Practical 1: Android Studio setup for Flutter development with along with Dart SDK.

Solution:

Step 1: Installing a Flutter.

i.   **System Requirements:**
- o Assure that your system meets the minimum requirements. Flutter supports macOS, Linux, and Windows.
- o On macOS, you need Xcode with the command-line tools installed.
- o On Linux, you need to have git, lib32stdc++6, and other dependencies installed.
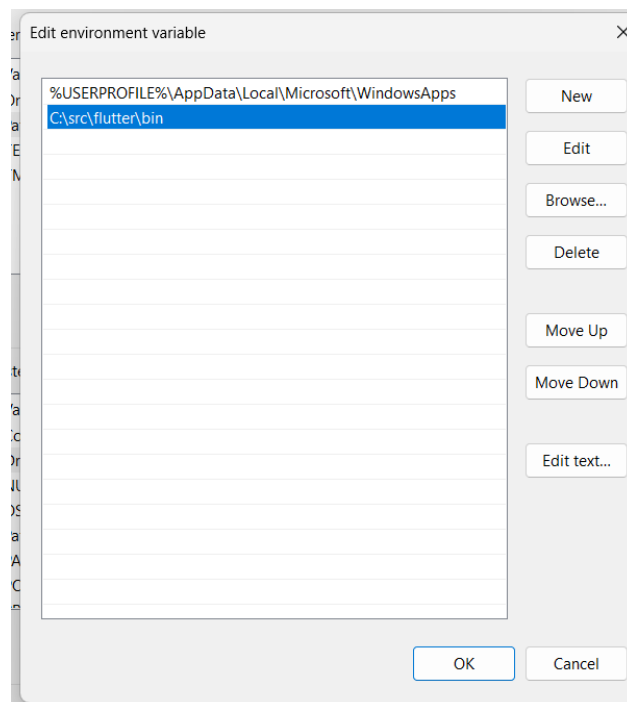
ii.   **Download Flutter:**
- o Visit Flutter Website for Installation of Flutter -> **https://docs.flutter.dev/get-started/install**.

iii.   **Extract Flutter:**
- o If you downloaded the ZIP file, extract it to a location on your machine. **(C:\src\flutter)**.

iv.   **Set Up Environment Variables:**
- o Add the **C:\src\flutter\bin** directory to your system's PATH variable.

v.   **Run flutter doctor:**
- o Open a terminal and run the following command: flutter doctor
- o This command checks your environment and displays a report of any missing dependencies or issues.

vi.   **Install Flutter Dependencies:**
- o Follow the instructions provided by flutter doctor to install any missing dependencies. This may include things like Android Studio, Xcode command-line tools, etc.
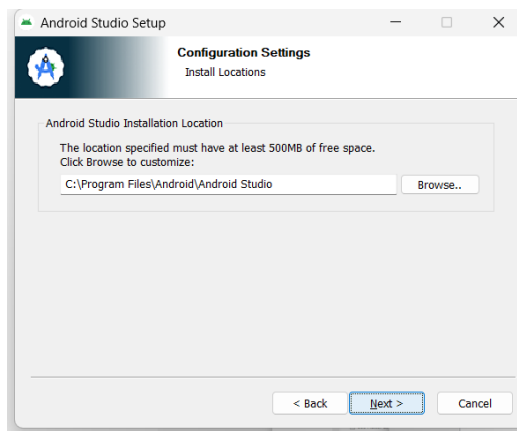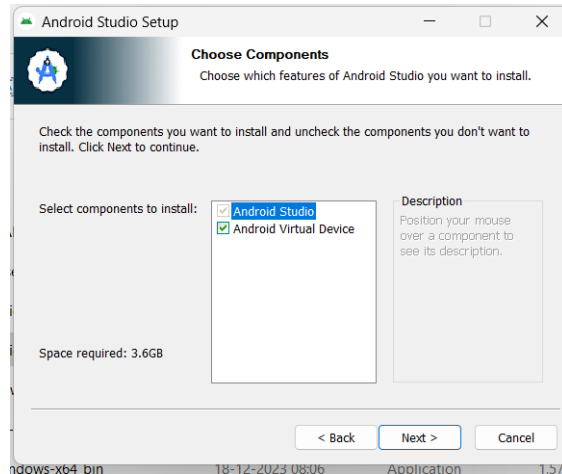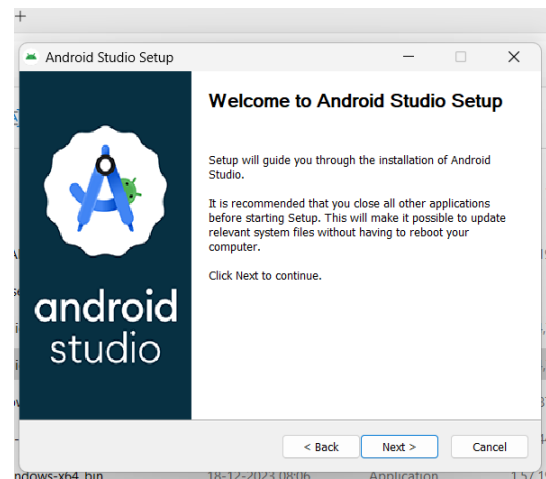
Step 2: Installing Android Studio.

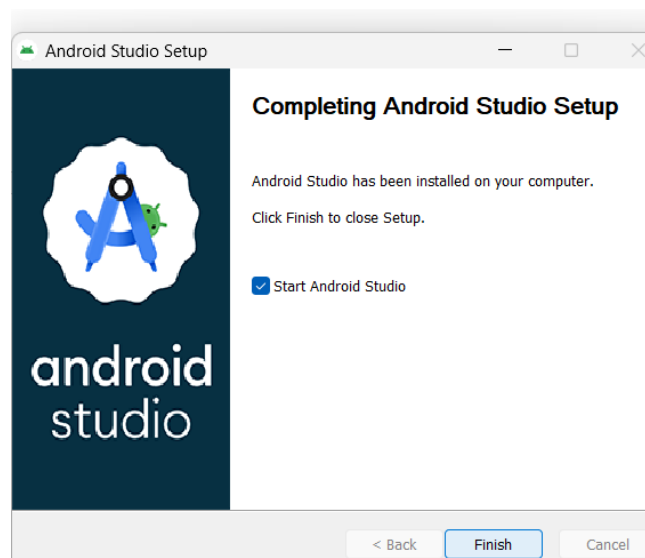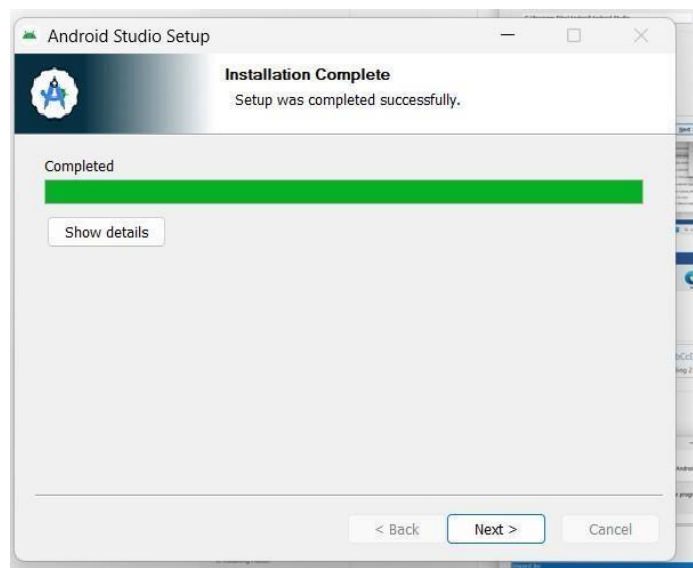**i.**   **Download Android Studio:**
- o   Visit the Android Studio download page.
- o   Click on the "Download" button and download the Windows version.

ii.   **Run the Installer:**
- o   Once the download is complete, run the installer executable (.exe) file.

iii.   **Follow Installation Wizard:**
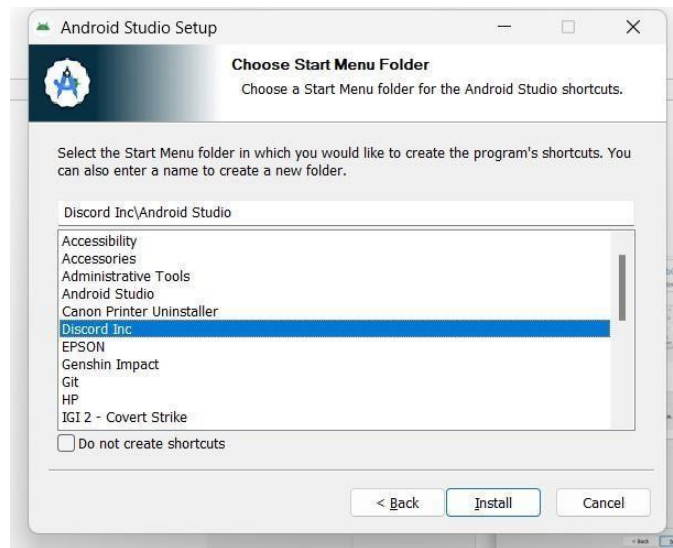
o **Android SDK Platforms**:
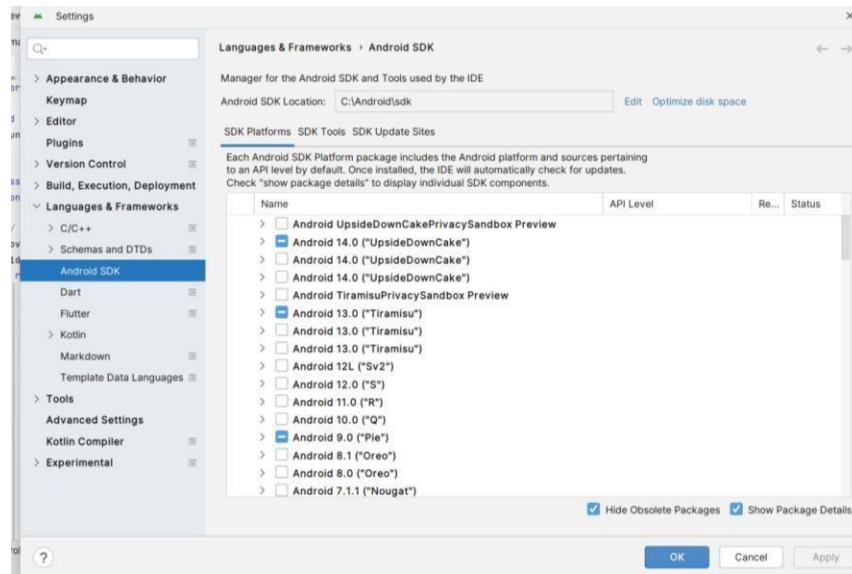
o **Android SDK Tools:**
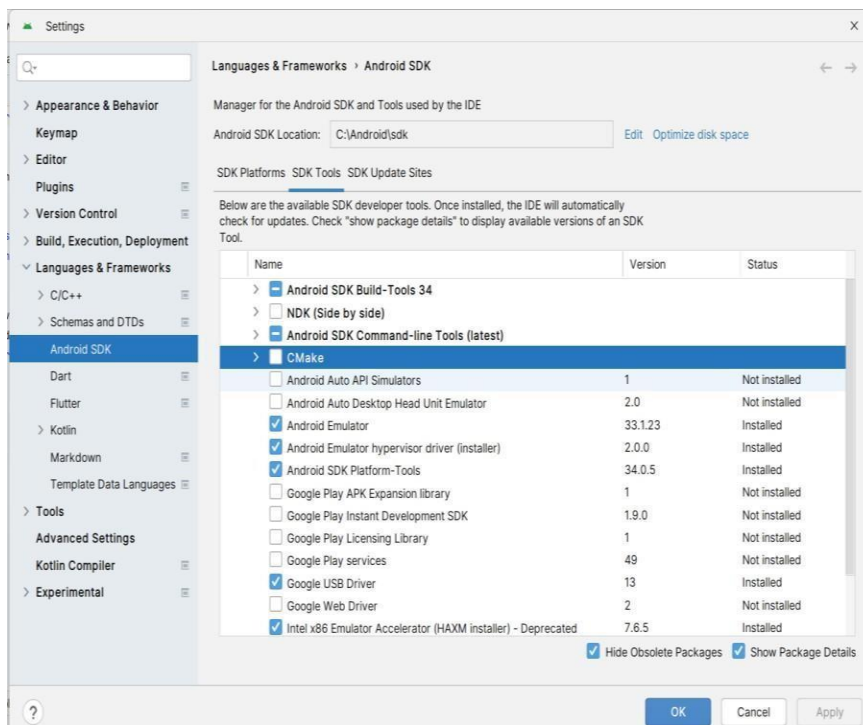
Step 3: Run Following Command for checking Flutter dependencies on after installation of android.

iv.   **Accept Android Licenses**
  o   Flutter doctor --android-licenses to develop for Android, you need to accept the Android licenses.
  o   Run the following command: **flutter doctor --android-licenses**

Practical 2: Create a "Hello Flutter" application. Main.dart:

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

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

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

 @override
 Widget build(BuildContext context) {
  return MaterialApp(
    title: 'Flutter Demo',
    theme: ThemeData(
    ),
    home: const MyHomePage(title: 'My Intro using Flutter App'),
  );
 }
}

class MyHomePage extends StatefulWidget {
 const MyHomePage({super.key, required this.title});

 final String title;

 @override
 State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {

 @override
 Widget build(BuildContext context) {

  return Scaffold(
    appBar: AppBar(

     backgroundColor: Theme.of(context).colorScheme.inversePrimary,
     title: Text(widget.title),
    ),
    body: Center(
     child: Column(
      mainAxisAlignment: MainAxisAlignment.center,
      children: <Widget>[
       //Image.asset("assets/images/deskt.png", height: 350, width: 350),
        const Text(
         'Name : - Nandan Bhimani',
         style: TextStyle(color: Colors.orange, fontSize: 18),
        ),
        const Text(
         '\nEnrollment No : - 92210103018',
         style: TextStyle(color: Colors.blue, fontSize: 18),
        ),
        const Text(
```

