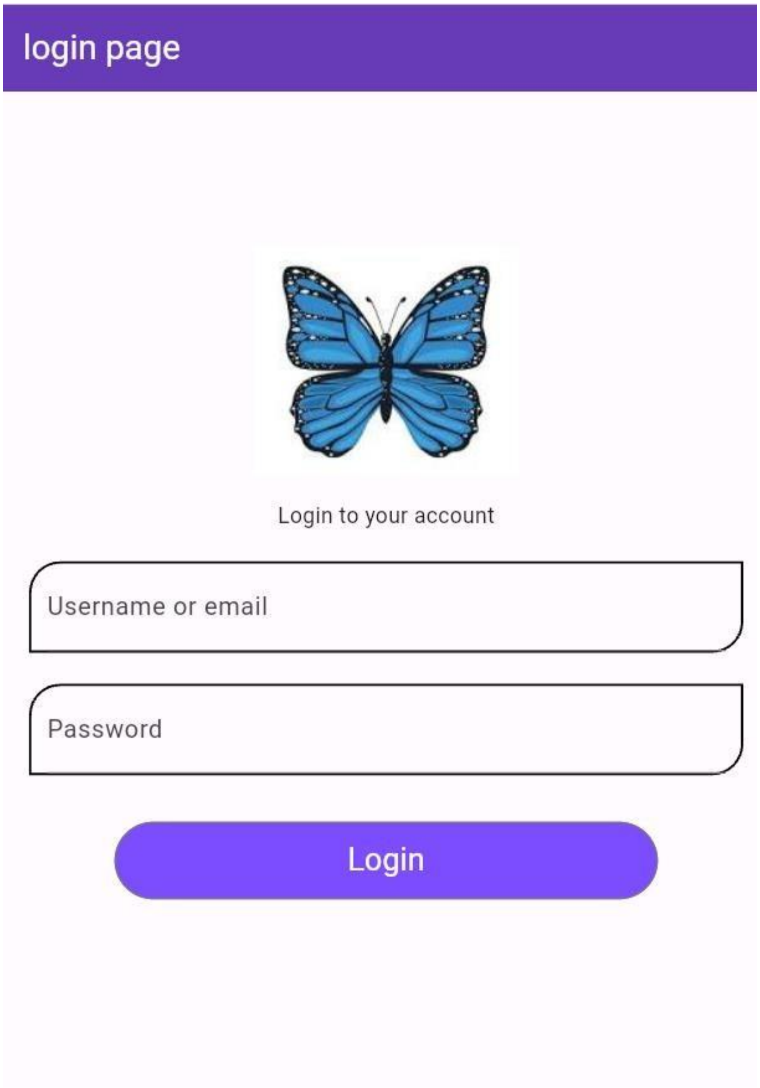
Practical 5 : Create and application with Flutter UI Components.

Main.dart

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

import 'LoginPage.dart';

void main() => runApp(MyApp());

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
    debugShowCheckedModeBanner: false,
    home: LoginPage(),
    );
  }
}

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

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("login page"),
        backgroundColor: Colors.deepPurple,
        foregroundColor: Colors.white,
        automaticallyImplyLeading: true,
      ),
      body: Container(
        /*alignment:Alignment.center ,
        decoration: BoxDecoration(
        gradient: LinearGradient(
            colors: [
              Colors.blueGrey,
              Colors.tealAccent
              ],
            begin: Alignment.topLeft,
            end: Alignment.topRight,
          )
        ),*/
        child: SingleChildScrollView(
          child: Column(
            children: [
              const SizedBox(height: 100),
              Image.asset("assets/img.png"),
              const SizedBox(height: 15),
              const Text("Login to your account"),
const SizedBox(height: 20),
Container(
  margin: const EdgeInsets.symmetric(horizontal: 20),
  padding: const EdgeInsets.symmetric(horizontal: 10),
  decoration: BoxDecoration(
    border: Border.all(color: Colors.black, width: 1.5),
    borderRadius: const BorderRadius.only(
      topLeft: Radius.circular(20),
         bottomRight: Radius.circular(20)),
   ),
  child: const TextField(
    //keyboardType: TextInputType.number,
    //
    decoration: InputDecoration(
      labelText: "Username or email",
     // border: OutlineInputBorder(),
    ),
   // spellCheckConfiguration: SpellCheckConfiguration(
     //  misspelledSelectionColor: Colors.red),
  ),
),
const SizedBox(height: 20),
Container(
  margin: const EdgeInsets.symmetric(horizontal: 20),
  padding: const EdgeInsets.symmetric(horizontal: 10),
  decoration: BoxDecoration(
    border: Border.all(color: Colors.black, width: 1.5),
    borderRadius: const BorderRadius.only(
        topLeft: Radius.circular(20),
        bottomRight: Radius.circular(20)),
  ),
  child: const TextField(
    obscureText: true,
    decoration: InputDecoration(
    labelText: "Password",

      // border: OutlineInputBorder(),
    ),
  ),
),
const SizedBox(height: 30),
Container(
  height: 50,
  width: 350,
  child: OutlinedButton(
    onPressed: () {},
    style: ButtonStyle(
      foregroundColor: MaterialStateProperty.all(Colors.white),
      backgroundColor:
      MaterialStateProperty.all(Colors.deepPurpleAccent),
    ),
                child: const Text(
                  "Login",
                  style: TextStyle(
                    fontSize: 20,
                  ),
                ),
              ),
            ),
          ],
        ),
      ),
    ),
  );
 }
}
```

Output:



login page

Login to your account

Username or email

Password

Login

Practical 6 : Create and application with Flutter UI Components.

Main.dart t

```
import 'package:flutter/material.dart';
import 'login_page.dart';
void main() {
  runApp(MyApp());
}
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
    debugShowCheckedModeBanner: false,
    home: LoginPage(),
    );
  }
}
```

login_page.dart

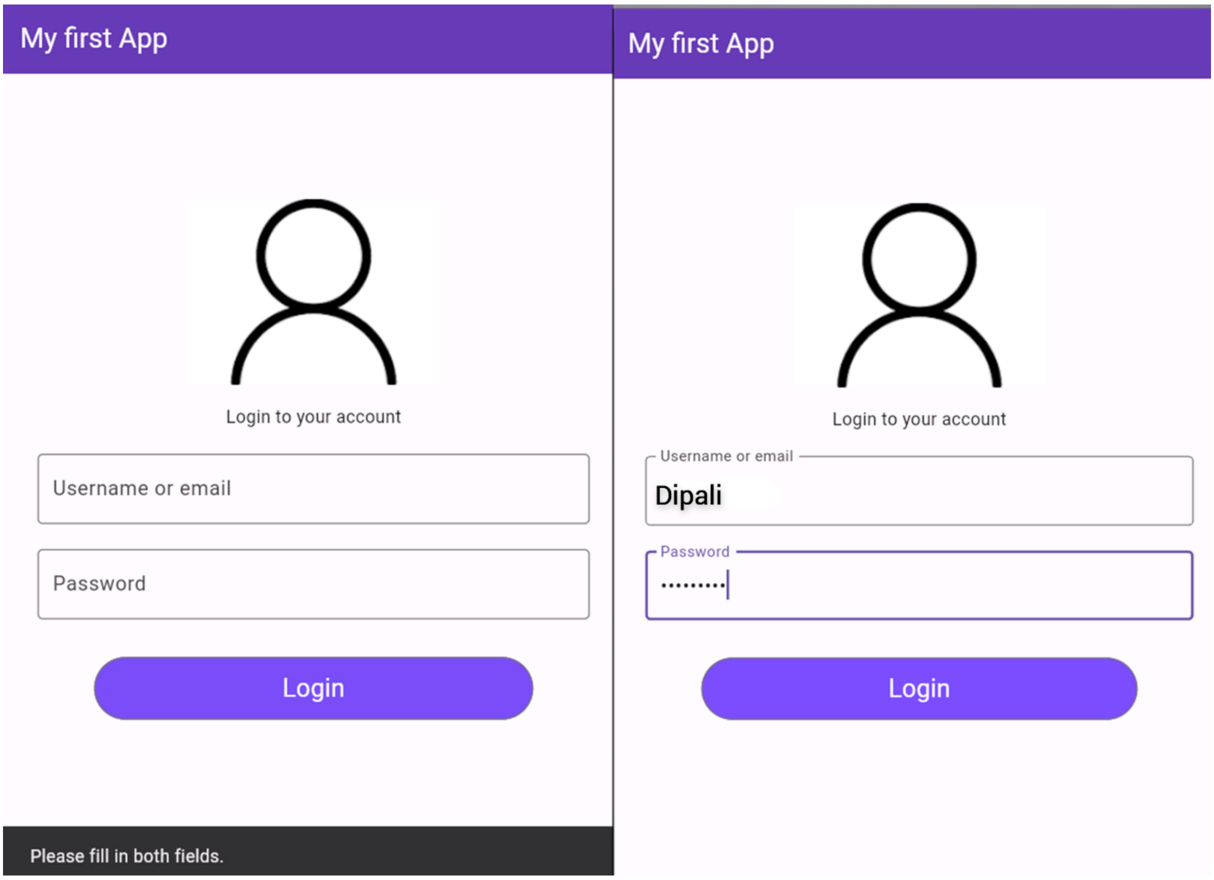
```
import 'package:flutter/material.dart';
import 'package:practical6/custom_gesture_page.dart';
class LoginPage extends StatefulWidget {
  const LoginPage({Key? key}) : super(key: key);
  @override
  _LoginPageState createState() => _LoginPageState();
}
class _LoginPageState extends State<LoginPage> {
  TextEditingController usernameController = TextEditingController();
  TextEditingController passwordController = TextEditingController();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("My first App"),
        backgroundColor: Colors.deepPurple,
        foregroundColor: Colors.white,
        automaticallyImplyLeading: true,
      ),
      body: Container(
        child: SingleChildScrollView(
          child: Column(
            children: [
              const SizedBox(height: 100),
              Image.asset("assets/user_logo.png"),
              const SizedBox(height: 15),
              const Text("Login to your account"),
              const SizedBox(height: 20),
Container(
  margin: const EdgeInsets.symmetric(horizontal: 20),
  padding: const EdgeInsets.symmetric(horizontal: 10),
  child: TextField(
    controller: usernameController,
    decoration: InputDecoration(
    labelText: "Username or email",
    border: OutlineInputBorder(),
    ),
  ),
),
const SizedBox(height: 20),
Container(
  margin: const EdgeInsets.symmetric(horizontal: 20),
  padding: const EdgeInsets.symmetric(horizontal: 10),
  child: TextField(
    controller: passwordController,
    obscureText: true,
    decoration: InputDecoration(
      labelText: "Password",
      border: OutlineInputBorder(),
    ),
  ),
),
const SizedBox(height: 30),
Container(
  height: 50,
  width: 350,
  child: OutlinedButton(
    onPressed: () {
      // Check if both username and password are filled
      if (usernameController.text.isNotEmpty &&
          passwordController.text.isNotEmpty) {
        // Navigate to CustomGesturePage
        Navigator.push(
          context,
          MaterialPageRoute(
            builder: (context) => CustomGesturePage(),
          ),
        );
      } else {
        // Show a message if fields are not filled
        ScaffoldMessenger.of(context).showSnackBar(
        const SnackBar(
            content: Text('Please fill in both fields.'),
          ),
        );
      }
    },
    style: ButtonStyle(
      foregroundColor: MaterialStateProperty.all(Colors.white),
      backgroundColor:
```

```
                    MaterialStateProperty.all(Colors.deepPurpleAccent),
                  ),
                  child: const Text(
                    "Login",
                    style: TextStyle(
                      fontSize: 20,
                    ),
                  ),
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

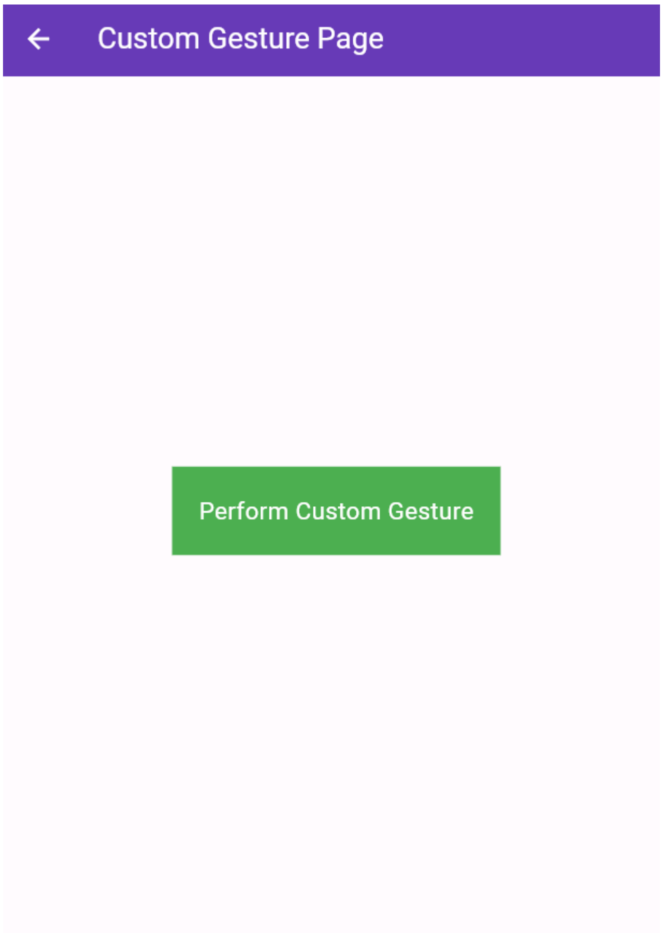
custom_gesture_page.dart

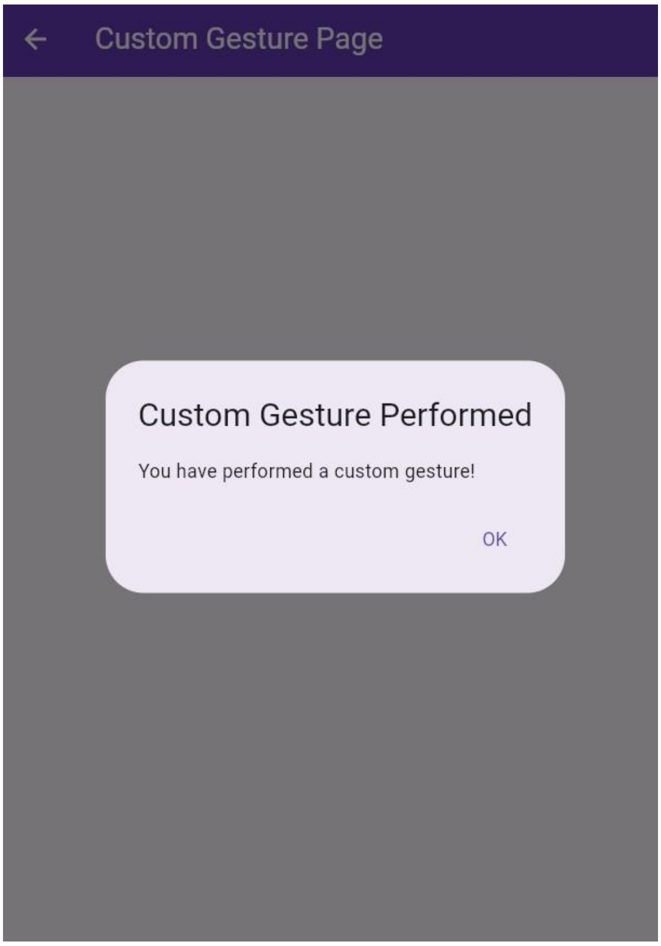
```
import 'package:flutter/material.dart';
class CustomGesturePage extends StatelessWidget {
  const CustomGesturePage({Key? key}) : super(key: key);
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text("Custom Gesture Page"),
        backgroundColor: Colors.deepPurple,
        foregroundColor: Colors.white,
        automaticallyImplyLeading: true,
      ),
      body: Center(
        child: GestureDetector(
          onTap: () {
            // Perform custom gesture action
            _showCustomGestureDialog(context);
          },
          child: Container(
            padding: const EdgeInsets.all(20),
            color: Colors.green,
            child: const Text(
              "Perform Custom Gesture",
              style: TextStyle(fontSize: 18, color: Colors.white),
            ),
          ),
        ),
      ),
    );
  }
  void _showCustomGestureDialog(BuildContext context) {
    showDialog(
      context: context,
      builder: (BuildContext context) {
        return AlertDialog(
          title: const Text("Custom Gesture Performed"),
          content: const Text("You have performed a custom gesture!"),
          actions: <Widget>[
            TextButton(
              onPressed: () {
                Navigator.of(context).pop();
              },
              child: const Text("OK"),
            ),
          ],
        );
      },
    );
  }
}
```

Output:



Please fill in both fields.





Custom Gesture Performed

You have performed a custom gesture!

Practical 7 : Create and application with Navigation in Flutter.

Main.dart

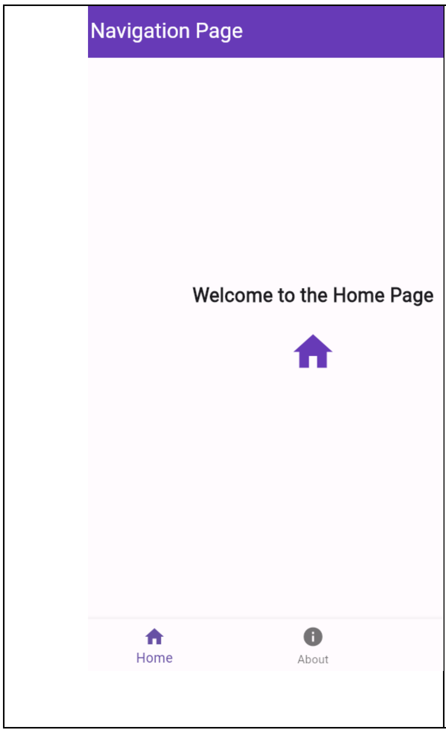
```
import 'package:flutter/material.dart';
import 'navigation_page.dart';
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
    debugShowCheckedModeBanner: false,
    home: NavigationPage(),
    );
  }
}
```

navigation_page.dart

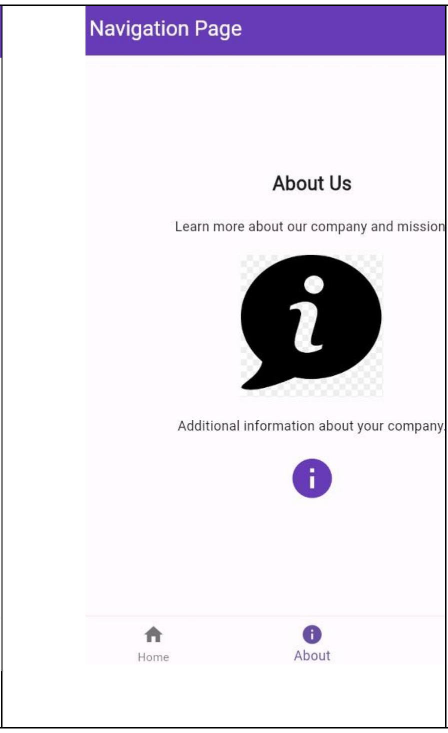
```
import 'package:flutter/material.dart';

class NavigationPage extends StatefulWidget {
  @override
  _NavigationPageState createState() => _NavigationPageState();
}

class _NavigationPageState extends State<NavigationPage> {
  int _currentIndex = 0;

  final List<Widget> _pages = [
    HomePage(),
    AboutPage(),
    ContactPage(),
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Navigation Page"),
        backgroundColor: Colors.deepPurple,
        foregroundColor: Colors.white,
      ),
      body: _pages[_currentIndex],
      bottomNavigationBar: BottomNavigationBar(
      currentIndex: _currentIndex,
        onTap: (index) {
```

```
setState(() {
                  _currentIndex = index;
});
                },
                items: [
BottomNavigationBarItem(icon: Icon(Icons.home), label: "Home"),
BottomNavigationBarItem(icon: Icon(Icons.info), label: "About"),
BottomNavigationBarItem(
                    icon: Icon(Icons.contact_mail), label: "Contact"),
                ],
              ),
            );
          }
        }

        // ... (existing code)

        class HomePage extends StatelessWidget {
          @override
          Widget build(BuildContext context) {
            return Center(
              child: Column(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
Text("Welcome to the Home Page",
   style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold)),
SizedBox(height: 20),
Icon(Icons.home, size: 50, color: Colors.deepPurple),
SizedBox(height: 20),

                ],
              ),
            );
          }
        }

        class AboutPage extends StatelessWidget {
          @override
          Widget build(BuildContext context) {
            return Center(
              child: Column(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
Text("About Us",
   style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold)),
SizedBox(height: 20),
Text("Learn more about our company and mission."),
SizedBox(height: 20),
Image.asset(
                  'assets/about_image.png', // Add the path to your image asset
                  width: 150,
                         height: 150,
),
SizedBox(height: 20),
Text("Additional information about your company."),
SizedBox(height: 20),
Icon(Icons.info, size: 50, color: Colors.deepPurple),
                      ],
                    ),
                  );
                }
              }

              class ContactPage extends StatelessWidget {
                @override
                Widget build(BuildContext context) {
                  return Center(
                    child: Column(
                      mainAxisAlignment: MainAxisAlignment.center,
                      children: [
Text("Contact Us",
    style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold)),
SizedBox(height: 20),
Text("Reach out to us for any inquiries or support."),
SizedBox(height: 20),
TextFormField(
                         decoration: InputDecoration(
                           labelText: 'Your Name',
                         ),
),
SizedBox(height: 10),
TextFormField(
                         decoration: InputDecoration(
                           labelText: 'Email Address',
                         ),
),
SizedBox(height: 20),
ElevatedButton( onPressed: () {
                           // Add your action for the form submission
                         },
                         child: Text('Submit'),
),
SizedBox(height: 20),
Icon(Icons.contact_mail, size: 50, color: Colors.deepPurple),
                      ],
                    ),
                  );
                }
              }
```

Output:


Navigation Page

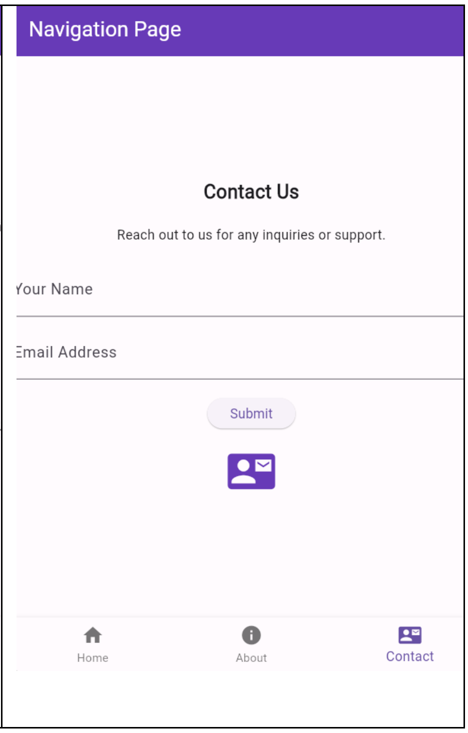
Welcome to the Home Page


Navigation Page

About Us

Learn more about our company and mission

Additional information about your company


Navigation Page

Contact Us

Reach out to us for any inquiries or support.

Practical 8 : Create and application with list view in Flutter.

Main.dart

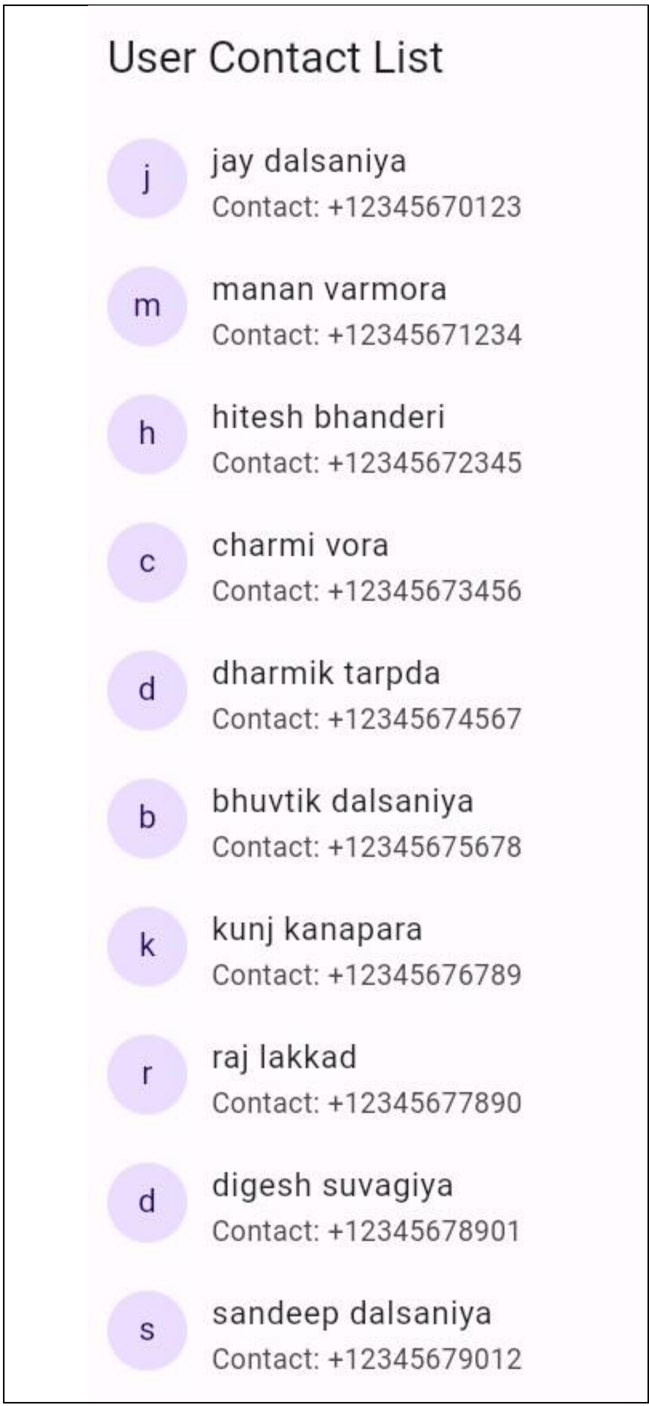
```
          import 'package:flutter/material.dart';
          import 'list_view.dart';void main() {
          runApp(MyApp());
          }

          class MyApp extends StatelessWidget {
            @override
Widget build(BuildContext context) { return
  MaterialApp( debugShowCheckedModeBanner:
  false, home: MyListView (),
  );
            }
          }
```

list_view.dart

```
          import 'package:flutter/material.dart';
          class MyListView extends StatelessWidget {
            final List<Map<String, dynamic>> items = [
  {
                'name': 'jay dalsaniya',
                'contactNumber': '+12345670123',
  },
  {
                'name': 'manan varmora',
                'contactNumber': '+12345671234',
  },
  {
                'name': 'hitesh bhanderi',
                'contactNumber': '+12345672345',
  },
  {
                'name': 'charmi vora',
                'contactNumber': '+12345673456',
  },
  {
                'name': 'dharmik tarpda',
                'contactNumber': '+12345674567',
  },
  {
                'name': 'bhuvtik dalsaniya',
                'contactNumber': '+12345675678',
  },
  {
                'name': 'kunj kanapara',
            'contactNumber': '+12345676789',
},
{
            'name': 'raj lakkad',
            'contactNumber': '+12345677890',
},
{
            'name': 'digesh suvagiya',
            'contactNumber': '+12345678901',
},
{
            'name': 'sandeep dalsaniya',
            'contactNumber': '+12345679012',
},
{
            'name': 'yash gandhi',
            'contactNumber': '+12345671234',
},
{
            'name': 'abhay gadara',
            'contactNumber': '+12345672345',
},
{
            'name': 'shubham dalsaniya',
            'contactNumber': '+12345673456',
},
{
            'name': 'jay mer',
            'contactNumber': '+12345674567',
},
{
            'name': 'sandeep chuhan',
            'contactNumber': '+12345675678',
},
{
            'name': 'yash kanani',
            'contactNumber': '+12345676789',
},
{
            'name': 'deep sanghani',
            'contactNumber': '+12345677890',
},
{
            'name': 'haresh godhani',
            'contactNumber': '+12345678901',
},
{
            'name': 'jasmin virmgama',
            'contactNumber': '+12345679012',
},
{
```

```
           'name': 'deep boda',
           'contactNumber': '+12345671234',
 },
 {
           'name': 'rutvik dalsaniya',
           'contactNumber': '+12345672345',
 },
 {
           'name': 'prit jogani',
           'contactNumber': '+12345673456',
 },
 {
           'name': 'harshad vanaliya',
           'contactNumber': '+12345674567',
 },
 {
           'name': 'raj padaliya',
           'contactNumber': '+12345675678',
 },
 // Add more users as needed
         ];

         @override
Widget build(BuildContext context) {return
 Scaffold(
           appBar: AppBar(
            title: Text('User Contact List'),
           ),
           body: ListView.builder(
            itemCount: items.length,
            itemBuilder: (context, index) {
            return ListTile(
              leading: CircleAvatar(
               child: Text(items[index]['name'][0]), // Display first letter of name
              ),
              title: Text(items[index]['name']),
              subtitle: Text('Contact: ${items[index]['contactNumber']}'),
              // Add any additional widgets here
             );
            },
           ),
 );
         }
        }
```

