Flutter- DEMO-LAB-6

Demo.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.blue),
        useMaterial3: true,
 ),
      home: const MyHomePage(title: 'I Love My College:CMRIT'),
    );
  }
}
class MyHomePage extends StatefulWidget
{
  const MyHomePage({super.key, required this.title});
  final String title;
  @override
  State<MyHomePage> createState() => _MyHomePageState();
}
class _MyHomePageState extends State<MyHomePage>
{
  int _counter = 0;
  void _incrementCounter()
{
    setState(()
{
      _counter--;
   });
  }
  @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.end,
          children: <Widget>
          [
            const Text(
              'Please pushed the button:',
            ),
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headlineMedium,
            ),
          ],
        ),
      ),
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: 'Increment',
        child: const Icon(Icons.add),
      ),
    );
  }
}
```

LAB-7-Intellij-Example

Example.dart

```
import 'dart:io';
import 'dart:math' as math;
void main()
{
  print("pizza price ,small: 150 Rs, Medium: 190 Rs, Large: 240 Rs");
  print("please enter your pizza size (Small, Medium, Large):");
  String pizzaSize=stdin.readLineSync()!;
  print("How Many pizza do you want to $pizzaSize");
  int pizzaQty=int.parse(stdin.readLineSync()!);
  //condition
  if(pizzaSize=='Small')
    {
      var result=pizzaQty*150;
      print("Your Total Payment is :$result");
    }
  else if(pizzaSize=='Medium')
    {
      var result=pizzaQty*190;
      print("Your Total Payment is:$result");
    }
  else if(pizzaSize=='Large')
    {
      var result=pizzaQty*240;
      print("Your Total Payment is:$result");
    }
  else
    {
      print("Invalid Pizza Size Input");

    }
}
```

LAB-8

main.dart

```
import 'package:flutter/material.dart';
import 'package:lab8/Dashboard.dart';

void main() async{
  runApp(MyApp());
}
class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
          return MaterialApp(
            title: 'pizza dashboard',
            theme: ThemeData(
              primarySwatch: Colors.deepPurple,
            ),
            debugShowCheckedModeBanner: false,
            home: Dashboard(),
          );
        // });
  }
}
```

Dashboard.dart

```
import 'package:flutter/material.dart';
import 'package:lab8/top_navigation_bar.dart';

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

  @override
  Widget build(BuildContext context) {
    // Material App
    return MaterialApp(
        home:Scaffold(
      appBar: AppBar(
        title: Text("Welcome to WOW PIZZA!"),
        centerTitle: true,
        backgroundColor: Colors.deepOrange,
      ),
      body: Column(children:[
        Image.asset('Images/meal.jpg',
          width:400.0,
          height:300.0,),
        Text("Wanna Order Something... ",
          style: TextStyle(decoration:TextDecoration.none,fontSize:30.0),
          textAlign: TextAlign.center,
        ),
        ElevatedButton(
          onPressed: () async => {
            Navigator.pushAndRemoveUntil(
                context,
                MaterialPageRoute(
                  builder: (context) => TopNavigationBar(),
                ),
                    (route) => false)
          },
          child: Text('Lets Go'),
          style: ElevatedButton.styleFrom(primary: Colors.amber),
        )
      ]
      )
    ));
  }
}
```

Navigationbar.dart

```
import 'package:flutter/material.dart';
class TopNavigationBar extends StatelessWidget{
  @override
  Widget build(BuildContext context){
    return DefaultTabController(length: 3,
        child: Scaffold(
          appBar: AppBar(
            backgroundColor: Colors.redAccent,
            title: Row(children: [
              Text('WOW Pizza',
                style:TextStyle(fontSize: 20.0),),
              SizedBox(width:160.0,),
              Container(child:GestureDetector(
                child:Image.asset(
                  'Images/twitter.png',
     fit:BoxFit.contain,
     height: 30,),
   ),),
   SizedBox(width:10.0,),
   Container(child:GestureDetector(
    child:Image.asset(
     'Images/facebook.png',
     fit:BoxFit.contain,
     height: 40,),
   ),),
  ],
  ),
  bottom: TabBar(indicatorColor: Colors.red,
    tabs:[
     Tab(icon: Container(height:40,
       child: Image.asset('lib/icons/pizza.png')),
       text:'Vegitable Pizza'
     ),
     Tab(icon: Container(height: 40,
       child: Image.asset('lib/icons/pizza.png')),
       text:'Cheese Pizza'
     ),
     Tab(icon: Container(height: 40,
       child: Image.asset('lib/icons/french-fries.png')),
       text:'Fries'
     ),
    ]
  )
),
body:TabBarView(
 children: [
  MaterialApp(home:Center(child:Column(children:[
   Image.asset('Images/Vpizza.png',
    width:400.0,
    height:300.0,),
   Text("Hi, I'm Pizza Assistant, what can I help you order?",
    style: TextStyle(decoration:TextDecoration.none,fontSize:30.0),
    textAlign: TextAlign.center,
   ),
  ]
  )
  ),
  ),
  MaterialApp(home:Center(child:Column(children:[
   Image.asset('Images/cheesepizza.png',
    width:400.0,
    height:300.0,),
   Text("Hi, I'm Pizza Assistant, what can I help you order?",
                  style: TextStyle(decoration:TextDecoration.none,fontSize:30.0),
                  textAlign: TextAlign.center,
                ),
              ]
              )
              ),
              ),
              MaterialApp(home:Center(child:Column(children:[
                Image.asset('Images/Fpizza.png',
                  width:400.0,
                  height:300.0,),
                Text("Hi, I'm Pizza Assistant, what can I help you order?",
                  style: TextStyle(decoration:TextDecoration.none,fontSize:30.0),
                  textAlign: TextAlign.center,
                ),
              ]
              )
              ),
              ),
            ],
          ),
        ));
  }
}
```

LAB-10
main.dart