```
         '\n Class : 6TC3 - C',
          style: TextStyle(color: Colors.green, fontSize: 18),
        ),
        const Text(
         '\n University : - Marwadi University',
          style: TextStyle(color: Colors.red, fontSize: 18),
        ),
       ],
     ),
    ),
   );
  }
 }
```

Output:

Practical 3: Create and application using Flutter Key Widgets:

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

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

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

 @override
 Widget build(BuildContext context) {
  return MaterialApp(
   title: 'Flutter Demo',
   theme: ThemeData(
    colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
    useMaterial3: true,
   ),
   home: const MyHomePage(title: 'Flutter Buttons'),
  );
 }
}

class MyHomePage extends StatefulWidget {
 const MyHomePage({super.key, required this.title});

 final String title;

 @override
 State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   appBar: AppBar(
    backgroundColor: Theme.of(context).colorScheme.inversePrimary,
    title: Text(widget.title),
   ),
   body: Center(
    child: Column(children: <Widget>[
     Text('Types of Button in Flutter',style: TextStyle(fontSize: 25),),
     Container(
      margin: EdgeInsets.all(35),
      child: FloatingActionButton(
      child: Text(
       'Float BTN',
      ),onPressed: () {},
      ),
     ),
    Container(
     margin: EdgeInsets.all(25),
     child: OutlinedButton(
```
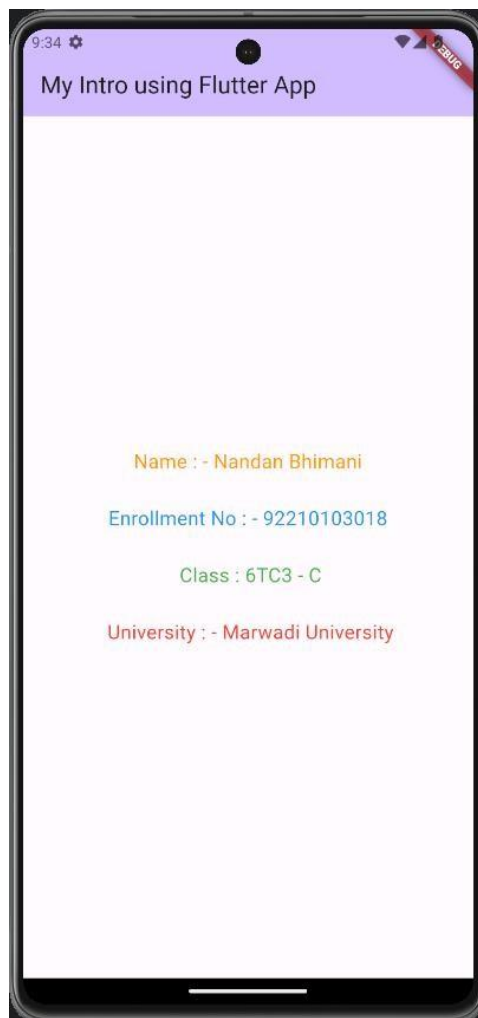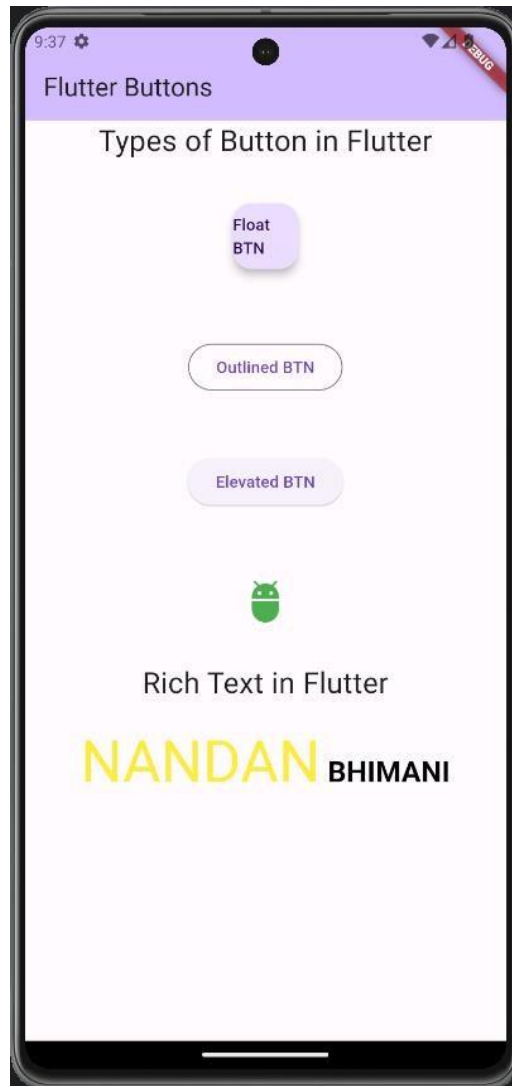
```dart
            child: Text(
             'Outlined BTN'
            ),onPressed: () {},
          ),
        ),
        Container(
          margin: EdgeInsets.all(25),
          child: ElevatedButton(
          onPressed: () {  },
            child: Text(
             'Elevated BTN'
            ),
          ),
        ),
          Container(
            margin: EdgeInsets.all(25),
            child: IconButton(onPressed: () {
            },
             icon: Icon(Icons.adb),
             iconSize: 40,
             color: Colors.green,
            ),
          ),
          Text('Rich Text in Flutter', style: TextStyle(fontSize: 25),),
          Container(
            padding: EdgeInsets.all(20),
            child: RichText(
             text: const TextSpan(
                text: "NANDAN",
                style: TextStyle(
                  color: Colors.yellow,
                  fontSize: 50,
                ),
                children: [
                 TextSpan(
                   text: ' BHIMANI',
                   style: TextStyle(
                     fontWeight: FontWeight.bold,
                     color: Colors.black,
                     fontSize: 25,
                   ),
                 )
                ]
            ),
          ),
        ),
        ),
        ],
      ),
    ),
  );
}
};
```

Output:

Practical 4: Create and application using Flutter Key Widgets:

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

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

class MyApp extends StatelessWidget {
 const MyApp({super.key});
 @override
 Widget build(BuildContext context) {
  return MaterialApp(
   title: 'Flutter P3',
   theme: ThemeData(

    colorScheme: ColorScheme.fromSeed(seedColor: Colors.deepPurple),
    useMaterial3: true,
   ),
   home: const MyHomePage(title: 'Flutter Row and Column'),
  );
 }
}

class MyHomePage extends StatefulWidget {
 const MyHomePage({super.key, required this.title});

 final String title;

 @override
 State<MyHomePage> createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
 @override
 Widget build(BuildContext context) {
  return Scaffold(
   appBar: AppBar(
    backgroundColor: Theme.of(context).colorScheme.inversePrimary,
    title: Text(widget.title),
   ),
   body: Column(
    children: <Widget>[
     Text('Row',style: TextStyle(fontSize: 25),),
     Center(
      child: Row(
       mainAxisAlignment: MainAxisAlignment.spaceEvenly,
       children: <Widget>[
        Container(
         margin: EdgeInsets.all(12.0),
         padding: EdgeInsets.all(8.0),
         decoration:BoxDecoration(
          borderRadius:BorderRadius.circular(8),
          color:Colors.lightBlue
         ),
         child: Text(
```

