

MPL EXP-4

Ansh Sarfare

D15A - 49

Aim: To create an interactive form using form widget

Code:

```
import 'package:flutter/material.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:get/get.dart';
import 'signup.dart';
import 'forgot.dart';

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

  @override
  State<Login> createState() => _LoginState();
}

class _LoginState extends State<Login> {
  TextEditingController email = TextEditingController();
  TextEditingController password = TextEditingController();

  signIn() async {
    if (email.text.isEmpty || password.text.isEmpty) {
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          content: Text("Please fill in all fields"),
          backgroundColor: Colors.redAccent,
        ),
      );
      return;
    }

    try {
      await FirebaseAuth.instance.signInWithEmailAndPassword(
        email: email.text,
        password: password.text,
      );
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          content: Text("Login Successful!"),
          backgroundColor: Colors.green,
        ),
      );
    } catch (e) {
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          content: Text("Login Failed: $e"),
          backgroundColor: Colors.red,
        ),
      );
    }
  }
}
```

```

);
Get.offAllNamed('/home'); // Navigate to home page
} catch (e) {
  ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(
      content: Text("Error: ${e.toString()}"),
      backgroundColor: Colors.red,
    ),
  );
}
}

```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    backgroundColor: Colors.grey[200],
    body: Center(
      child: Padding(
        padding: const EdgeInsets.all(20.0),
        child: Card(
          elevation: 5,
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(20),
          ),
          child: Padding(
            padding: const EdgeInsets.all(25.0),
            child: Column(
              mainAxisAlignment: MainAxisAlignment.min,
              crossAxisAlignment: CrossAxisAlignment.center,
              children: [
                Text(
                  "Welcome Back!",
                  style: TextStyle(
                    fontSize: 24,
                    fontWeight: FontWeight.bold,
                    color: Colors.blueAccent,
                  ),
                ),
                SizedBox(height: 10),
                Text(
                  "Sign in to continue",
                  style: TextStyle(fontSize: 16, color: Colors.grey[700]),
                ),
                SizedBox(height: 20),

```

```

TextField(
  controller: email,
  decoration: InputDecoration(
    labelText: "Email",
    prefixIcon: Icon(Icons.email, color: Colors.blueAccent),
    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(10),
    ),
  ),
  keyboardType: TextInputType.emailAddress,
),
 SizedBox(height: 15),
TextField(
  controller: password,
  decoration: InputDecoration(
    labelText: "Password",
    prefixIcon: Icon(Icons.lock, color: Colors.blueAccent),
    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(10),
    ),
  ),
  obscureText: true,
),
 SizedBox(height: 20),
ElevatedButton(
  onPressed: signIn,
  style: ElevatedButton.styleFrom(
    backgroundColor: Colors.blueAccent,
    foregroundColor: Colors.white,
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(10),
    ),
  ),
  padding: EdgeInsets.symmetric(vertical: 12, horizontal: 40),
),
  child: Text(
    "Login",
    style: TextStyle(fontSize: 18),
  ),
),
 SizedBox(height: 15),
TextButton(
  onPressed: () => Get.to(() => Forgot()),
  child: Text(
    "Forgot Password?",

```

```
        style: TextStyle(color: Colors.blueAccent),
      ),
    ),
    Divider(thickness: 1, height: 30),
    TextButton(
      onPressed: () => Get.to(() => Signup()),
      child: Text(
        "Don't have an account? Sign Up now",
        style: TextStyle(color: Colors.blueAccent),
      ),
    ),
  ],
),
),
),
),
),
),
),
);
}
```

Output:



