Aim -
To Connect Flutter UI with firebase database.

Theory -
Firebase is a product of Google which helps developers to build, manage, and grow their apps easily. It helps developers to build their apps faster and in a more secure way. No programming is required on the firebase side which makes it easy to use its features more efficiently. It provides services to android, ios, web, and unity. It provides cloud storage. It uses NoSQL for the database for the storage of data.

Features of Firebase:
Mainly there are 3 categories in which firebase provides its services.

Build better applications
This feature mainly includes backend services that help developers to build and manage their applications in a better way. Services included under this feature are :

Realtime Database: The Firebase Realtime Database is a cloud-based NoSQL database that manages your data at the blazing speed of milliseconds. In simplest term, it can be considered as a big JSON file.

Cloud Firestore: The cloud Firestore is a NoSQL document database that provides services like store, sync, and query through the application on a global scale. It stores data in the form of objects also known as Documents. It has a key-value pair and can store all kinds of data like, strings, binary data, and even JSON trees.

Authentication: Firebase Authentication service provides easy to use UI libraries and SDKs to authenticate users to your app. It reduces the manpower and effort required to develop and maintain the user authentication service. It even handles tasks like merging accounts, which if done manually can be hectic.

Remote Config: The remote configuration service helps in publishing updates to the user immediately. The changes can range from changing components of the UI to changing the behavior of the applications. These are often used while publishing seasonal offers and contents to the application that has a limited life.

Hosting: Firebase provides hosting of applications with speed and security. It can be used to host Static or Dynamic websites and microservices. It has the capability of hosting an application with a single command.
Firebase Cloud Messaging(FCM): The FCM service provides a connection between the server and the application end users, which can be used to receive and send messages and notifications. These connections are reliable and battery-efficient.

Code-

Auth.dart

```
import
'package:firebase_auth/firebase_auth.dart';

class AuthService {
  final _auth = FirebaseAuth.instance;

  Stream<User?> get userStream {
    return _auth.authStateChanges();
  }

  // Sign in anonymously
  Future signInAnonymously() async {
    try {
      UserCredential result = await
_auth.signInAnonymously();
      User? user = result.user;
      return user;
    } catch (e) {
      print(e.toString());
      return null;
    }
  }

  // Register using email and password
  Future register(email, password) async {
    try {
      UserCredential r = await
_auth.createUserWithEmailAndPassword(
          email: email, password: password);
      return r.user;
    } catch (e) {
      print(e.toString());
      return null;
    }
  }

  // Sign in using email and password
  Future signInEmailPassword(String email,
String password) async {
    try {
      var r = await
_auth.signInWithEmailAndPassword(
          email: email, password: password);
      print(r.user);
      return r.user;
    } catch (e) {
      print(e.toString());
      return null;
    }
  }

  // Sign out
  Future signOut() async {
    try {
      return await _auth.signOut();
    } catch (e) {
      print(e.toString());
      return null;
    }
  }
}
```

Signin.dart

```
import 'package:flutter/material.dart';
import
'package:home_workout_app/screens/authe
nticate/authenticate.dart';
import
'package:home_workout_app/screens/home
/homepage.dart';
import './auth.dart';

class SignIn extends StatefulWidget {
  final Function toggleView;
  SignIn({super.key, required
this.toggleView});
  @override
  State<SignIn> createState() =>
_SignInState();
}

class _SignInState extends State<SignIn> {
  final _auth = AuthService();
  String email = '';
  String password = '';
  var _formKey = GlobalKey<FormState>();

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      body: Container(
        padding:
EdgeInsets.symmetric(horizontal: 20),
        child: Form(
          key: _formKey,
          child: Column(
            mainAxisAlignment:
MainAxisAlignment.center,
            crossAxisAlignment:
CrossAxisAlignment.center,
            children: [
              Text(
                'Sign in',
                style: TextStyle(
                  fontSize: 30,
                  fontWeight: FontWeight.w400,
                  color: Colors.grey[700],
                ),
              ),
              SizedBox(
                height: 30,
              ),
              TextFormField(
                validator: (val) {
                  if (val!.isEmpty) {
                    return "Enter email address!";
                  }
                },
                onChanged: (value) {
                  email = value;
                },
                decoration: InputDecoration(
                  label: Text('Email'),
                  border: OutlineInputBorder(),
                ),
```

```
              ),
              SizedBox(
                height: 10,
              ),
              TextFormField(
                validator: (val) {
                  if (val!.isEmpty) {
                    return "Enter password!";
                  }
                },
                onChanged: (value) {
                  password = value;
                },
                decoration: InputDecoration(
                  label: Text('Password'),
                  border: OutlineInputBorder(),
                ),
                obscureText: true,
              ),
              Row(
                mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                crossAxisAlignment:
CrossAxisAlignment.center,
                children: [
                  TextButton(
                    onPressed: () {
                      widget.toggleView();
                    },
                    child: Text(
                      'Create account',
                      style: TextStyle(
                        color: Colors.blue,
                        fontSize: 15,
                      ),
                    ),
                  ),
                  TextButton(
                    onPressed: () {},
                    child: Text(
                      'Forgot Password?',
                      style: TextStyle(
                        color: Colors.blue,
                      ),
                    ),
                  ),
                ],
              ),
              ElevatedButton(
                onPressed: () async {
                  try {
                    if
(_formKey.currentState!.validate()) {
                      var result =
                          await
_auth.signInEmailPassword(email,
password);
                      print(result);
                      if (result == null) {
                        AlertDialog(
                          title: Text("Error"),
                          content: Text("Enter valid
credentials."),
                        );
                      }
                    }
                  } catch (e) {
                    print(e.toString());
                  }
                },
                child: Text('Login'),
                style: ElevatedButton.styleFrom(
                  padding:
EdgeInsets.symmetric(horizontal: 30,
vertical: 15),
                ),
              ),
              // AlertDialog(
              //   title: Text("Error"),
              //   content: Text("Enter valid
credentials."),
              // )
            ],
          ),
        ),
      ),
    );
  }
```

```
}
```