```dart
            'Flutter',
            style: TextStyle(
                color: Colors.red,
                fontSize: 20
            ),
          ),
        ),
        Container(
          margin: EdgeInsets.all(15.0),
          padding:  EdgeInsets.all(8.0),
          decoration:BoxDecoration(
              borderRadius:BorderRadius.circular(8),
              color:Colors.green
          ),
          child: Text("Android",style: TextStyle(color:Colors.orange,fontSize:25),),
        ),
        Container(
          margin: EdgeInsets.all(12.0),
          padding:  EdgeInsets.all(8.0),
          decoration:BoxDecoration(
              borderRadius:BorderRadius.circular(8),
              color:Colors.yellow
          ),
          child: Text("Firebase",style: TextStyle(color:Colors.blue,fontSize:25),),
        )
      ],
    ),
  ),
),
Column(
  children: <Widget>[
    Text('Column',style: TextStyle(fontSize: 25),),
    Container(
      margin: EdgeInsets.all(20.0),
      padding: EdgeInsets.all(12.0),
      decoration: BoxDecoration(
        borderRadius: BorderRadius.circular(8),
        color: Colors.red,
      ),
      child: Text(
        "React.js",
        style: TextStyle(color: Colors.yellowAccent, fontSize: 20),
      ),
    ),
    Container(
      margin: EdgeInsets.all(20.0),
      padding: EdgeInsets.all(12.0),
      decoration: BoxDecoration(
        borderRadius: BorderRadius.circular(8),
        color: Colors.red,
      ),
      child: Text(
        "Flutter",
        style: TextStyle(color: Colors.yellowAccent, fontSize: 20),
      ),
    ),
    Container(
```
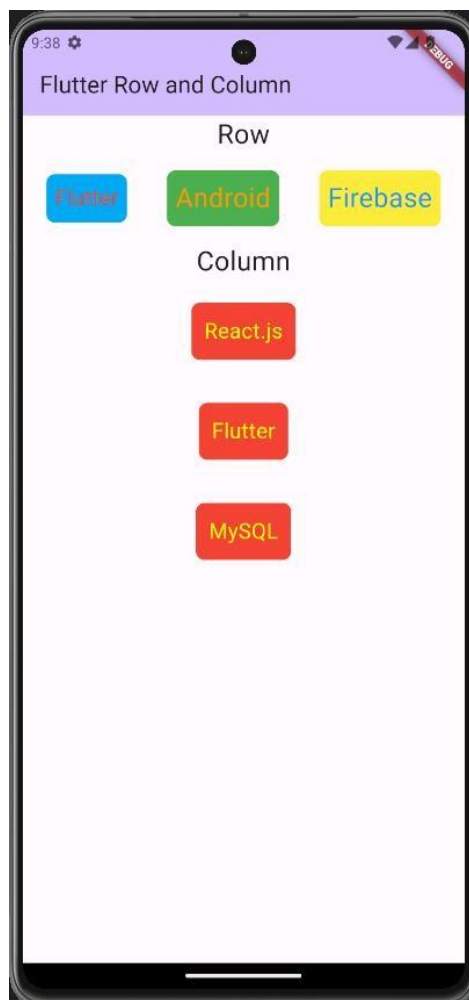
```
            margin: EdgeInsets.all(20.0),
            padding: EdgeInsets.all(12.0),
            decoration: BoxDecoration(
              borderRadius: BorderRadius.circular(8),
              color: Colors.red,
            ),
            child: Text(
              "MySQL",
              style: TextStyle(color: Colors.yellowAccent, fontSize: 20),
            ),
          ),
        ],
      ),
    ],
  ),
);
}
}
```

Output:

Practical 5: Create and application with Flutter UI Components.

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

void main() {
 runApp(MaterialApp(
  debugShowCheckedModeBanner: false,
  home:Scaffold(
   appBar: AppBar(
    title: const Text("Practical - 5"),
    backgroundColor: Colors.yellowAccent,
    foregroundColor: Colors.black87,
   ),

   body: const LoginPage(),
  ),
 ),
 );
}
```

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

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

 @override
 Widget build(BuildContext context) {
  return SingleChildScrollView(
  child: Column(
    children: [
     const SizedBox(height: 75,),
     Container(
      height: 200,
      width: 250,
      child: Image.asset("assets/images/login.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),
      ),
      child: const TextField(
       decoration: InputDecoration(
        hintText: "Username and Email",
       ),
       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),
      ),
        child: const TextField(
          obscureText: true,
          decoration: InputDecoration(
          hintText: "Password",
          ),
        ),
      ),
      const SizedBox(height: 30,),
      Container(
        height: 50,
        width: 350,
        child: OutlinedButton(
          onPressed: () {
          print('Button Passed');
          },
          style: ButtonStyle(
            foregroundColor: MaterialStateProperty.all(Colors.black87),
            backgroundColor: MaterialStateProperty.all(Colors.red),
          ),
          child: const Text("Login",
          style: TextStyle(fontSize: 20),),
        ),
      )
    ],
    ),
  );
 }
}
```

Output:
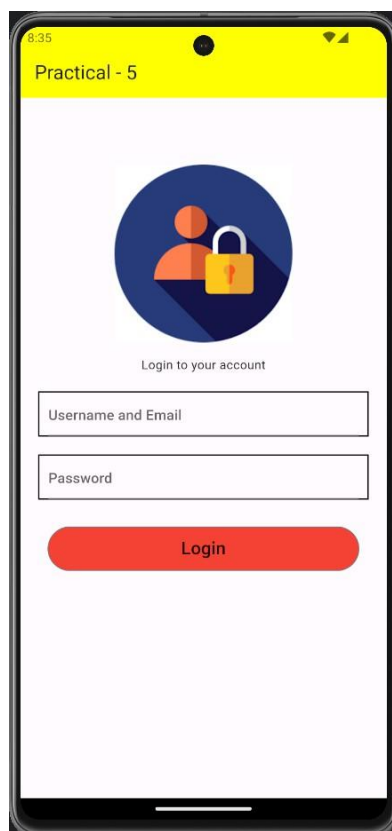
Practical 6: Create and application with Flutter UI Components.

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

void main() {
 runApp(MaterialApp(
   debugShowCheckedModeBanner: false,
   home:Scaffold(
     appBar: AppBar(
       title: const Text("Practical - 6"),
       backgroundColor: Colors.red,
       foregroundColor: Colors.black87,
     ),

     body: const SignUpPage(),
   ),
 ),
 );
}
```

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

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

 @override
 Widget build(BuildContext context) {
  return SingleChildScrollView(
  child: Column(
    children: [
      const SizedBox(height: 75,),
      Container(
        height: 150,
        width: 150,
        child: Image.asset("assets/images/signup.jpeg"),
      ),
      const SizedBox(height: 15,),
      const Text("Register New 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),),
        child: const TextField(
          decoration: InputDecoration(
            hintText: "Name",),
          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),
  ),
  child: const TextField(
    decoration: InputDecoration(
    hintText: "Email Address",
    ),
    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),
  ),
  child: TextField(
    keyboardType: TextInputType.number,
    inputFormatters: [
    FilteringTextInputFormatter.digitsOnly,
    ],
    decoration: InputDecoration(
      hintText: "Mobile Number",
    ),
    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),
  ),
  child: const TextField(
    obscureText: true,
    decoration: InputDecoration(
    hintText: "Password",
    ),
    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),
  ),
```

```
    child: const TextField(
      obscureText: true,
      decoration: InputDecoration(
        hintText: "Confirm Password",
      ),
      spellCheckConfiguration: SpellCheckConfiguration(
        misspelledSelectionColor: Colors.red),
    ),
  ),
  const SizedBox(height: 30,),
  Container(
    height: 50,
    width: 350,
    child: OutlinedButton(
      onPressed: () {
      print('Button Passed');
      },
      style: ButtonStyle(
        foregroundColor: MaterialStateProperty.all(Colors.black87),
      backgroundColor: MaterialStateProperty.all(Colors.yellow), ),
      child: const Text("Register",
        style: TextStyle(fontSize: 20),),),),
  )
  ],
  ),
);
}
}
```

Output:
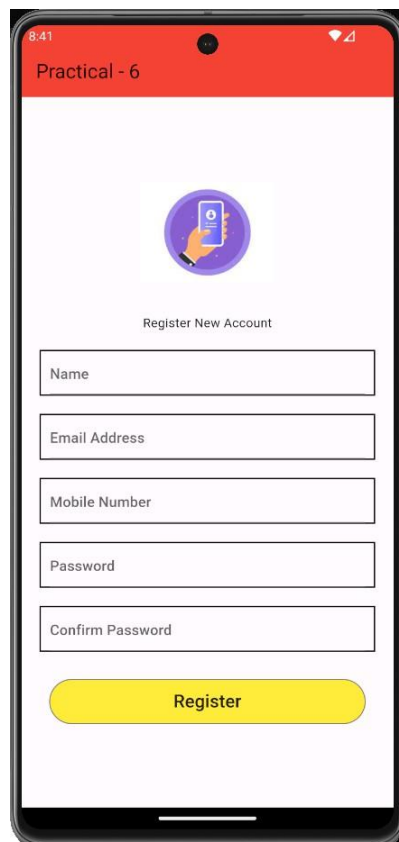
Practical 7: Create and application with Navigation in Flutter.

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

void main() {
 runApp(MaterialApp(
  debugShowCheckedModeBanner: false,
  home:Scaffold(
   appBar: AppBar(
    title: const Text("Practical - 7"),
    foregroundColor: Colors.black87,
   ),

   body:  LoginPage(),
  ),
 ),
 );
}
```

