Experiment 03

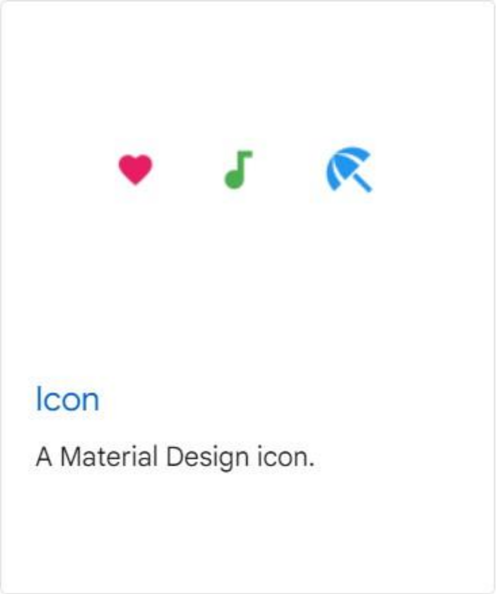
Aim: To include icons, images, fonts in Flutter app.

Theory:

Icons:

Flutter provides support for Material icons and Cupertino icons out of the box. To use these icons, you can use the Icon widget and specify the icon you want to use by passing the appropriate icon data.

```
Icon(Icons.favorite); // Material icon
```

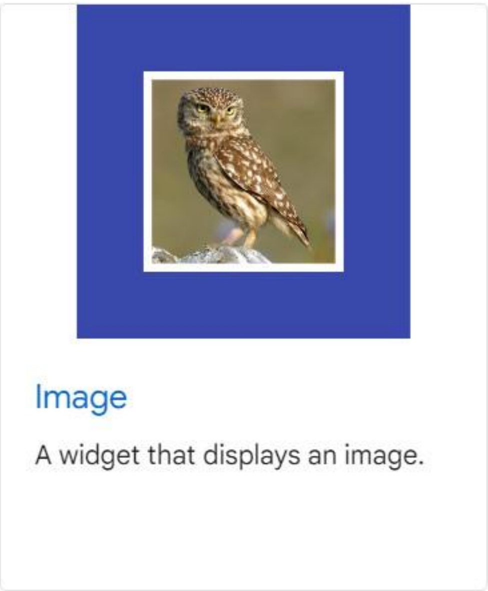


A Material Design icon.

Images:

You can include images in your Flutter app by placing them in the assets folder in your project directory. Then, you need to declare these assets in the pubspec.yaml file.

```
flutter:

  assets:

    - assets/my_image.png
```



Image

A widget that displays an image.

Fonts:

To include custom fonts in your Flutter app, you first need to add the font files (e.g., .ttf, .otf) to your project directory. Then, declare the font files in the pubspec.yaml file.

```
        flutter:
        fonts:
    - family: MyCustomFont          fonts:
    - asset: assets/fonts/my_custom_font.ttf
```

After declaring the fonts, you can use them in your Flutter app by specifying the font family in text widgets.

```
Text( 'Hello World', style:

 TextStyle( fontFamily:

 'MyCustomFont',

 ),

);
```

Code:

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

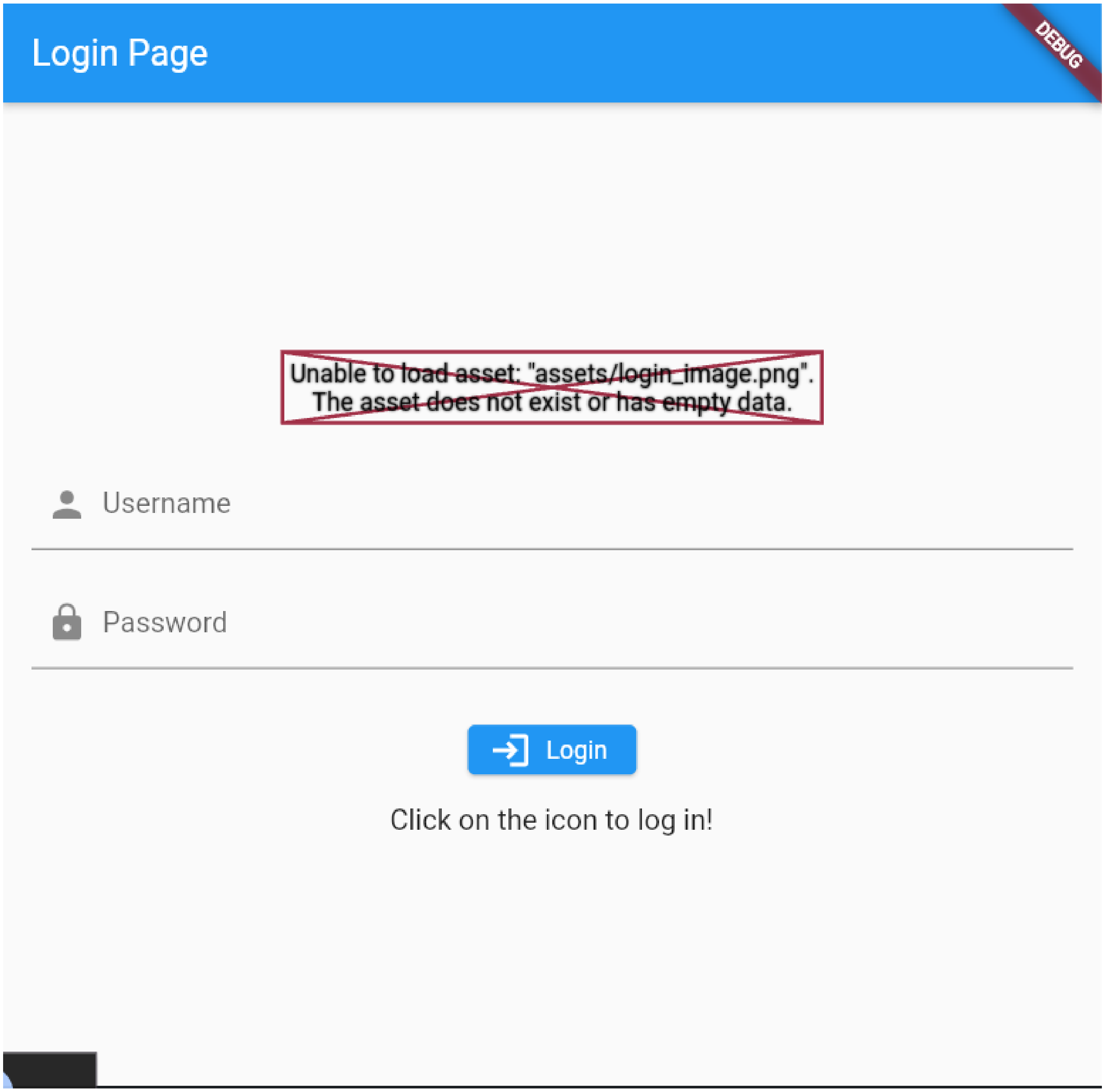
```
void main() {
  runApp(MyApp());
}

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

class LoginScreen extends StatelessWidget {
  final TextEditingController usernameController =
TextEditingController();
  final TextEditingController passwordController =
TextEditingController();

  void _handleLogin() {
    // Add authentication logic here
    String username = usernameController.text;
    String password = passwordController.text;

    // Example: Check if username and password are not empty
    if (username.isNotEmpty && password.isNotEmpty) {
      // Successful login - you can navigate to the next screen
      print('Login successful');
    } else {
      // Failed login - show an error message
      print('Login failed. Please enter both username and password.');
    }
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Login Page'),
      ),
      body: Padding(
        padding: EdgeInsets.all(16.0),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
            // Image
            Image.asset(
              'assets/login_image.png', // Replace with your image asset
              width: 150.0,
              height: 150.0,
            ),
            SizedBox(height: 20.0),
            // Username TextField with icon
            TextField(
              controller: usernameController,
              decoration: InputDecoration(
                labelText: 'Username',
                prefixIcon: Icon(Icons.person),
              ),
            ),
            SizedBox(height: 16.0),
            // Password TextField with icon
            TextField(
              controller: passwordController,
              obscureText: true,
              decoration: InputDecoration(
                labelText: 'Password',
                prefixIcon: Icon(Icons.lock),
              ),
            ),
            SizedBox(height: 32.0),
            // Login Button with icon
            ElevatedButton.icon(
              onPressed: _handleLogin,
              icon: Icon(Icons.login),
              label: Text('Login'),
            ),
            SizedBox(height: 16.0),
            // Custom Font Text
            Text(
              'Click on the icon to log in!',
              style: TextStyle(
                fontFamily: 'CustomFont',
                fontSize: 16.0,
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

Pubsec.yaml:

```
flutter:
  uses-material-design: true
  fonts:
    - family: CustomFont
      fonts:
        - asset: fonts/CustomFont.ttf  # Adjust the path based on your
project structure
  assets:
    - assets/login_image.png
```

Output:

Conclusion:



Unable to load asset: "assets/login_image.png".
The asset does not exist or has empty data.

Click on the icon to log in!

I learnt to implement custom font, icon and image in Flutter application.