Output:



User Contact List

j jay dalsaniya
Contact: +12345670123

m manan varmora
Contact: +12345671234

h hitesh bhanderi
Contact: +12345672345

c charmi vora
Contact: +12345673456

d dharmik tarpda
Contact: +12345674567

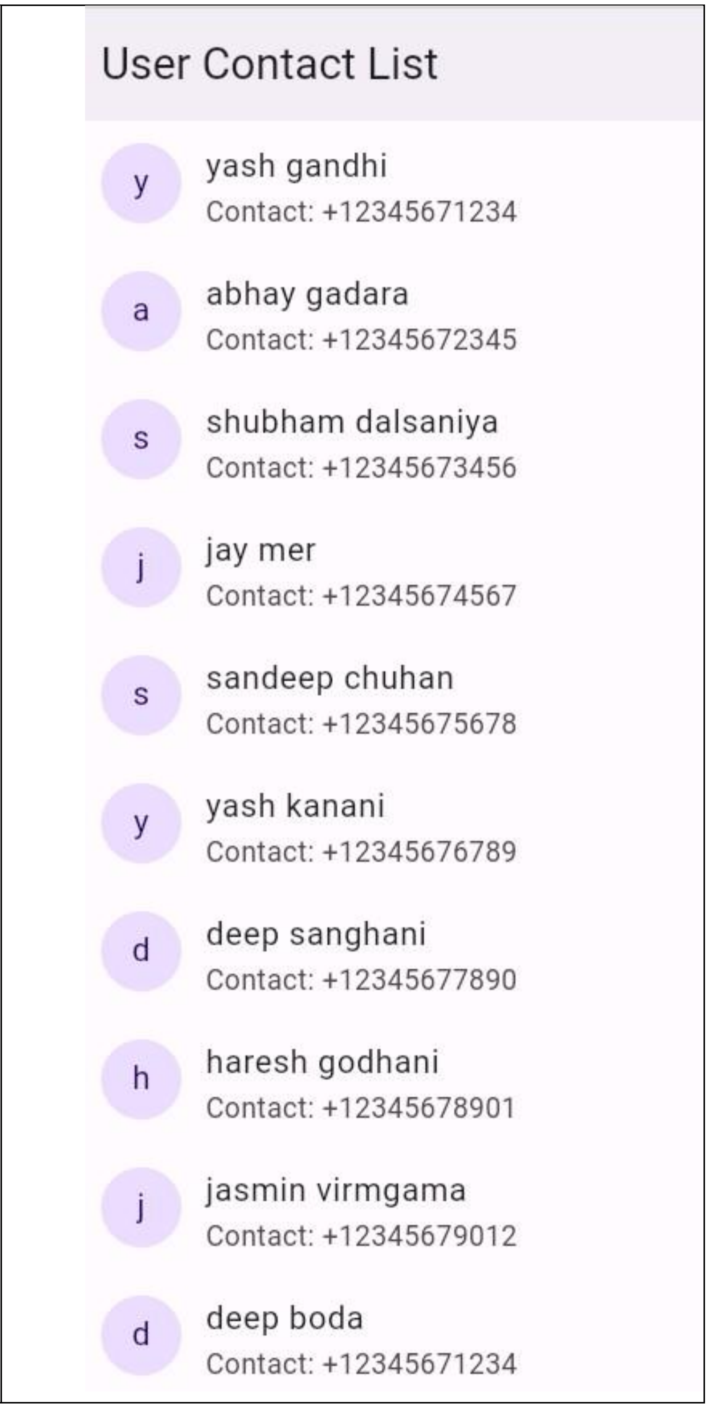
b bhuvtik dalsaniya
Contact: +12345675678

k kunj kanapara
Contact: +12345676789

r raj lakkad
Contact: +12345677890

d digesh suvagiya
Contact: +12345678901

s sandeep dalsaniya
Contact: +12345679012



User Contact List

y yash gandhi
Contact: +12345671234

a abhay gadara
Contact: +12345672345

s shubham dalsaniya
Contact: +12345673456

j jay mer
Contact: +12345674567

s sandeep chuhan
Contact: +12345675678

y yash kanani
Contact: +12345676789

d deep sanghani
Contact: +12345677890

h haresh godhani
Contact: +12345678901

j jasmin virmgama
Contact: +12345679012

d deep boda
Contact: +12345671234

Practical 9 : Create and application with grid view in Flutter.

Main.dart

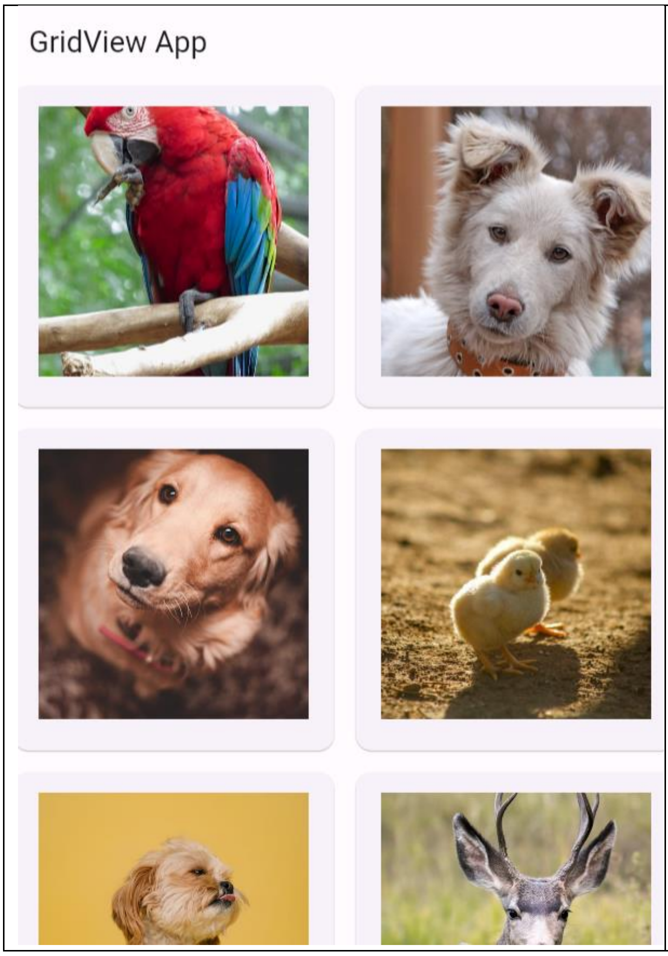
```
import 'package:flutter/material.dart';
import 'grid_view.dart';
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: MyGridView (),
    );
  }
}
```