login.dart:
```dart
import 'package:flutter/material.dart';
import 'package:practical7/custom_geasture.dart';
import 'package:practical7/forpass.dart';
import 'package:practical7/signup.dart';

class LoginPage extends StatelessWidget {

LoginPage({super.key});

 final TextEditingController usernameController = TextEditingController();
 final TextEditingController passwordController = TextEditingController();

 @override
 Widget build(BuildContext context) {
  return MaterialApp(
  debugShowCheckedModeBanner: false,
  home: Scaffold(
    body: Container(
     margin: const EdgeInsets.all(24),
     child: Column(
      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
      children: [
       _header(context),
       _inputField(context),
       _forgotPassword(context),
       _signup(context),
      ],
     ),
    ),
   ),
  );
 }
```

```dart
_header(context) {
  return const Column(
  children: [
    Text(
      "Welcome Back",
      style: TextStyle(fontSize: 40, fontWeight: FontWeight.bold),
    ),
    Text("Enter your credential to login"),
  ],
 );
}

_inputField(context) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.stretch,
    children: [
    TextFormField(
      controller: usernameController,
      decoration: InputDecoration(
        hintText: "Username",
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(18),
          borderSide: BorderSide.none
        ),
        fillColor: Colors.purple.withOpacity(0.1),
        filled: true,
        prefixIcon: const Icon(Icons.person)),
      validator: (value) {
       if (value!.isEmpty) {
        return 'Please enter your username';
        }
        return null;
       },
     ),
     const SizedBox(height: 10),
     TextFormField(
      controller: passwordController,
      decoration: InputDecoration(
      hintText: "Password",
        border: OutlineInputBorder(
          borderRadius: BorderRadius.circular(18),
          borderSide: BorderSide.none),
        fillColor: Colors.purple.withOpacity(0.1),
        filled: true,
        prefixIcon: const Icon(Icons.password),
      ),
      obscureText: true,
      validator: (value) {
       if (value!.isEmpty) {
        return 'Please enter your password';
        }
        return null;
       },
     ),
     const SizedBox(height: 10),
     ElevatedButton(
```

```dart
      onPressed: () {
        if (usernameController.text == 'admin' &&
          passwordController.text == 'admin123!@#') {
          Navigator.push(context, MaterialPageRoute(builder: (context)=> const custom_geasture()));
          print('Login Successful');
        } else {
          print('Invalid Credentials');
        }
      },
      style: ElevatedButton.styleFrom(
        shape: const StadiumBorder(),
        padding: const EdgeInsets.symmetric(vertical: 16),
        backgroundColor: Colors.purple,
      ),
      child: const Text(
        "Login",
        style: TextStyle(fontSize: 20, color: Colors.white),
      ),
    )
    ],
  );
}

_forgotPassword(context) {
  return TextButton(
  onPressed: () {
    Navigator.pushReplacement(context, MaterialPageRoute(builder: (context)=> const ForPassPage()));
    },
    child: const Text("Forgot password?",
      style: TextStyle(color: Colors.purple),
    ),
  );
}

_signup(context) {
  return Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      const Text("Don't have an account? "),
      TextButton(
        onPressed: () {
          Navigator.push(context, MaterialPageRoute(builder: (context)=> SignupPage()));
        },
        child: const Text("Sign Up", style: TextStyle(color: Colors.purple),)
      )
    ],
  );
}
}
```

signup.dart:

```dart
import  'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:practical7/login.dart';

import 'custom_geasture.dart';

class SignupPage extends StatelessWidget {
  SignupPage({super.key});

  final TextEditingController emailController = TextEditingController();
  final TextEditingController usernameController = TextEditingController();
  final TextEditingController passwordController = TextEditingController();
  final TextEditingController conpasswordController = TextEditingController();
  final TextEditingController mobileController = TextEditingController();

  @override
  Widget build(BuildContext context) {

    return MaterialApp(
      debugShowCheckedModeBanner: false,
      home: Scaffold(
        body: SingleChildScrollView(
          child: Container(
            padding: const EdgeInsets.symmetric(horizontal: 40),
            height: MediaQuery.of(context).size.height - 50,
            width: double.infinity,
            child: Column(
              mainAxisAlignment: MainAxisAlignment.spaceEvenly,
              crossAxisAlignment: CrossAxisAlignment.stretch,
              children: <Widget>[
                Column(
                  children: <Widget>[
                    const SizedBox(height: 60.0),
                    const Text(
                      "Sign up",
                      style: TextStyle(
                        fontSize: 30,
                        fontWeight: FontWeight.bold,
                      ),
                    ),
                    const SizedBox(height: 20,),
                    Text(
                      "Create your account",
                      style: TextStyle(fontSize: 15, color: Colors.grey[700]),
                    )
                  ],
                ),
                Column(
                  children: <Widget>[
                    TextFormField(
```

```
          decoration: InputDecoration(
            hintText: "Username",
            border: OutlineInputBorder(
               borderRadius: BorderRadius.circular(18),
               borderSide: BorderSide.none),
            fillColor: Colors.purple.withOpacity(0.1),
            filled: true,
            prefixIcon: const Icon(Icons.person)),
          validator: (value) {
           if (value!.isEmpty) {
            return 'Please enter your username';
           }
           return null;
          },
        ),
        const SizedBox(height: 20),
        TextFormField(
         controller: emailController,
         decoration: InputDecoration(
            hintText: "Email",
            border: OutlineInputBorder(
               borderRadius: BorderRadius.circular(18),
               borderSide: BorderSide.none),
            fillColor: Colors.purple.withOpacity(0.1),
            filled: true,
            prefixIcon: const Icon(Icons.email)),
          validator: (value) {
           if (value!.isEmpty) {
            return 'Please enter your email address';
           }
           return null;
          },
        ),
        const SizedBox(height: 20),
        TextFormField(
         controller: mobileController,
         decoration: InputDecoration(
            hintText: "Mobile",
            border: OutlineInputBorder(
               borderRadius: BorderRadius.circular(18),
               borderSide: BorderSide.none),
            fillColor: Colors.purple.withOpacity(0.1),
            filled: true,
            prefixIcon: const Icon(Icons.person)),
         keyboardType: TextInputType.number,
         inputFormatters: <TextInputFormatter>[
         FilteringTextInputFormatter.digitsOnly
         ],
          validator: (value) {
           if (value!.isEmpty) {
            return 'Please enter your Mobile Number';
```

```
              }
              return null;
            },
          ),
          const SizedBox(height: 20),
          TextFormField(
            controller: passwordController,
            decoration: InputDecoration(
            hintText: "Password",
              border: OutlineInputBorder(
                  borderRadius: BorderRadius.circular(18),
                  borderSide: BorderSide.none),
              fillColor: Colors.purple.withOpacity(0.1),
              filled: true,
              prefixIcon: const Icon(Icons.password),
            ),
            obscureText: true,
            validator: (value) {
             if (value!.isEmpty) {
               return 'Please enter your password';
             }
             return null;
            },
          ),
          const SizedBox(height: 20),
          TextFormField(
            controller: conpasswordController,
            decoration: InputDecoration(
            hintText: "Confirm Password",
            border: OutlineInputBorder(
                  borderRadius: BorderRadius.circular(18),
                  borderSide: BorderSide.none),
              fillColor: Colors.purple.withOpacity(0.1),
              filled: true,
              prefixIcon: const Icon(Icons.password),
            ),
            obscureText: true,
            validator: (value) {
             if(value != passwordController.text)
             return "Password Doesn't Match";
             if (value!.isEmpty) {
               return 'Please enter your password';
             }
             return null;
            },
          ),
        ],
      ),
      Container(
          padding: const EdgeInsets.only(top: 3, left: 3),
          child: ElevatedButton(
```

```
              onPressed: () {
                if (emailController.text == 'admin@gmail.com' &&
                  passwordController.text == 'admin123!@#') {
                Navigator.pushReplacement(context, MaterialPageRoute(builder: (context)=> const
custom_geasture()));
                  print('Login Successful');
                } else {
                  print('Invalid Credentials');
                }
              },
              child: const Text(
                "Sign up",
                style: TextStyle(fontSize: 20, color: Colors.white),
              ),
              style: ElevatedButton.styleFrom(
                shape: const StadiumBorder(),
                padding: const EdgeInsets.symmetric(vertical: 16),
                backgroundColor: Colors.purple,
              ),
            )
          ),
          Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: <Widget>[
              const Text("Already have an account?"),
              TextButton(
                onPressed: () {
                  Navigator.push(context, MaterialPageRoute(builder: (context)=> LoginPage()));
                },
                child: const Text("Login", style: TextStyle(color: Colors.purple),)
              )
            ],
          )
        ],
      ),
     ),
    ),
   );
  }
}
```