```
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/material.dart';
import 'package:lab10/login.dart';

void main() async{
  WidgetsFlutterBinding.ensureInitialized();
  await Firebase.initializeApp();
  runApp(MyApp());
}
class MyApp extends StatelessWidget {
  final Future<FirebaseApp> _initialization = Firebase.initializeApp();
  @override
  Widget build(BuildContext context) {
    return FutureBuilder(
        future: _initialization,
        builder: (context, snapshot) {
          // Check for Errors
          if (snapshot.hasError) {
            print("Something Went Wrong");
          }
          if (snapshot.connectionState == ConnectionState.waiting) {
            return Center(child: CircularProgressIndicator());
          }
          return MaterialApp(
            title: 'Flutter Firebase EMail Password Auth',
            theme: ThemeData(
              primarySwatch: Colors.deepOrange,
            ),
            debugShowCheckedModeBanner: false,
            home: Login(),
          );
        });
  }
}
```

Login.dart

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:lab10/signup.dart';
import 'package:lab10/dashboard.dart';

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

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

class _LoginState extends State<Login> {
  final _formKey = GlobalKey<FormState>();

  var email = "";
  var password = "";
  // Create a text controller and use it to retrieve the current value
  // of the TextField.
  final emailController = TextEditingController();
  final passwordController = TextEditingController();

  userLogin() async {
    try {
      await FirebaseAuth.instance
          .signInWithEmailAndPassword(email: email, password: password);
      Navigator.pushReplacement(
        context,
        MaterialPageRoute(
          builder: (context) => Dashboard(),
        ),
      );
    } on FirebaseAuthException catch (e) {
```

```
    if (e.code == 'user-not-found') {
      print("No User Found for that Email");
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          backgroundColor: Colors.orangeAccent,
          content: Text(
            "No User Found for that Email",
            style: TextStyle(fontSize: 18.0, color: Colors.black),
          ),
        ),
      );
    } else if (e.code == 'wrong-password') {
      print("Wrong Password Provided by User");
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          backgroundColor: Colors.orangeAccent,
          content: Text(
            "Wrong Password Provided by User",
            style: TextStyle(fontSize: 18.0, color: Colors.black),
          ),
        ),
      );
    }
  }
}

@override
void dispose() {
  // Clean up the controller when the widget is disposed.
  emailController.dispose();
  passwordController.dispose();
  super.dispose();
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("User Login"),
    ),
    body: Form(
      key: _formKey,
      child: Padding(
        padding: EdgeInsets.symmetric(vertical: 20, horizontal: 30),
        child: ListView(
          children: [
            Container(
              margin: EdgeInsets.symmetric(vertical: 10.0),
  child: TextFormField(
    autofocus: false,
    decoration: InputDecoration(
      labelText: 'Email: ',
      labelStyle: TextStyle(fontSize: 20.0),
      border: OutlineInputBorder(),
      errorStyle:
      TextStyle(color: Colors.redAccent, fontSize: 15),
    ),
    controller: emailController,
    validator: (value) {
      if (value == null || value.isEmpty) {
        return 'Please Enter Email';
      } else if (!value.contains('@')) {
        return 'Please Enter Valid Email';
      }
      return null;
    },
  ),
),
Container(
  margin: EdgeInsets.symmetric(vertical: 10.0),
  child: TextFormField(
    autofocus: false,
    obscureText: true,
    decoration: InputDecoration(
      labelText: 'Password: ',
      labelStyle: TextStyle(fontSize: 20.0),
      border: OutlineInputBorder(),
      errorStyle:
      TextStyle(color: Colors.redAccent, fontSize: 15),
    ),
    controller: passwordController,
    validator: (value) {
      if (value == null || value.isEmpty) {
        return 'Please Enter Password';
      }
      return null;
    },
  ),
),
Container(
  margin: EdgeInsets.only(left: 60.0),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      ElevatedButton(
        onPressed: () {
            // Validate returns true if the form is valid, otherwise false.
            if (_formKey.currentState!.validate()) {
              setState(() {
                email = emailController.text;
                password = passwordController.text;
              });
              userLogin();
            }
          },
          child: Text(
            'Login',
            style: TextStyle(fontSize: 18.0),
          ),
        ),
        TextButton(
          onPressed: () => {
          },
          child: Text(
            'Forgot Password ?',
            style: TextStyle(fontSize: 14.0),
          ),
        ),
      ],
    ),
  ),
  Container(
    child: Row(
      mainAxisAlignment: MainAxisAlignment.center,
      children: [
        Text("Don't have an Account? "),
        TextButton(
          onPressed: () => {
            Navigator.pushAndRemoveUntil(
                context,
                PageRouteBuilder(
                  pageBuilder: (context, a, b) => Signup(),
                  transitionDuration: Duration(seconds: 0),
                ),
                    (route) => false)
          },
          child: Text('Signup'),
        ),

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

SignUp.dart

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:lab10/Login.dart';

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

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

class _SignupState extends State<Signup> {
  final _formKey = GlobalKey<FormState>();

  var email = "";
  var password = "";
  var confirmPassword = "";
  // Create a text controller and use it to retrieve the current value
  // of the TextField.
  final emailController = TextEditingController();
  final passwordController = TextEditingController();
  final confirmPasswordController = TextEditingController();

  @override
  void dispose() {
    // Clean up the controller when the widget is disposed.
    emailController.dispose();
    passwordController.dispose();
    confirmPasswordController.dispose();
    super.dispose();
  }

  registration() async {
    if (password == confirmPassword) {
      try {
        UserCredential userCredential = await FirebaseAuth.instance
            .createUserWithEmailAndPassword(email: email, password: password);
        print(userCredential);
        ScaffoldMessenger.of(context).showSnackBar(
      SnackBar(
        backgroundColor: Colors.redAccent,
        content: Text(
          "Registered Successfully. Please Login..",
          style: TextStyle(fontSize: 20.0),
        ),
      ),
    );
    Navigator.pushReplacement(
      context,
      MaterialPageRoute(
        builder: (context) => Login(),
      ),
    );
  } on FirebaseAuthException catch (e) {
    if (e.code == 'weak-password') {
      print("Password Provided is too Weak");
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          backgroundColor: Colors.orangeAccent,
          content: Text(
            "Password Provided is too Weak",
            style: TextStyle(fontSize: 18.0, color: Colors.black),
          ),
        ),
      );
    } else if (e.code == 'email-already-in-use') {
      print("Account Already exists");
      ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          backgroundColor: Colors.orangeAccent,
          content: Text(
            "Account Already exists",
            style: TextStyle(fontSize: 18.0, color: Colors.black),
          ),
        ),
      );
    }
  }
} else {
  print("Password and Confirm Password doesn't match");
  ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(
      backgroundColor: Colors.orangeAccent,
      content: Text(
        "Password and Confirm Password doesn't match",
        style: TextStyle(fontSize: 16.0, color: Colors.black),
      ),
      ),
    );
  }
}

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: Text("User SignUp"),
    ),
    body: Form(
      key: _formKey,
      child: Padding(
        padding: EdgeInsets.symmetric(vertical: 20, horizontal: 30),
        child: ListView(
          children: [
            Container(
              margin: EdgeInsets.symmetric(vertical: 10.0),
              child: TextFormField(
                autofocus: false,
                decoration: InputDecoration(
                  labelText: 'Email: ',
                  labelStyle: TextStyle(fontSize: 20.0),
                  border: OutlineInputBorder(),
                  errorStyle:
                  TextStyle(color: Colors.redAccent, fontSize: 15),
                ),
                controller: emailController,
                validator: (value) {
                  if (value == null || value.isEmpty) {
                    return 'Please Enter Email';
                  } else if (!value.contains('@')) {
                    return 'Please Enter Valid Email';
                  }
                  return null;
                },
              ),
            ),
            Container(
              margin: EdgeInsets.symmetric(vertical: 10.0),
              child: TextFormField(
                autofocus: false,
                obscureText: true,
                decoration: InputDecoration(
                  labelText: 'Password: ',
                  labelStyle: TextStyle(fontSize: 20.0),
                  border: OutlineInputBorder(),
      errorStyle:
      TextStyle(color: Colors.redAccent, fontSize: 15),
    ),
    controller: passwordController,
    validator: (value) {
      if (value == null || value.isEmpty) {
        return 'Please Enter Password';
      }
      return null;
    },
  ),
),
Container(
  margin: EdgeInsets.symmetric(vertical: 10.0),
  child: TextFormField(
    autofocus: false,
    obscureText: true,
    decoration: InputDecoration(
      labelText: 'Confirm Password: ',
      labelStyle: TextStyle(fontSize: 20.0),
      border: OutlineInputBorder(),
      errorStyle:
      TextStyle(color: Colors.redAccent, fontSize: 15),
    ),
    controller: confirmPasswordController,
    validator: (value) {
      if (value == null || value.isEmpty) {
        return 'Please Enter Password';
      }
      return null;
    },
  ),
),
Container(
  child: Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      ElevatedButton(
        onPressed: () {
          // Validate returns true if the form is valid, otherwise false.
          if (_formKey.currentState!.validate()) {
            setState(() {
              email = emailController.text;
              password = passwordController.text;
              confirmPassword = confirmPasswordController.text;
            });
            registration();
          }
```

```
                      },
                      child: Text(
                        'Sign Up',
                        style: TextStyle(fontSize: 18.0),
                      ),
                    ),
                  ],
                ),
              ),
              Container(
                child: Row(
                  mainAxisAlignment: MainAxisAlignment.center,
                  children: [
                    Text("Already have an Account? "),
                    TextButton(
                        onPressed: () => {
                          Navigator.pushReplacement(
                            context,
                            PageRouteBuilder(
                              pageBuilder:
                                  (context, animation1, animation2) =>
                                  Login(),
                              transitionDuration: Duration(seconds: 0),
                            ),
                          )
                        },
                        child: Text('Login'))
                  ],
                ),
              )
            ],
          ),
        ),
      ),
    );
  }
}
```

Dashboard.dart

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

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

  @override
  Widget build(BuildContext context) {
    // Material App
  return MaterialApp(
    home:Scaffold(
      appBar: AppBar(
       title: Text("CMR INSTITUTE OF TECHNOLOGY"),
       centerTitle: true,
       backgroundColor: Colors.deepOrange,
      ),
      body: Center(
       child: Image(
        image: NetworkImage('https://i0.wp.com/cmrithyderabad.edu.in/wp-content/uploads/2021/09/cropped-CMR-IT-logo-1.webp?w=731&ssl=1'),
       ),
      )
    ));
 }
}
```

LAB-9

main.dart
SignUp.dart
Login.dart
Dashboard.dart

For Above four files Refer: **Lab-10**

ForgotPassword.dart

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:lab9/login.dart';
import 'package:lab9/signup.dart';

class ForgotPassword extends StatefulWidget {
 ForgotPassword({Key? key}) : super(key: key);
 @override
 _ForgotPasswordState createState() => _ForgotPasswordState();
}
class _ForgotPasswordState extends State<ForgotPassword> {
 final _formKey = GlobalKey<FormState>();
 var email = "";
 final emailController = TextEditingController();
 @override
 void dispose() {
  // Clean up the controller when the widget is disposed.
  emailController.dispose();
  super.dispose();
 }
 resetPassword() async {
  try {
   await FirebaseAuth.instance.sendPasswordResetEmail(email: email);
   ScaffoldMessenger.of(context).showSnackBar(
    SnackBar(
     backgroundColor: Colors.orangeAccent,
     content: Text(
      'Password Reset Email has been sent !',
      style: TextStyle(fontSize: 18.0),
     ),       ),      );    }
on FirebaseAuthException catch (e) {
   if (e.code == 'user-not-found') {
    print('No user found for that email.');
    ScaffoldMessenger.of(context).showSnackBar(
     SnackBar(
      backgroundColor: Colors.orangeAccent,
      content: Text(
       'No user found for that email.',
       style: TextStyle(fontSize: 18.0),
      ),         ),        );      }    }    }

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   appBar: AppBar(
    title: Text("Reset Password"),
   ),
   body: Column(
    children: [
     Container(
      margin: EdgeInsets.only(top: 20.0),
      child: Text(
       'Reset Link will be sent to your email id !',
       style: TextStyle(fontSize: 20.0),
      ),
     ),
     Expanded(
      child: Form(
       key: _formKey,
       child: Padding(
        padding: EdgeInsets.symmetric(vertical: 20, horizontal: 30),
        child: ListView(
         children: [
          Container(
           margin: EdgeInsets.symmetric(vertical: 10.0),
           child: TextFormField(
            autofocus: false,
            decoration: InputDecoration(
             labelText: 'Email: ',
             labelStyle: TextStyle(fontSize: 20.0),
             border: OutlineInputBorder(),
      errorStyle:
      TextStyle(color: Colors.redAccent, fontSize: 15),
    ),
    controller: emailController,
    validator: (value) {
      if (value == null || value.isEmpty) {
        return 'Please Enter Email';
      } else if (!value.contains('@')) {
        return 'Please Enter Valid Email';
      }
      return null;
    },                        ),                      ),
Container(
  margin: EdgeInsets.only(left: 60.0),
  child: Row(
    mainAxisAlignment: MainAxisAlignment.center,
    children: [
      ElevatedButton(
        onPressed: () {
          // Validate returns true if the form is valid, otherwise false.
          if (_formKey.currentState!.validate()) {
            setState(() {
              email = emailController.text;
            });
            resetPassword();
          }
        },
        child: Text(
          'Send Email',
          style: TextStyle(fontSize: 18.0),
        ),
      ),
      TextButton(
        onPressed: () => {
          Navigator.pushAndRemoveUntil(
              context,
              PageRouteBuilder(
                pageBuilder: (context, a, b) => Login(),
                transitionDuration: Duration(seconds: 0),
              ),
                  (route) => false)
        },
        child: Text(
          'Login',
          style: TextStyle(fontSize: 14.0),
        ),                          ),                        ],                      ),                    ),
Container(
  child: Row(
            mainAxisAlignment: MainAxisAlignment.center,
            children: [
             Text("Don't have an Account? "),
             TextButton(
               onPressed: () => {
                Navigator.pushAndRemoveUntil(
                  context,
                  PageRouteBuilder(
                   pageBuilder: (context, a, b) =>
                     Signup(),
                   transitionDuration:
                   Duration(seconds: 0),
                  ),
                    (route) => false)
               },
               child: Text('Signup'))
            ],   ),     )                 ],                ),              ),            ),          ),        ],      ),    );  }        }
```