grid_view.dart

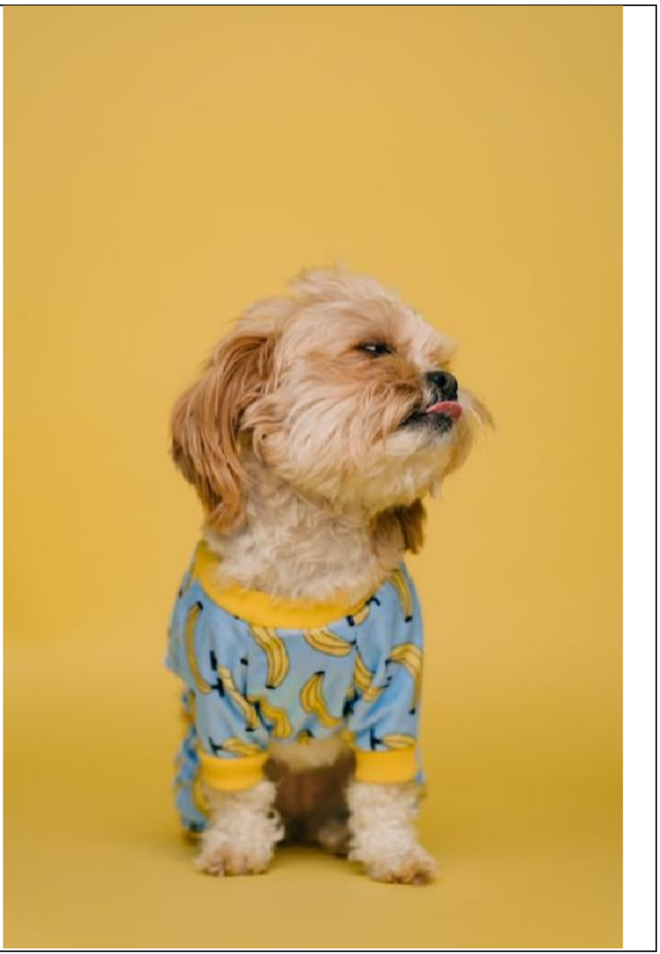
```
import 'package:flutter/material.dart';
import 'package:photo_view/photo_view.dart';

class MyGridView extends StatelessWidget {
  final List<String> items = List.generate(20, (index) => 'Item $index');
  final List<String> imageUrls = [
    "https://images.unsplash.com/photo-1604457407295-
8aa34e462dcf?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8OTh8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1587027077233-
c7a2e15825cf?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTAxfHx8ZW58MHx8fHx8",
    "https://images.unsplash.com/photo-1526489550178-
7bd5d9944f4f?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTcyfHx8ZW58MHx8fHx8",
    "https://images.unsplash.com/photo-1546272989-
40c92939c6c2?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8Njd8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1583511655802-
41f2ccc2cc8f?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8NDV8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1476922027627-
aa7293e3aaa8?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8NDN8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1470688090067-
6d429c0b2600?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8NDB8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1594031633878-
```

```
c59f0c8c16fd?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8Mzh8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1559214369-
a6b1d7919865?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTA5fHx8ZW58MHx8fHx8",
    "https://images.unsplash.com/photo-1578326626553-
39f72c545b07?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8Mjl8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1562569633-
622303bafef5?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MzB8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1536146021566-
627ce3c4d813?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MjV8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1547269646-
2e1478ba0333?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTV8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1573595287099-
eabfdb20ec6a?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTl8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1597953601374-
1ff2d5640c85?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MjN8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1612372992401-
ca1c6cebf839?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTE2fHx8ZW58MHx8fHx8",
    "https://images.unsplash.com/photo-1600252278397-
5fe47c86d3aa?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTF8fHxlbnwwfHx8fHw%3D",
    "https://images.unsplash.com/photo-1590787996529-
a542c86ca267?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MTB8fHxlbnwwfHx8fHw%3D",
    'https://images.unsplash.com/photo-1565413294262-
fa587c396965?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8MXx8fGVufDB8fHx8fA%3D%3D',
    'https://images.unsplash.com/photo-1623944864235-
db595bfccaad?w=500&auto=format&fit=crop&q=60&ixlib=rb-
4.0.3&ixid=M3wxMjA3fDB8MHxleHBsb3JlLWZlZWR8Nnx8fGVufDB8fHx8fA%3D%3D',
    // Add more image URLs as needed
  ];

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('GridView App'),
      ),
      body: GridView.builder(
        gridDelegate: SliverGridDelegateWithFixedCrossAxisCount(
          crossAxisCount: 2,
          crossAxisSpacing: 8.0,
          mainAxisSpacing: 8.0,
        ),
        itemCount: items.length,
        itemBuilder: (context, index) {
          return GestureDetector(
            onTap: () {
              Navigator.push(
                context,
                MaterialPageRoute(
                  builder: (context) => ZoomableImageView(
                    imageUrl: imageUrls[index % imageUrls.length],
                  ),
                ),
              );
            },
            child: Card(
              child: Column(
                mainAxisAlignment: MainAxisAlignment.center,
                children: [
                  Image.network(
                    imageUrls[index % imageUrls.length],
                    width: 200.0,
                    height: 200.0,
                    fit: BoxFit.cover,
                  ),
                  SizedBox(height: 8.0),
                  //Text(items[index]),
                ],
              ),
            ),
          );
        },
      ),
    );
  }
}

class ZoomableImageView extends StatelessWidget {
  final String imageUrl;

  ZoomableImageView({required this.imageUrl});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: GestureDetector(
        onTap: () {
          Navigator.pop(context);
        },
        child: Container(
          color: Colors.black,
          alignment: Alignment.center,
          child: PhotoView(
            imageProvider: NetworkImage(imageUrl),
            backgroundDecoration: BoxDecoration(
              color: Colors.black,
            ),
          ),
        ),
      ),
    );
  }
}
```

Output:



GridView App



Practical 10 : Create and application Crud Operation with SQLite in Flutter.

Main.dart

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

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

class MyApp extends StatelessWidget {
  const MyApp({Key? key}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      // Remove the debug banner
        debugShowCheckedModeBanner: false,
        title: 'SQLITE',
        theme: ThemeData(
          primarySwatch: Colors.orange,
        ),
        home: const HomePage());
  }
}

class HomePage extends StatefulWidget {
  const HomePage({Key? key}) : super(key: key);

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

class _HomePageState extends State<HomePage> {
  // All journals
  List<Map<String, dynamic>> _journals = [];

  bool _isLoading = true;
  // This function is used to fetch all data from the database
  void _refreshJournals() async {
    final data = await SQLHelper.getItems();
    setState(() {
      _journals = data;
      _isLoading = false;
    });
  }

  @override
  void initState() {
    super.initState();
  _refreshJournals(); // Loading the diary when the app starts
}

final TextEditingController _titleController = TextEditingController();
final TextEditingController _descriptionController = TextEditingController();

// This function will be triggered when the floating button is pressed
// It will also be triggered when you want to update an item
void _showForm(int? id) async {
  if (id != null) {
    // id == null -> create new item
    // id != null -> update an existing item
    final existingJournal =
    _journals.firstWhere((element) => element['id'] == id);
    _titleController.text = existingJournal['title'];
    _descriptionController.text = existingJournal['description'];
  }

  showModalBottomSheet(
      context: context,
      elevation: 5,
      isScrollControlled: true,
      builder: (_) => Container(
        padding: EdgeInsets.only(
          top: 15,
          left: 15,
          right: 15,
          // this will prevent the soft keyboard from covering the text fields
          bottom: MediaQuery.of(context).viewInsets.bottom + 120,
        ),
        child: Column(
          mainAxisSize: MainAxisSize.min,
          crossAxisAlignment: CrossAxisAlignment.end,
          children: [
            TextField(
              controller: _titleController,
              decoration: const InputDecoration(hintText: 'Title'),
            ),
            const SizedBox(
              height: 10,
            ),
            TextField(
              controller: _descriptionController,
              decoration: const InputDecoration(hintText: 'Description'),
            ),
            const SizedBox(
              height: 20,
            ),
            ElevatedButton(
              onPressed: () async {
                // Save new journal
```

```
                    if (id == null) {
                      await _addItem();
                    }

                    if (id != null) {
                      await _updateItem(id);
                    }

                    // Clear the text fields
                    _titleController.text = '';
                    _descriptionController.text = '';
                  )
                ],
              ),
            ));
          }// Close the bottom sheet Navigator.of(context).pop();
},
child: Text(id == null ? 'Create New' : 'Update'),

          // Insert a new journal to the database
            Future<void> _addItem() async {
            await SQLHelper.createItem(
                  _titleController.text, _descriptionController.text);
              _refreshJournals();
            }

            // Update an existing journal
            Future<void> _updateItem(int id) async {
            await SQLHelper.updateItem(
                  id, _titleController.text, _descriptionController.text);
              _refreshJournals();
            }

            // Delete an item
            void _deleteItem(int id) async {
              await SQLHelper.deleteItem(id);
              ScaffoldMessenger.of(context).showSnackBar(const SnackBar(
                content: Text('Successfully deleted a journal!'),
              ));
              _refreshJournals();
            }

            @override
            Widget build(BuildContext context) {
              return Scaffold(
                appBar: AppBar(
                  title: const Text('SQL'),
                ),
                body: _isLoading
                    ? const Center(
                  child: CircularProgressIndicator(),
   )
     : ListView.builder(
    itemCount: _journals.length,
    itemBuilder: (context, index) => Card(
     color: Colors.orange[200],
     margin: const EdgeInsets.all(15),
     child: ListTile(
       title: Text(_journals[index]['title']),
       subtitle: Text(_journals[index]['description']),
       trailing: SizedBox(
        width:  100,
        child: Row(
        children: [
          IconButton(
           icon: const Icon(Icons.edit),
           onPressed: () => _showForm(_journals[index]['id']),
          ),
          IconButton(
           icon: const Icon(Icons.delete),
           onPressed: () =>
             _deleteItem(_journals[index]['id']),
          ),
         ],
        ),
       )),
    ),
   ),
   floatingActionButton: FloatingActionButton(
    child: const Icon(Icons.add),
    onPressed: () => _showForm(null),
   ),
  );
 }
}
```

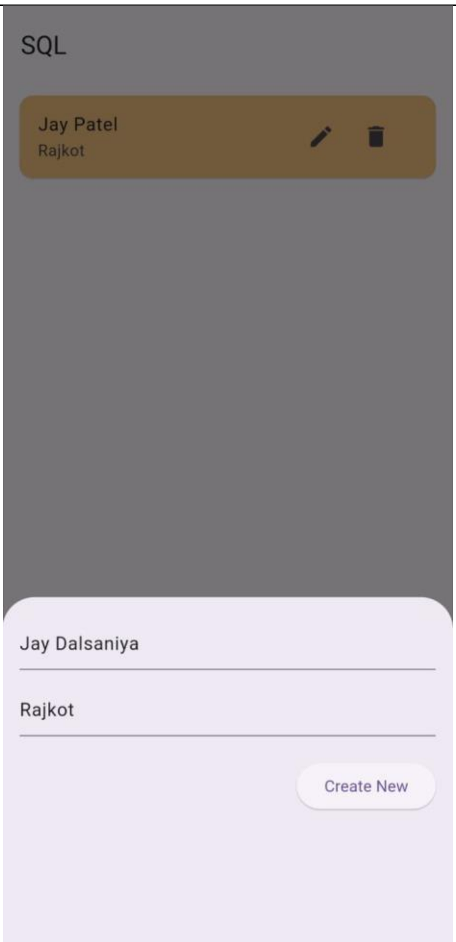
sqlHelper.dart

```
import 'package:flutter/foundation.dart';
import 'package:sqflite/sqflite.dart' as sql;

class SQLHelper {
 static Future<void> createTables(sql.Database database) async {
  await database.execute("""CREATE TABLE items(
    id INTEGER PRIMARY KEY AUTOINCREMENT NOT NULL,
    title TEXT,
    description TEXT,
    createdAt TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP
   )
```