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

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

  @override
  Widget build(BuildContext context) {
```

```dart
return MaterialApp(
  debugShowCheckedModeBanner: false,
  home: Scaffold(
    body: Container(
      margin: const EdgeInsets.all(24),
      child: Column(
        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
        children: [
          _header(context),
          _inputField(context),
        ],
      ),
    ),
  ),
);
}

_header(context) {
  return const Column(
  children: [
    Text(
      "Reset Your Password",
      style: TextStyle(fontSize: 40, fontWeight: FontWeight.bold),
    ),
  ],
  );
}

_inputField(context) {
  return Column(
    crossAxisAlignment: CrossAxisAlignment.stretch,
    children: [
      TextField(
        decoration:   InputDecoration(
          hintText: "Email  Address",
          border: OutlineInputBorder(
            borderRadius: BorderRadius.circular(18),
            borderSide: BorderSide.none
          ),
          fillColor: Colors.purple.withOpacity(0.1),
          filled: true,
          prefixIcon: const Icon(Icons.person)),
      ),
      const SizedBox(height: 10),
      ElevatedButton(
        onPressed: () {
        Navigator.push(context, MaterialPageRoute(builder: (context)=> LoginPage()));
        },
        style: ElevatedButton.styleFrom(
          shape: const StadiumBorder(),
          padding: const EdgeInsets.symmetric(vertical: 16),
```

