Experiment No.2

- **Aim:** To design Flutter UI by including common widgets.

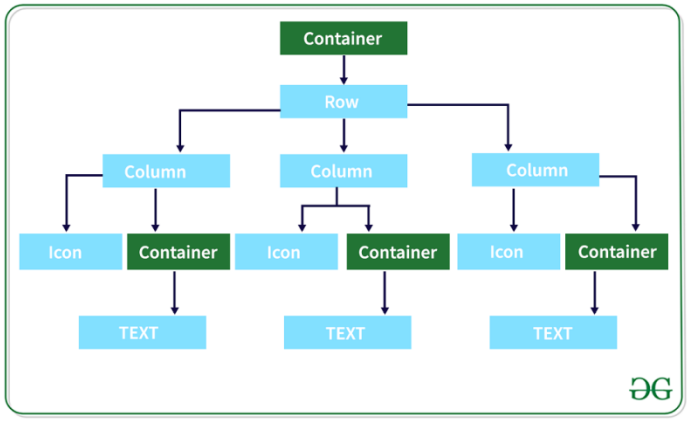
- **Theory:**

Widgets:

1. Widgets are the building blocks of Flutter UI.
2. Everything in Flutter is a widget, from structural elements like Container and Row to input elements like TextField and Button.
3. Widgets can be categorized as either StatelessWidget or StatefulWidget based on whether they can change over time.
4. StatelessWidget represents widgets that do not require mutable state, while StatefulWidget represents widgets that require mutable state.

Layouts and Structure:

1. Flutter provides a wide range of layout widgets to arrange UI elements in various configurations.
2. Common layout widgets include Column, Row, Stack, GridView, ListView, and Container.
3. Each layout widget has its own properties and behaviors for organizing child widgets.
4. Using Material Design Widgets:
5. Material Design is a design language developed by Google, providing guidelines for visual, motion, and interaction design.
6. Flutter includes a set of Material Design widgets that adhere to these guidelines, making it easy to create apps with a consistent look and feel.
7. Common Material Design widgets include AppBar, BottomNavigationBar, FloatingActionButton, Card, SnackBar, etc.



Customizing Widgets and Themes:

1. Flutter allows for extensive customization of widgets to match the design requirements of your app.
2. You can customize the appearance of widgets using properties like color, padding, margin, border, etc.
3. Themes in Flutter allow you to define a consistent set of visual properties (such as colors, typography, and shapes) that can be applied throughout your app.
4. The Theme widget and ThemeData class enable you to define and apply themes easily.

Handling User Input and Navigation:

1. Flutter provides various widgets for handling user input, such as TextField, Checkbox, RadioButton, Slider, GestureDetector, etc.
2. Navigation in Flutter is typically managed using a Navigator widget, which maintains a stack of routes/pages.
3. Common navigation widgets include MaterialPageRoute, CupertinoPageRoute, Navigator, BottomNavigationBar, etc.

- **Code:-**

1. Padding Widget :

```
Padding(
  padding: const EdgeInsets.symmetric(horizontal: 25.0),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.end,
    children: [
      Text(
        'Forgot Password?',
        style: TextStyle(color: Colors.grey[600]),
      ), // Text
    ],
  ), // Row
), // Padding
```

This widget adds padding around the child widget (`Row`) to create spacing. To add visual separation and alignment for the "Forgot Password?" text.

2. Divider Widget :

```
Padding(
  padding: const EdgeInsets.symmetric(horizontal: 25.0),
  child: Row(
    children: [
      Expanded(
        child: Divider(
          thickness: 0.5,
          color: Colors.grey[400],
        ), // Divider
      ), // Expanded
      Padding(
        padding: const EdgeInsets.symmetric(horizontal: 10.0),
        child: Text(
          'Or continue with',
          style: TextStyle(color: Colors.grey[700]),
        ), // Text
      ), // Padding
      Expanded(
        child: Divider(
          thickness: 0.5,
          color: Colors.grey[400],
        ), // Divider
      ), // Expanded
    ],
  ), // Row
), // Padding
```

This widget displays a horizontal line (`Divider`) with text in between. To visually separate sections of the UI and provide textual context.

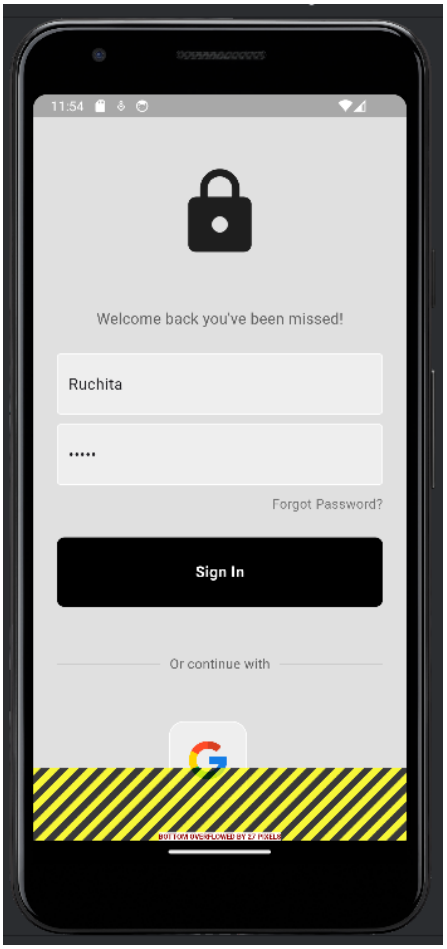
3. Row Widget :

```
Row(
  mainAxisAlignment: MainAxisAlignment.center,
  children: const [
    // google button
    SquareTile(imagePath: 'lib/images/google.png'),

    SizedBox(width: 25),
  ],
), // Row
```

This widget displays a row containing sign-in buttons. To offer options for users to sign in using different methods, such as Google or Apple sign-in.

- **Output:-**



- **Conclusion:**

Through designing Flutter UIs by incorporating common widgets, I've successfully gained hands-on experience in creating intuitive and visually appealing user interfaces. This process has not only enabled me to master the utilization of fundamental widgets but has also enriched my understanding of effective UI design principles and practices.