```
      """);
  }
// id: the id of a item
// title, description: name and description of your activity
// created_at: the time that the item was created. It will be automatically handled by SQLite

  static Future<sql.Database> db() async {
    return sql.openDatabase(
      'dbtech.db',
      version: 1,
      onCreate: (sql.Database database, int version) async {
        await createTables(database);
      },
    );
  }

  // Create new item (journal)
  static Future<int> createItem(String title, String? descrption) async {
    final db = await SQLHelper.db();

    final data = {'title': title, 'description': descrption};
    final id = await db.insert('items', data,
        conflictAlgorithm: sql.ConflictAlgorithm.replace);
    return id;
  }

  // Read all items (journals)
  static Future<List<Map<String, dynamic>>> getItems() async {
    final db = await SQLHelper.db();
    return db.query('items', orderBy: "id");
  }

  // Read a single item by id
  // The app doesn't use this method but I put here in case you want to see it
  static Future<List<Map<String, dynamic>>> getItem(int id) async {
    final db = await SQLHelper.db();
    return db.query('items', where: "id = ?", whereArgs: [id], limit: 1);
  }

  // Update an item by id
  static Future<int> updateItem(
      int id, String title, String? descrption) async {
    final db = await SQLHelper.db();

    final data = {
      'title': title,
      'description': descrption,
      'createdAt': DateTime.now().toString()
    };

    final result =
    await db.update('items', data, where: "id = ?", whereArgs: [id]);
    return result;
  }

  // Delete
  static Future<void> deleteItem(int id) async {
    final db = await SQLHelper.db();
    try {
      await db.delete("items", where: "id = ?", whereArgs: [id]);
    } catch (err) {
      debugPrint("Something went wrong when deleting an item: $err");
    }
  }
}
```

```
dependencies:
  flutter:
    sdk: flutter
  sqflite: ^2.0.0
  path: ^1.9.0
  path_provider: any
```

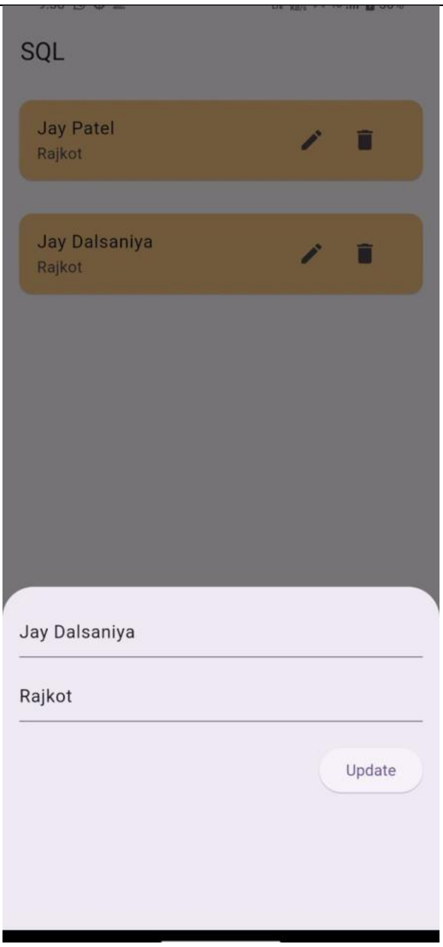
Output:


SQL

Jay Patel
Rajkot

Jay Dalsaniya

Rajkot

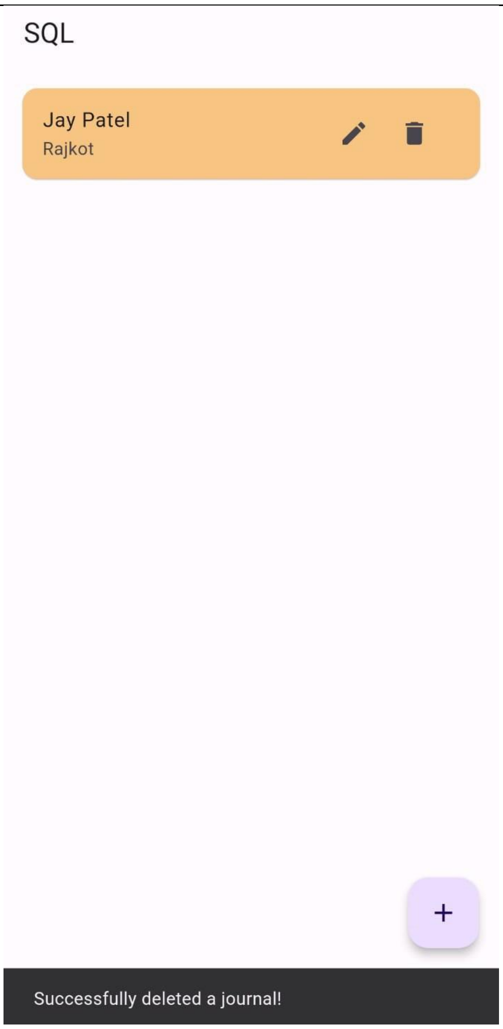

SQL

Jay Patel
Rajkot

Jay Dalsaniya
Rajkot

Jay Dalsaniya

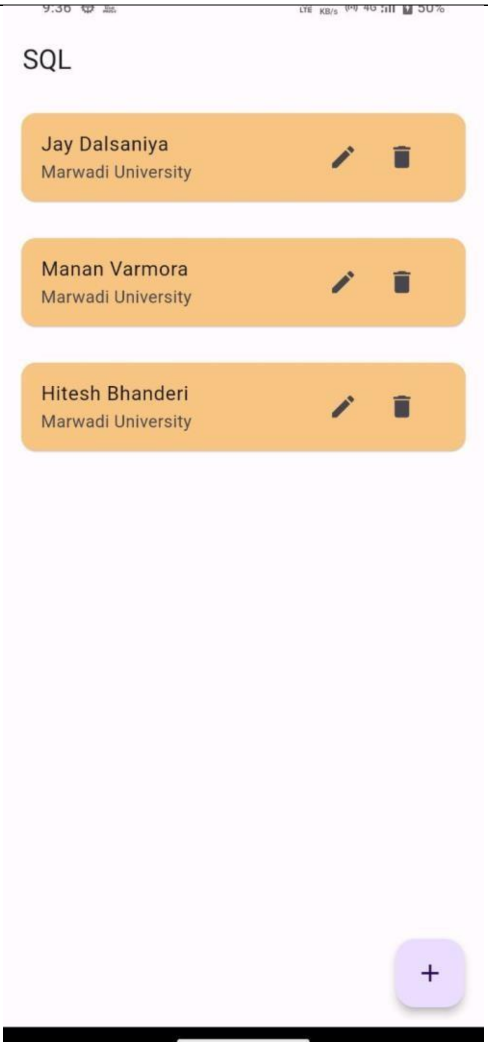
Rajkot



SQL

Jay Patel
Rajkot

Successfully deleted a journal!



SQL

Jay Dalsaniya
Marwadi University

Manan Varmora
Marwadi University

Hitesh Bhanderi
Marwadi University

Practical 11 : Create and application Connecting to REST API in Flutter

Main.dart

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

void main() {
runApp(MyApp());
}
class MyApp extends StatelessWidget {
@override
Widget build(BuildContext context) {
return MaterialApp(
debugShowCheckedModeBanner: false,
title: 'Flutter REST API Demo',
theme: ThemeData(
primarySwatch: Colors.blue,
),
home: DataScreen(),
);
}
}
```

api_service.dart

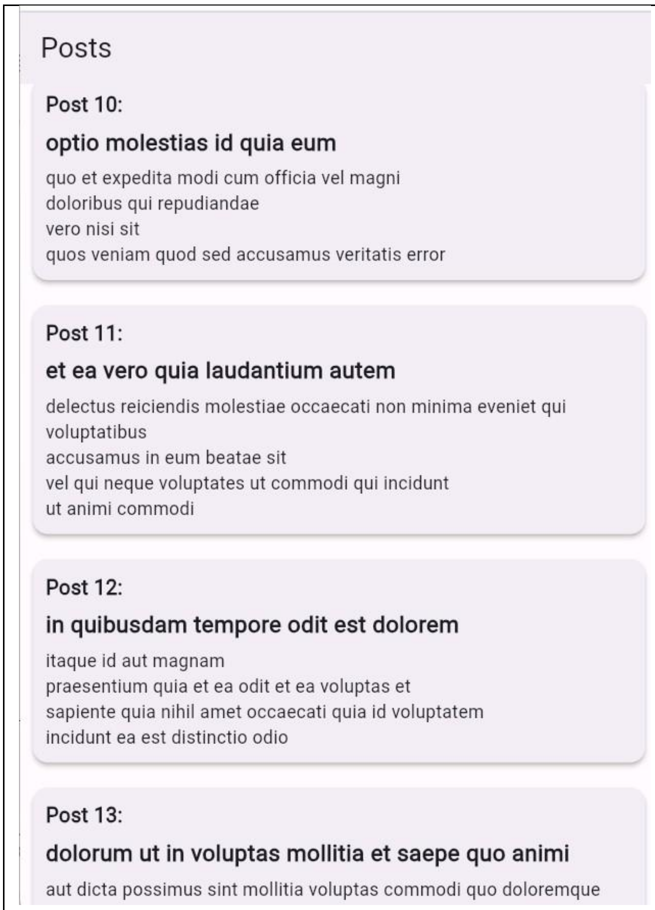
```
import 'dart:convert';
import 'package:http/http.dart' as http;

class Post  {
  final int userId;
  final int id;
  final String title;
  final String body;

  Post({
    required this.userId,
    required this.id,
    required this.title,
    required this.body,
  });

  factory Post.fromJson(Map<String, dynamic> json) {
    return Post(
      userId: json['userId'],
      id: json['id'],
      title: json['title'],
      body: json['body'],
    );
  }
}

class ApiService {
  static const String baseUrl = 'https://jsonplaceholder.typicode.com/todos/1';

  static Future<List<Post>> fetchPosts() async {
    final response = await http.get(Uri.parse('$baseUrl/posts'));

    if (response.statusCode == 200) {
      List<dynamic> jsonResponse = json.decode(response.body);
      return jsonResponse.map((post) => Post.fromJson(post)).toList();
    } else {
      throw Exception('Failed to load posts');
    }
  }
}
```

data_screen.dart

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

class DataScreen extends StatefulWidget {
  @override
  _DataScreenState createState() => _DataScreenState();
}

class _DataScreenState extends State<DataScreen> {
  late Future<List<Post>> posts;