```
      backgroundColor: Colors.purple,
    ),
    child: const Text(
      "Reset Password",
      style: TextStyle(fontSize: 20, color: Colors.white),
    ),
  )
 ],
);
}
}
```

custom_geasture.dart:

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

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

  @override
  State<custom_geasture> createState() => _custom_geastureState();
}

class _custom_geastureState extends State<custom_geasture> {

  Color color1=Colors.orange;
  String displayText = 'Orange';
  IconData icn = Icons.temple_hindu_outlined;

  @override
  Widget build(BuildContext context) {

    return Scaffold(
      appBar: AppBar(
      centerTitle: true,
        title: Text("Custom Geasture"),
      ),
      body: GestureDetector(
        onTap: () {
          setState(() {
            if(color1==Colors.orange) {
            color1 = Colors.blue;
            displayText = 'Blue';
             icn = Icons.radar;
            }else if(color1==Colors.blue){
             color1=Colors.green;
             displayText = 'Green';
             icn = Icons.add_business;}
            else{
            color1=Colors.orange;
            displayText = 'Orange';
             icn = Icons.temple_hindu_rounded;
            }
```
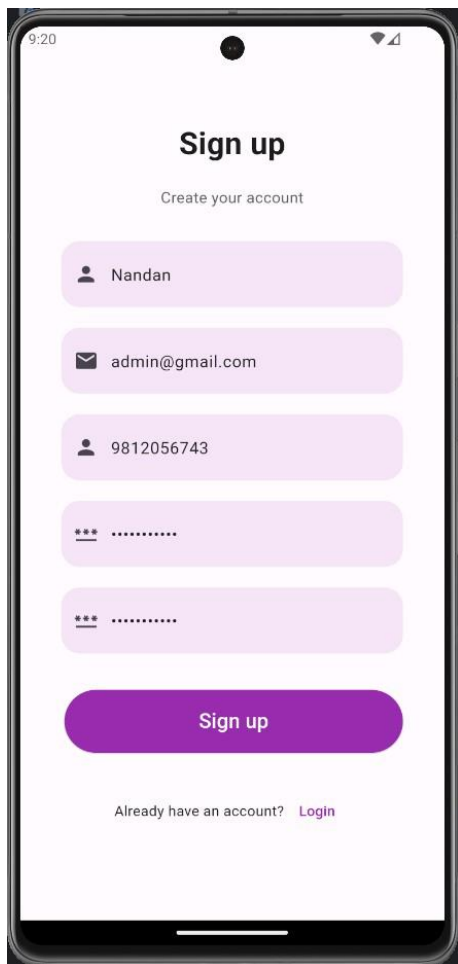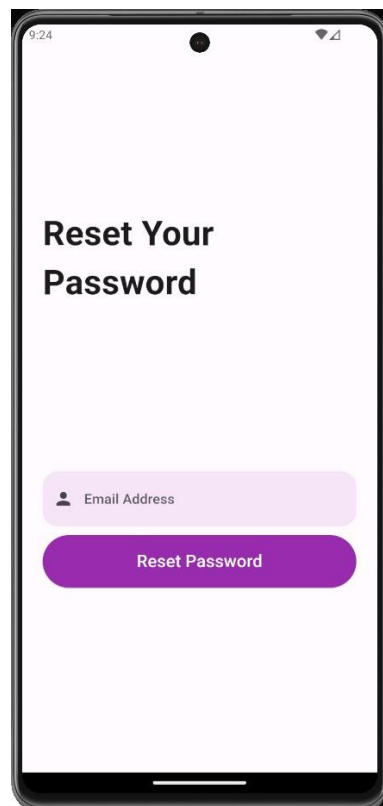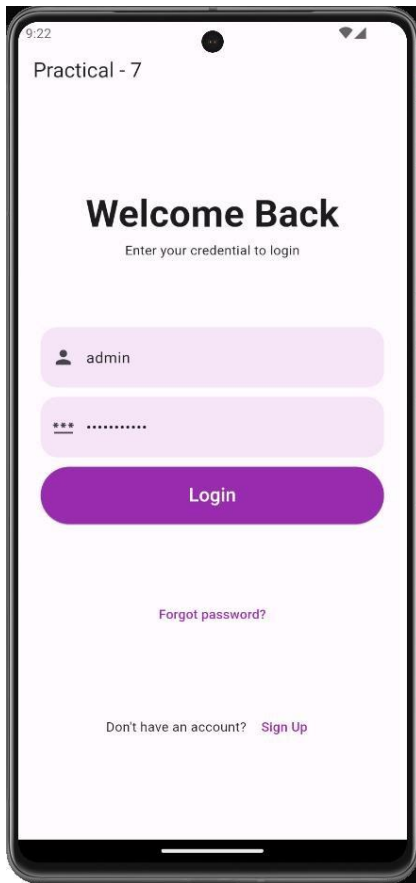
```
        });
      },
    child: Center(
      child: Container(
      height: 1000,
       width: 1000,
       color: color1,
       child: Center(
       child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: [
           Icon(
           icn,
           size: 50,
           color: Colors.white,),
           Text(displayText,
           style: TextStyle(
           fontSize: 50,
             color: Colors.white,
           ),),
           Container(

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

Output:

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

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

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

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

 // This widget is the root of your application.
 @override
 Widget build(BuildContext context) {
  return MaterialApp(
  debugShowCheckedModeBanner: false,
  home: Scaffold(
    appBar: AppBar(
     title: Text('List View', style: TextStyle(color: Colors.cyan),),
    ),
    body: Center(
     child: Container(
     child: ListView(
       padding: const EdgeInsets.all(8),
       children: <Widget>[
        ListTile(
         leading: Icon(Icons.android),
         title: Text('Android'),
         subtitle: Text("App Development", style: TextStyle(color: Colors.green),),
        ),
        ListTile(
         leading: Icon(Icons.javascript),
         title: Text('JavaScript'),
         subtitle: Text("Web Development", style: TextStyle(color: Colors.yellow),),
        ),ListTile(
         leading: Icon(Icons.css),
         title: Text('CSS'),
         subtitle: Text("Web Development", style: TextStyle(color: Colors.red),),
        ),ListTile(
         leading: Icon(Icons.html),
         title: Text('HTML'),
         subtitle: Text("Web Development", style: TextStyle(color: Colors.blue),),
        ),
        ListTile(
         leading: Icon(Icons.apple),
         title: Text('Swift'),
         subtitle: Text("IOS Development", style: TextStyle(color: Colors.grey),),
        ),
       ],
     ),
    ),
   ),
  ),
 );
 }
}
```
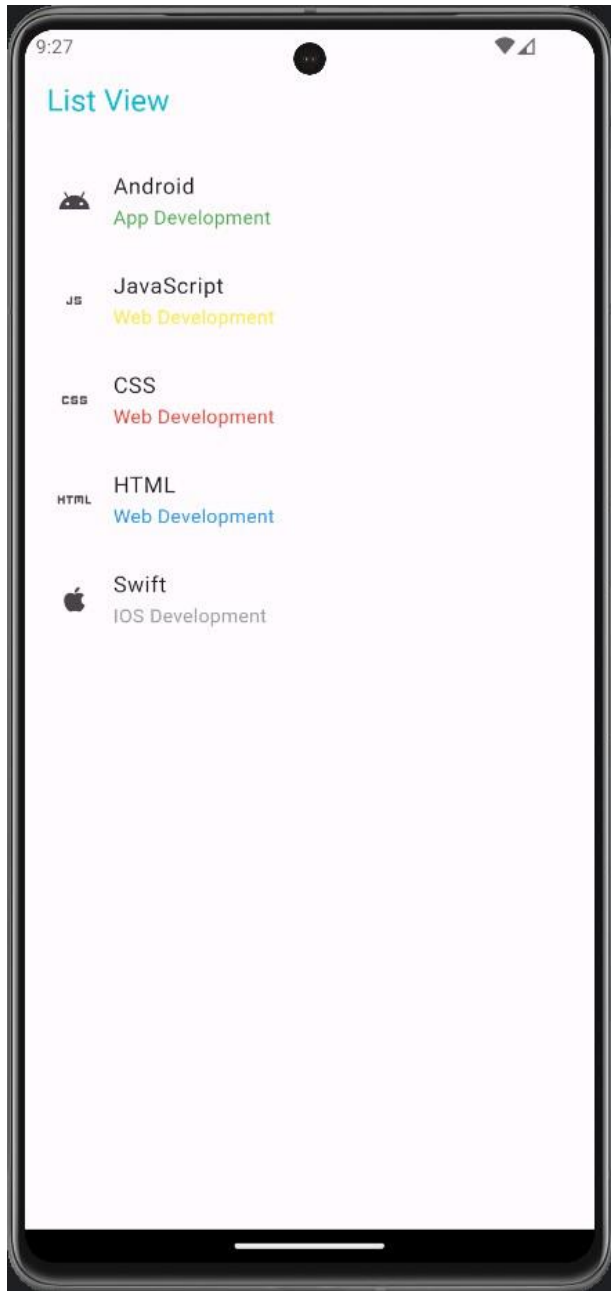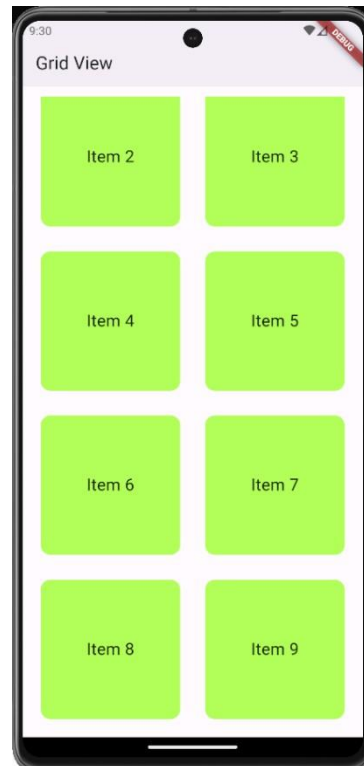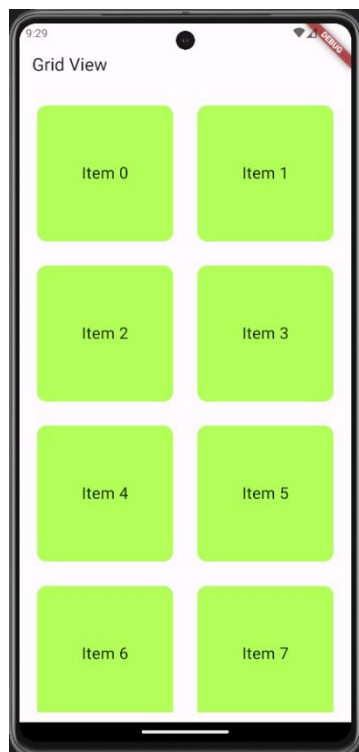
Output:

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

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

void main() {
 runApp(MyApp());}
class MyApp extends StatelessWidget {
 @override
 Widget build(BuildContext context) {
  return MaterialApp(
   home: Scaffold(
    appBar: AppBar(
     title: Text("Grid View"),
    ),
    body: Container(
     padding: EdgeInsets.all(12.0),
     child: GridView.count(
     crossAxisCount: 2,
      crossAxisSpacing: 10.0,
      mainAxisSpacing: 10.0,
      shrinkWrap: true,
      children: List.generate(10, (index) {
       return Padding(
        padding: const EdgeInsets.all(10.0),
        child: Container(
         alignment: Alignment.center,
         decoration: BoxDecoration(
         color: Colors.lightGreenAccent,
          borderRadius: BorderRadius.circular(12.0),
         ),
         child: Text('Item $index',
          style: TextStyle(fontSize: 20, color: Colors.black87),),),),
      );
     }),
    ), ), ),
   );
  }
 }
```

Output:

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

Main.dart

```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
```

```dart
                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(),
```

