

Name: Anshi Tiwari

Class: D15C

Roll No: 56

Experiment 03 : To include icons, images, fonts in Flutter app

Theory:

In Flutter, icons, images, and custom fonts enhance the visual appeal and user experience of an app:

- **Icons:** Flutter provides built-in icons through the Icons class in the Material package. You can also use custom icons by adding icon packs like FontAwesome or by including custom icon files.
- **Images:** Images can be included using the Image widget, which supports loading from:
- **Network:** Using Image.network() for online images.
- **Assets:** Using Image.asset() for images stored locally in the app's assets folder. These need to be declared in the pubspec.yaml file.
- **Fonts:** Custom fonts can be added by: Placing font files in the app's assets folder, Declaring them in the pubspec.yaml file, Applying them using the TextStyle class in widgets like Text.

Code:

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

```
class MyApp extends StatelessWidget {  
  const MyApp({super.key});
```

```
  @override
```

```
  Widget build(BuildContext context) {  
    return MaterialApp(  
      debugShowCheckedModeBanner: false,  
      title: 'Music Recommendation App',  
      theme: ThemeData.light().copyWith(  
        scaffoldBackgroundColor: Colors.white,  
        appBarTheme: const AppBarTheme(  
          backgroundColor: Colors.white,  
          foregroundColor: Colors.black,
```

```

    ),
    ),
    home: const LoginPage(),
  );
}
}

```

```

class LoginPage extends StatefulWidget {
  const LoginPage({super.key});

  @override
  _LoginPageState createState() => _LoginPageState();
}

```

```

class _LoginPageState extends State<LoginPage> {
  final TextEditingController _emailController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();
  bool _isPasswordVisible = false; // To manage password visibility

```

```

  void _login() {
    String email = _emailController.text;
    String password = _passwordController.text;

    if (email == "test@gmail.com" && password == "password123") {
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(
          builder: (context) => const MusicRecommendationPage()),
      );
    } else {
      ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(content: Text("Invalid email or password")),
      );
    }
  }
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(title: const Text('Login')),

```

```

body: Padding(
  padding: const EdgeInsets.all(16.0),
  child: Column(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      Image.network(
        'https://cdn-icons-png.flaticon.com/512/2907/2907253.png',
        height: 100,
        errorBuilder: (context, error, stackTrace) {
          return const Icon(Icons.error, size: 100, color: Colors.red);
        },
      ),
      const SizedBox(height: 20),
      TextField(
        controller: _emailController,
        decoration: const InputDecoration(
          labelText: "Email",
          enabledBorder: OutlineInputBorder(
            borderSide: BorderSide(color: Colors.black54),
          ),
          focusedBorder: OutlineInputBorder(
            borderSide: BorderSide(color: Colors.black),
          ),
        ),
      ),
      const SizedBox(height: 10),
      TextField(
        controller: _passwordController,
        obscureText: !_isPasswordVisible,
        decoration: InputDecoration(
          labelText: "Password",
          enabledBorder: const OutlineInputBorder(
            borderSide: BorderSide(color: Colors.black54),
          ),
          focusedBorder: const OutlineInputBorder(
            borderSide: BorderSide(color: Colors.black),
          ),
        ),
        suffixIcon: IconButton(
          icon: Icon(
            _isPasswordVisible

```

```

        ? Icons.visibility
        : Icons.visibility_off,
        color: Colors.black54,
      ),
      onPressed: () {
        setState(() {
          _isPasswordVisible = !_isPasswordVisible;
        });
      },
    ),
  ),
  const SizedBox(height: 20),
  ElevatedButton(
    onPressed: _login,
    style: ElevatedButton.styleFrom(
      backgroundColor: const Color.fromARGB(255, 177, 209, 236),
    ),
    child: const Text("Login"),
  ),
  const SizedBox(height: 20),
  ElevatedButton(
    onPressed: _login,
    style: ElevatedButton.styleFrom(
      backgroundColor: const Color.fromARGB(255, 177, 209, 236),
    ),
    child: const Text("New User? Signup"),
  ),
],
),
),
);
}
}

```

Explanation:

This Flutter code creates a Music Recommendation App with a simple login screen. Here's a brief breakdown:

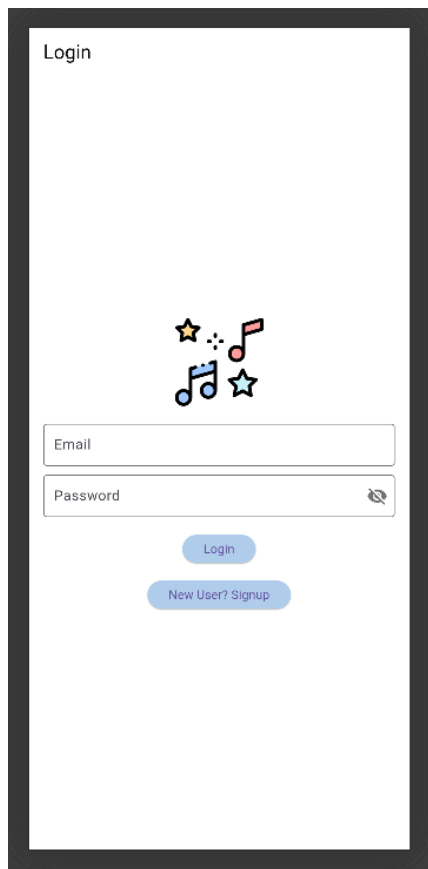
UI Design:

Displays:

- An image at the top.
- Text fields for email and password.
- A toggle to show or hide the password.
- A login button.
- A signup button.

This app uses Flutter's widgets and navigation to create a clean and functional login screen.

Output:



Conclusion:

This Flutter code builds a simple login screen for a music recommendation app. It verifies user credentials and navigates to the next page if they are correct. The design is clean, with input fields and password visibility toggle.