ChangePassword.dart

```
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:lab9/login.dart';

class ChangePassword extends StatefulWidget {
 ChangePassword({Key? key}) : super(key: key);
 @override
 _ChangePasswordState createState() => _ChangePasswordState();
}
class _ChangePasswordState extends State<ChangePassword> {
 final _formKey = GlobalKey<FormState>();
 var newPassword = "";
 final newPasswordController = TextEditingController();
 @override
 void dispose() {
  // Clean up the controller when the widget is disposed.
  newPasswordController.dispose();
  super.dispose();
 }
 final currentUser = FirebaseAuth.instance.currentUser;
 changePassword() async {
  try {
   await currentUser!.updatePassword(newPassword);
   FirebaseAuth.instance.signOut();
   Navigator.pushReplacement(
    context,
    MaterialPageRoute(builder: (context) => Login()),
   );
   ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
          backgroundColor: Colors.orangeAccent,
          content: Text(
            'Your Password has been Changed. Login again !',
            style: TextStyle(fontSize: 18.0),
          ),        ),      );    }
catch (e) {
                        }
  }
  @override
  Widget build(BuildContext context) {
    return Form(
      key: _formKey,
      child: Padding(
        padding: EdgeInsets.symmetric(vertical: 20, horizontal: 30),
        child: ListView(
          children: [
            Container(
              margin: EdgeInsets.symmetric(vertical: 10.0),
              child: TextFormField(
                autofocus: false,
                obscureText: true,
                decoration: InputDecoration(
                  labelText: 'New Password: ',
                  hintText: 'Enter New Password',
                  labelStyle: TextStyle(fontSize: 20.0),
                  border: OutlineInputBorder(),
                  errorStyle: TextStyle(color: Colors.redAccent, fontSize: 15),
                ),
                controller: newPasswordController,
                validator: (value) {
                  if (value == null || value.isEmpty) {
                    return 'Please Enter Password';
                  }
                  return null;
                },              ),            ),
            ElevatedButton(
              onPressed: () {
                if (_formKey.currentState!.validate()) {
                  setState(() {
                    newPassword = newPasswordController.text;
                  });
                  changePassword();
                }
              },
              child: Text(
                'Change Password',
                style: TextStyle(fontSize: 18.0),              ),            ),          ],        ),      ),    );  }    }
```