```dart
            )
          : 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

```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
    )
```

```dart
    """);
  }
// 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");
   }
  }
}
```
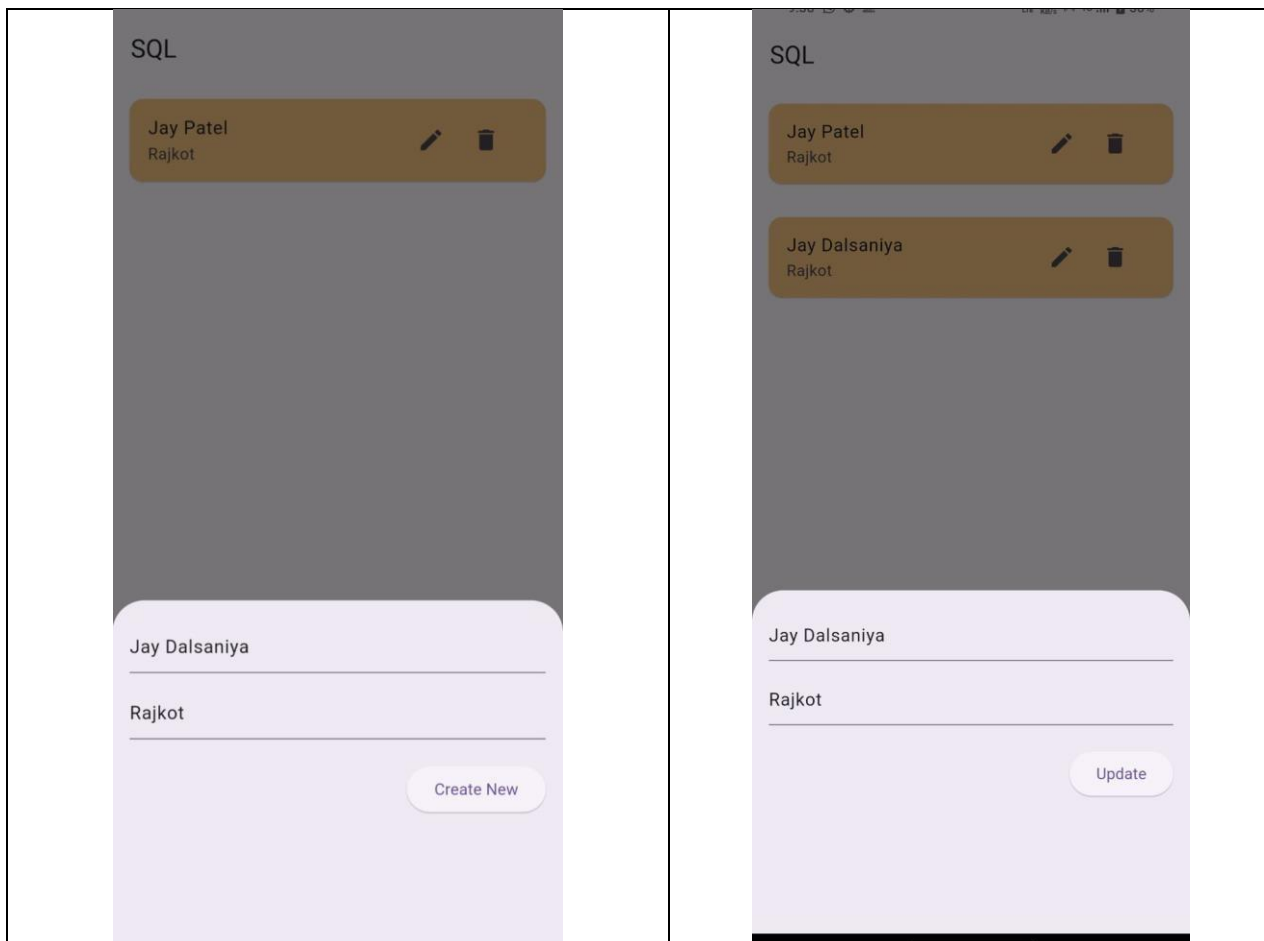
```yaml
dependencies:
 flutter:
   sdk: flutter
 sqflite: ^2.0.0
 path: ^1.9.0
 path_provider: any
```
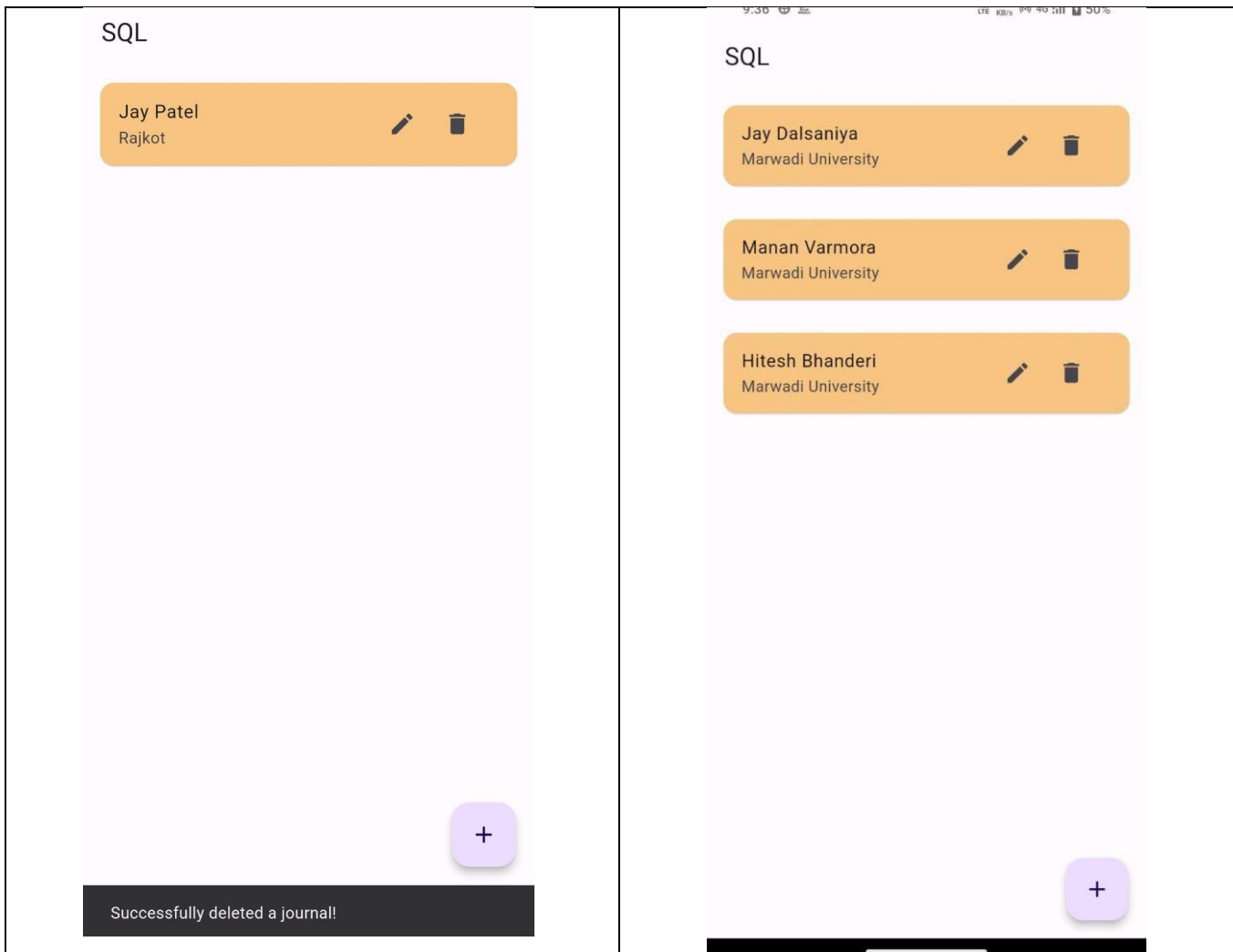
Output:

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

```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

```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'],
    );
  }
```

```dart
}

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

```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(
```

```dart
              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();
    },
  ),
  ),
  );
  }
}
```

```yaml
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) {
```

```dart
      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

```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:

Practical 13 : Create and application using Hardware Interaction in Flutter

Main.dart

```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+9Output:

Flashlight

Flashlight Off