpose: The Container widget serves as a flexible box model, allowing the creation of customizable boxes within the UI.
- Usage: It is extensively employed for layout and styling, offering options to set background color, padding, margin, alignment, and more.
- Example: Utilize a Container to define a specific section of the UI, providing it with a distinctive background color and padding for visual separation.

### 2. Text:

- Purpose: The Text widget is a fundamental element for displaying text with various styling options within the app.
- Usage: Customization options include font size, weight, color, and other text-related properties. It's essential for presenting textual information in a visually appealing manner.
- Example: Employ a Text widget to exhibit dynamic textual content or display a welcoming message with customizable styling.

### 3. Card:

- Purpose: Card is a material design widget used to visually group related information, enhancing the aesthetic appeal of the UI.
- Usage: Frequently encapsulating other widgets, it provides a consistent look for grouped elements, commonly used in lists or as standalone UI elements.
- Example: Implement a Card to present detailed information about specific items in the app, offering a visually cohesive and structured display.

### 4. ListTile:

- Purpose: ListTile is a convenient widget for displaying a single fixed-height row in a list, often containing text and an optional icon.
- Usage: Ideal for representing information in a list format with support for tap interactions. Useful for settings lists or displaying categorized data.

- Example: Utilize a ListTile to represent items in a settings list, incorporating icons to convey the type of setting alongside descriptive text.

## 5. Buttons:

- Purpose: Flutter provides various button widgets such as ElevatedButton, TextButton, and OutlinedButton, each catering to different button styles.
- Usage: Essential for user interactions, buttons trigger actions, navigate between screens, or submit forms. They are highly customizable in appearance and behavior.
- Example: Apply an ElevatedButton for form submission, a TextButton for a simple action, and an OutlinedButton for a less prominent action within the UI.

## 6. Row & Column:

- Purpose: Row and Column are layout widgets facilitating the arrangement of child widgets horizontally (Row) or vertically (Column).
- Usage: Essential for organizing UI elements side by side or vertically stacked. Allows for flexibility in sizing and proportions among child widgets.
- Example: Utilize a Row to align an icon and text horizontally or a Column to stack multiple widgets vertically, creating structured layouts within the app.

## 7. Scaffold:

- Purpose: The Scaffold widget forms the basic structural element in a Flutter app, providing a visual canvas for the entire screen.
- Usage: Typically houses major visual components like AppBar, Body, Drawer, BottomNavigationBar, defining the app's overall layout structure.
- Example: Employ a Scaffold to structure the main screen of the app, incorporating an AppBar for navigation, a central Body for content, and potentially a BottomNavigationBar for intuitive navigation.

## 8. AppBar:

- Purpose: The AppBar is a specialized widget designed for displaying a material design app bar at the top of the screen.
- Usage: Contains elements like title, leading/trailing widgets, and actions, offering a consistent navigation and branding area for the app.

- Example: Utilize an AppBar to showcase the title of the current screen along with navigation buttons, actions, or settings, contributing to the overall navigation and user experience.

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

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

```
class MyApp extends StatelessWidget {  
  const MyApp({Key? key}) : super(key: key);
```

```
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
      title: 'Flutter Example',  
      theme: ThemeData(  
        primarySwatch: Colors.blue,  
      ),  
      home: const MyHomePage(),  
    );  
  }  
}
```

```
class MyHomePage extends StatelessWidget {  
  const MyHomePage({Key? key}) : super(key: key);
```

```
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: const Text(  
          'Flutter Example App',
```

```
        style: TextStyle(
          color: Colors.white,
        ),
      ),
      backgroundColor: Colors.blue,
      centerTitle: true,
    ),
    body: Padding(
      padding: const EdgeInsets.all(16.0),
      child: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            const Text(
              'Welcome to Flutter!',
              style: TextStyle(
                fontSize: 24,
                fontWeight: FontWeight.bold,
              ),
            ),
            const SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                // Add your logic here
              },
              style: ElevatedButton.styleFrom(
                primary: Colors.blue,
                onPrimary: Colors.white,
              ),
              child: const Text('Press Me'),
            ),
            const SizedBox(height: 20),
            Container(
              decoration: BoxDecoration(
                color: Colors.greenAccent,
```

```
        borderRadius: BorderRadius.circular(10),
      ),
      padding: const EdgeInsets.all(16),
      child: Column(
        crossAxisAlignment: CrossAxisAlignment.start,
        children: [
          const Text(
            'Details',
            style: TextStyle(
              fontSize: 20,
              fontWeight: FontWeight.bold,
            ),
          ),
          const SizedBox(height: 10),
          Text(
            'Developer: John Doe',
            style: TextStyle(
              fontSize: 16,
              fontWeight: FontWeight.bold,
            ),
          ),
          Text(
            'Email: john.doe@example.com',
            style: TextStyle(
              fontSize: 16,
              fontWeight: FontWeight.bold,
            ),
          ),
        ],
      ),
    ],
  ),
),
),
```

);  
}  
}

