

FLUTTER

TASK-1

Create a sign-in/sign-up page in Flutter which includes, building the user interface (UI) and implementing the authentication logic.

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

void main() {
  runApp(const MyApp());
}
class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return const MaterialApp(
      debugShowCheckedModeBanner: false,
      home: AuthPage(),
    );
  }
}

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

  @override
  State<AuthPage> createState() => _AuthPageState();
}
class _AuthPageState extends State<AuthPage> {
  final _formKey = GlobalKey<FormState>();
  final TextEditingController _nameController = TextEditingController();
  final TextEditingController _emailController = TextEditingController();
  final TextEditingController _passwordController = TextEditingController();

  bool isLogin = true;
  final Map<String, Map<String, String>> _users = {};
  void _submit() {
    if (_formKey.currentState!.validate()) {
      String name = _nameController.text.trim();
      String email = _emailController.text.trim();
      String password = _passwordController.text.trim();
      if (isLogin) {
        if (_users.containsKey(email) &&
            _users[email]!["password"] == password) {
          String savedName = _users[email]!["name"]!;
          _showMessage("✔ Login successful. Welcome back, $savedName!");
        } else {
          _showMessage("✘ Invalid email or password.");
        }
      }
    }
  }
}
```

```

    } else {
      if (_users.containsKey(email)) {
        _showMessage("⚠ User already exists. Please login.");
      } else {
        // Save new user
        _users[email] = {"name": name, "password": password};
        _showMessage("🎉 Signup successful! Welcome, $name!");
        setState() {
          isLogin = true;
        });
      }
    }
  }
}

void _showMessage(String message) {
  showDialog(
    context: context,
    builder: (_) => AlertDialog(
      title: Text(isLogin ? "Login" : "Signup"),
      content: Text(message),
      actions: [
        TextButton(
          onPressed: () => Navigator.pop(context),
          child: const Text("OK"),
        )
      ],
    ),
  );
}

InputDecoration myDecoration(String label, IconData icon) {
  return InputDecoration(
    labelText: label,
    prefixIcon: Icon(icon, color: Colors.deepPurple),
    border: OutlineInputBorder(
      borderRadius: BorderRadius.circular(12),
    ),
  );
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text(isLogin ? "Login Page" : "Signup Page"),
      backgroundColor: Colors.deepPurple,
    ),
    body: Padding(
      padding: const EdgeInsets.all(20),
      child: Form(
        key: _formKey,
        child: Column(

```

```

children: [
  if (!isLoggedIn) ...[
    TextFormField(
      controller: _nameController,
      decoration: myDecoration("Full Name", Icons.person),
      validator: (value) =>
        value!.isEmpty ? "Enter your name" : null,
    ),
    const SizedBox(height: 15),
  ],
  TextFormField(
    controller: _emailController,
    decoration: myDecoration("Email", Icons.email),
    validator: (value) =>
      value!.isEmpty ? "Enter email" : null,
  ),
  const SizedBox(height: 15),
  TextFormField(
    controller: _passwordController,
    obscureText: true,
    decoration: myDecoration("Password", Icons.lock),
    validator: (value) => value!.length < 6
      ? "Password must be at least 6 characters"
      : null,
  ),
  const SizedBox(height: 25),
  ElevatedButton(
    onPressed: _submit,
    style: ElevatedButton.styleFrom(
      backgroundColor: Colors.deepPurple,
      foregroundColor: Colors.white,
      minimumSize: const Size(double.infinity, 50),
    ),
  ),
  child: Text(isLoggedIn ? "Login" : "Signup"),
),
const SizedBox(height: 15),
TextButton(
  onPressed: () {
    setState(() {
      isLoggedIn = !isLoggedIn;
    });
  },
  child: Text(
    isLoggedIn
      ? "Don't have an account? Sign Up"
      : "Already have an account? Login",
    style: const TextStyle(color: Colors.deepPurple),
  ),
),
),
],


```


```
    },  
    },  
    },  
    );  
}  
}
```


OUTPUT:

Signup Page

Full Name




 Email

 Password

Signup

Already have an account? Login

Login Page

Email —
 durga@gmail.com

Password —


Login


🎉 Signup successful! Welcome, kanaga durga!

OK

Dart 3.9.0 • Flutter 3.35.2

🔧 Stable channel

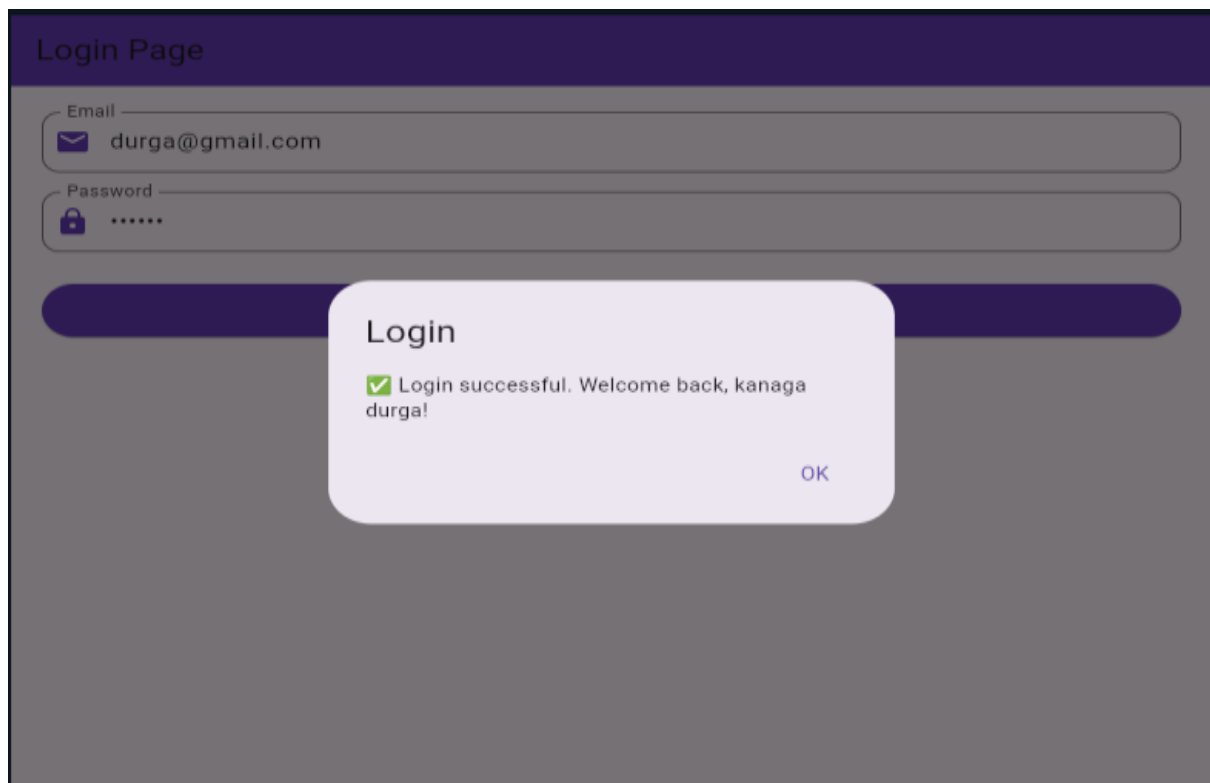
Login Page

Email —
 durga@gmail.com

Password —


Login

Don't have an account? Sign Up



Github link: <https://github.com/KanagaDurga-code/Flutter/tree/main>