  @override
  void initState() {
    super.initState();
    posts = ApiService.fetchPosts();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Posts'),
      ),
      body: Center(
        child: FutureBuilder<List<Post>>(
          future: posts,
          builder: (context, snapshot) {
            if (snapshot.hasData) {
            return ListView.builder(
                itemCount: snapshot.data!.length,
                itemBuilder: (context, index) {
                return Card(
                    elevation: 3,
                    margin: EdgeInsets.all(10),
                    child: Padding(
                      padding: EdgeInsets.all(10),
                      child: Column(
                        crossAxisAlignment: CrossAxisAlignment.start,
                        children: [
                          Text(
                            'Post ${index + 1}:', // Add label here
                            style: TextStyle(
                              fontWeight: FontWeight.bold,
                              fontSize: 16,
                            ),
                          ),
                          SizedBox(height: 5),
                          Text(
                            snapshot.data![index].title,
                            style: TextStyle(
                              fontWeight: FontWeight.bold,
                              fontSize: 18,
                            ),
                          ),
                          SizedBox(height: 5),
                          Text(snapshot.data![index].body),
                        ],
                      ),
                    ),
                  );
                },
              );
            } else if (snapshot.hasError) { return
              Text("${snapshot.error}");
            }

            // By default, show a loading spinner.
            return CircularProgressIndicator();
          },
        ),
      ),
    );
  }
}
```

```
dev_dependencies:
  flutter_test:
    sdk: flutter http:
  ^0.13.
```

Output:



Posts

Post 10:

optio molestias id quia eum

quo et expedita modi cum officia vel magni
doloribus qui repudiandae
vero nisi sit
quos veniam quod sed accusamus veritatis error

Post 11:

et ea vero quia laudantium autem

delectus reiciendis molestiae occaecati non minima eveniet qui
voluptatibus
accusamus in eum beatae sit
vel qui neque voluptates ut commodi qui incidunt
ut animi commodi

Post 12:

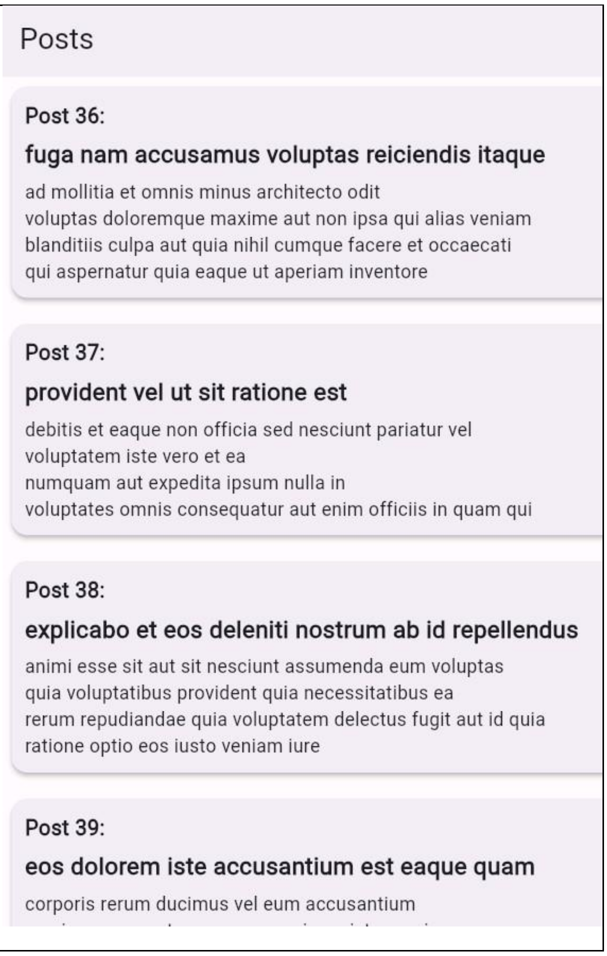
in quibusdam tempore odit est dolorem

itaque id aut magnam
praesentium quia et ea odit et ea voluptas et
sapiente quia nihil amet occaecati quia id voluptatem
incidunt ea est distinctio odio

Post 13:

dolorum ut in voluptas mollitia et saepe quo animi

aut dicta possimus sint mollitia voluptas commodi quo doloremque



Posts

Post 36:

fuga nam accusamus voluptas reiciendis itaque

ad mollitia et omnis minus architecto odit
voluptas doloremque maxime aut non ipsa qui alias veniam
blanditiis culpa aut quia nihil cumque facere et occaecati
qui aspernatur quia eaque ut aperiam inventore

Post 37:

provident vel ut sit ratione est

debitis et eaque non officia sed nesciunt pariatur vel
voluptatem iste vero et ea
numquam aut expedita ipsum nulla in
voluptates omnis consequatur aut enim officiis in quam qui

Post 38:

explicabo et eos deleniti nostrum ab id repellendus

animi esse sit aut sit nesciunt assumenda eum voluptas
quia voluptatibus provident quia necessitatibus ea
rerum repudiandae quia voluptatem delectus fugit aut id quia
ratione optio eos iusto veniam iure

Post 39:

eos dolorem iste accusantium est eaque quam

corporis rerum ducimus vel eum accusantium

Practical 12 : Create and application Parsing JSON data from REST API in Flutter.

Main.dart

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

void main() {
runApp(MyApp());
}
class MyApp extends StatelessWidget {
@override
Widget build(BuildContext context) {
return MaterialApp(
debugShowCheckedModeBanner: false,
title: 'Flutter REST API Demo',
theme: ThemeData(
primarySwatch: Colors.blue,
),
home: DataScreen(),
);
}
}
```

api_service.dart

```
import 'dart:convert';
import 'package:http/http.dart' as http;

class Post  {
  final int userId;
  final int id;
  final String title;
  final String body;

  Post({
    required this.userId,
    required this.id,
    required this.title,
    required this.body,
  });

  factory Post.fromJson(Map<String, dynamic> json) {
    return Post(
      userId: json['userId'],
      id: json['id'],
      title: json['title'],
      body: json['body'],
    );
  }
}

class ApiService {
  static const String baseUrl = 'https://jsonplaceholder.typicode.com/todos/1';

  static Future<List<Post>> fetchPosts() async {
    final response = await http.get(Uri.parse('$baseUrl/posts'));

    if (response.statusCode == 200) {
      List<dynamic> jsonResponse = json.decode(response.body);
      return jsonResponse.map((post) => Post.fromJson(post)).toList();
    } else {
      throw Exception('Failed to load posts');
    }
  }
}
```

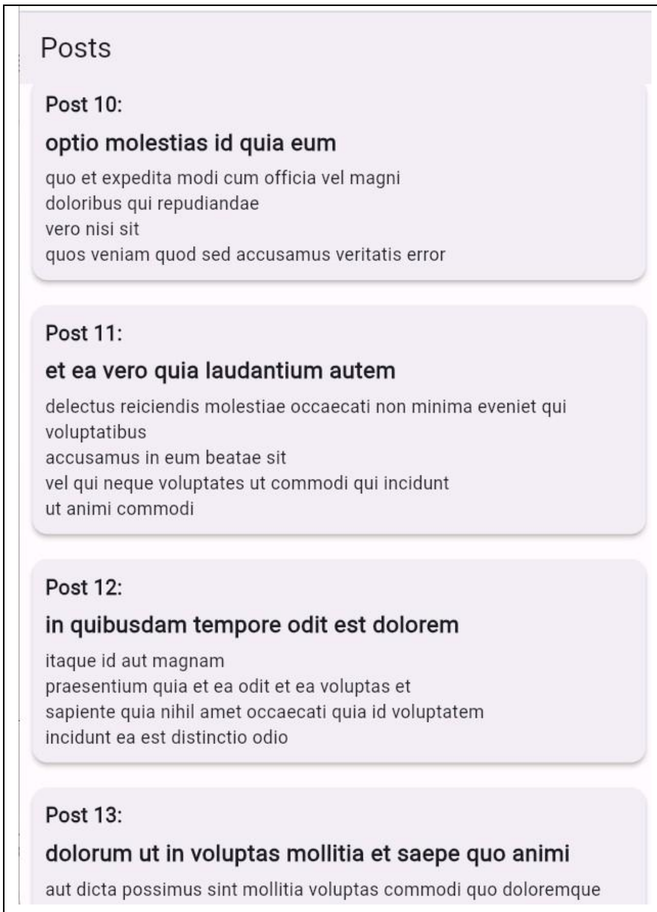
data_screen.dart

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

class DataScreen extends StatefulWidget {
  @override
  _DataScreenState createState() => _DataScreenState();
}

class _DataScreenState extends State<DataScreen> {
  late Future<List<Post>> posts;

  @override
  void initState() {
    super.initState();
    posts = ApiService.fetchPosts();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Posts'),
      ),
      body: Center(
        child: FutureBuilder<List<Post>>(
          future: posts,
          builder: (context, snapshot) {
            if (snapshot.hasData) {
            return ListView.builder(
                itemCount: snapshot.data!.length,
                itemBuilder: (context, index) {
                  return Card(
                    elevation: 3,
                    margin: EdgeInsets.all(10),
                    child: Padding(
                      padding: EdgeInsets.all(10),
                      child: Column(
                        crossAxisAlignment: CrossAxisAlignment.start,
                        children: [
                          Text(
                            'Post ${index + 1}:', // Add label here
                            style: TextStyle(
                              fontWeight: FontWeight.bold,
                              fontSize: 16,
                            ),
                          ),
                          SizedBox(height: 5),
                          Text(
                            snapshot.data![index].title,
                            style: TextStyle(
                              fontWeight: FontWeight.bold,
                              fontSize: 18,
                            ),
                          ),
                          SizedBox(height: 5),
                          Text(snapshot.data![index].body),
                        ],
                      ),
                    ),
                  );
                },
              );
            } else if (snapshot.hasError) {
              return Text("${snapshot.error}");
            }

            // By default, show a loading spinner.
            return CircularProgressIndicator();
          },
        ),
      ),
    );
  }
}
```

post_model.dart

```
class Post  {
  final int userId;
  final int id;
  final String title;
  final String body;
Post({
required this.userId, required this.id, required this.title, required this.body,
});

factory Post.fromJson(Map<String, dynamic> json) { return Post(
userId: json['userId'], id: json['id'],
title: json['title'], body: json['body'],
);
}
}

dev_dependencies: flutter_test:
sdk: flutter http: ^0.13.3
```

Output:



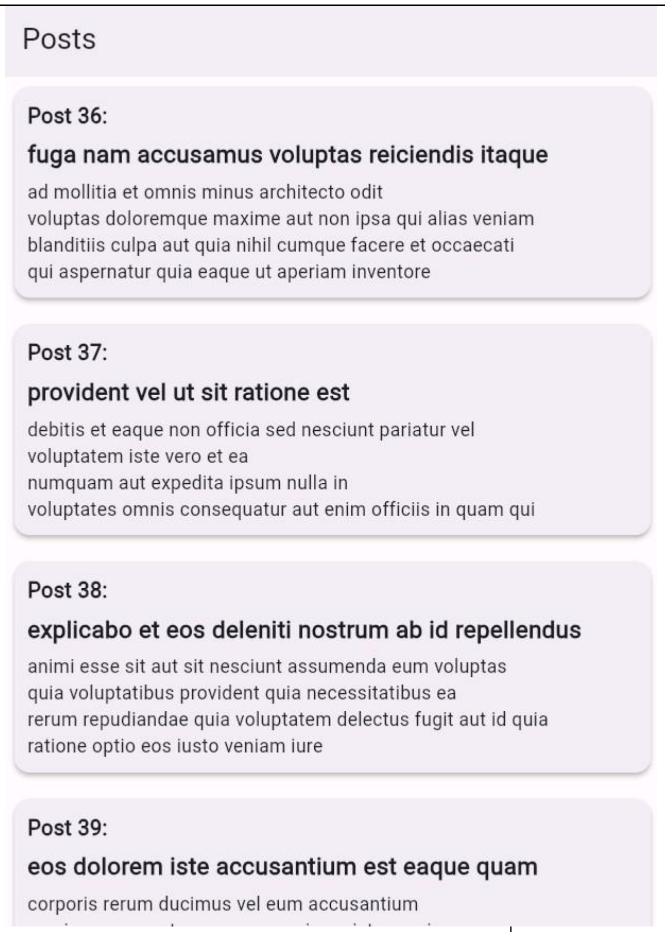
Posts

Post 10:
optio molestias id quia eum
quo et expedita modi cum officia vel magni
doloribus qui repudiandae
vero nisi sit
quos veniam quod sed accusamus veritatis error

Post 11:
et ea vero quia laudantium autem
delectus reiciendis molestiae occaecati non minima eveniet qui
voluptatibus
accusamus in eum beatae sit
vel qui neque voluptates ut commodi qui incidunt
ut animi commodi

Post 12:
in quibusdam tempore odit est dolorem
itaque id aut magnam
praesentium quia et ea odit et ea voluptas et
sapiente quia nihil amet occaecati quia id voluptatem
incidunt ea est distinctio odio

Post 13:
dolorum ut in voluptas mollitia et saepe quo animi
aut dicta possimus sint mollitia voluptas commodi quo doloremque



Posts

Post 36:
fuga nam accusamus voluptas reiciendis itaque
ad mollitia et omnis minus architecto odit
voluptas doloremque maxime aut non ipsa qui alias veniam
blanditiis culpa aut quia nihil cumque facere et occaecati
qui aspernatur quia eaque ut aperiam inventore

Post 37:
provident vel ut sit ratione est
debitis et eaque non officia sed nesciunt pariatur vel
voluptatem iste vero et ea
numquam aut expedita ipsum nulla in
voluptates omnis consequatur aut enim officiis in quam qui

Post 38:
explicabo et eos deleniti nostrum ab id repellendus
animi esse sit aut sit nesciunt assumenda eum voluptas
quia voluptatibus provident quia necessitatibus ea
rerum repudiandae quia voluptatem delectus fugit aut id quia
ratione optio eos iusto veniam iure

Post 39:
eos dolorem iste accusantium est eaque quam
corporis rerum ducimus vel eum accusantium

Practical 13 : Create and application using Hardware Interaction in Flutter

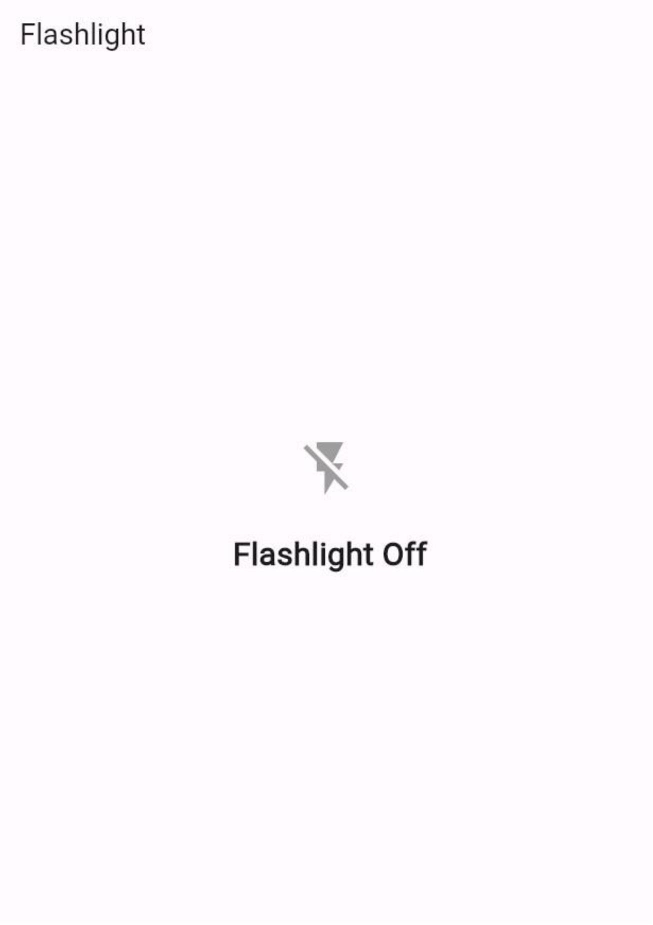
Main.dart

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

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

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
    debugShowCheckedModeBanner: false,
    title: 'Flashlight App',
      theme: ThemeData(
        primarySwatch: Colors.blue,
        visualDensity: VisualDensity.adaptivePlatformDensity,
      ),
      home: FlashlightPage(),
    );
  }
}

class FlashlightPage extends StatefulWidget {
  @override
  _FlashlightPageState createState() => _FlashlightPageState();
}

class _FlashlightPageState extends State<FlashlightPage> {
  late CameraController _controller;
  bool _flashOn = false;

  @override
  void initState() {
    super.initState();
    _initializeCamera();
  }

  void _initializeCamera() async {
    final cameras = await availableCameras();
    final camera = cameras.firstWhere(
          (camera) => camera.lensDirection == CameraLensDirection.back,
    );

    _controller = CameraController(camera, ResolutionPreset.low);
    await _controller.initialize();

    // Set initial flash mode to off
    if (_controller.value.isInitialized) {
```

```
    _controller.setFlashMode(FlashMode.off);
  }
}

@override
void dispose() {
  _controller.dispose();
  super.dispose();
}

void _toggleFlashlight() {
  if (_controller.value.isInitialized) {
    final flashMode = _controller.value.flashMode;
    if (flashMode == FlashMode.off) {
      _controller.setFlashMode(FlashMode.torch);
      setState(() {
        _flashOn = true;
      });
    } else {
      _controller.setFlashMode(FlashMode.off);
      setState(() {
        _flashOn = false;
      });
    }
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text('Flashlight'),
    ),
    body: Center(
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          IconButton(
            icon: Icon(
              _flashOn ? Icons.flash_on : Icons.flash_off,
              size: 48,
              color: _flashOn ? Colors.yellow : Colors.grey,
            ),
            onPressed: _toggleFlashlight,
          ),
          SizedBox(height: 16),
          Text(
            _flashOn ? 'Flashlight On' : 'Flashlight Off',
            style: TextStyle(
              fontSize: 24,
              fontWeight: FontWeight.bold,
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

```
dependencies:flutter:
    camera: ^0.10.5+9
```

Output:



Flashlight

Flashlight Off