Authenticate.dart

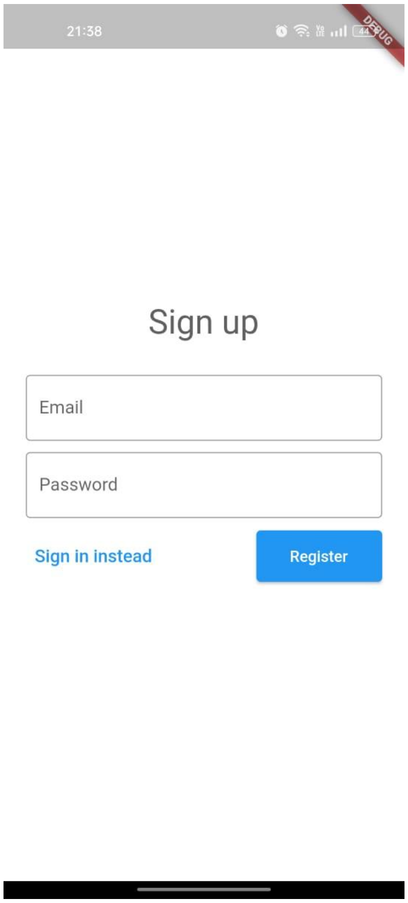
```
import 'package:flutter/material.dart';
import './register.dart';
import './signin.dart';

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

  @override
  State<Authenticate> createState() =>
_AuthenticateState();
}

class _AuthenticateState extends
State<Authenticate> {
  bool showSignIn = true;
  void toggleView() {
    setState(() {
      showSignIn = !showSignIn;
    });
  }

  @override
  Widget build(BuildContext context) {
    if (showSignIn)
      return SignIn(toggleView: toggleView);
    else
      return Register(toggleView: toggleView);
  }
}
```

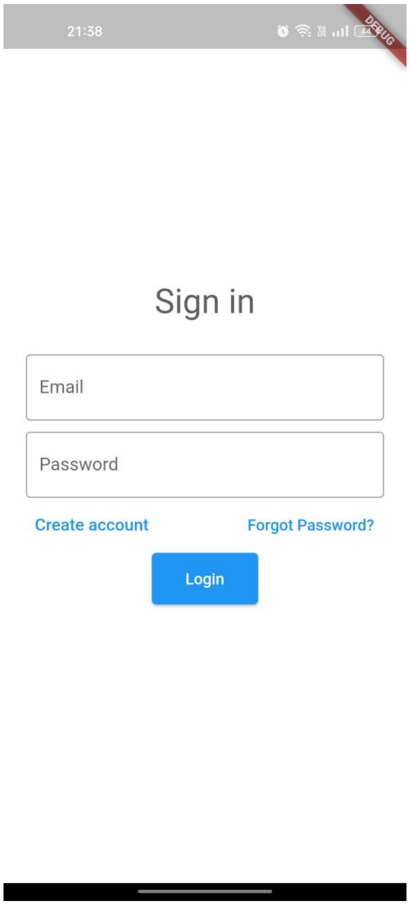
Register.dart

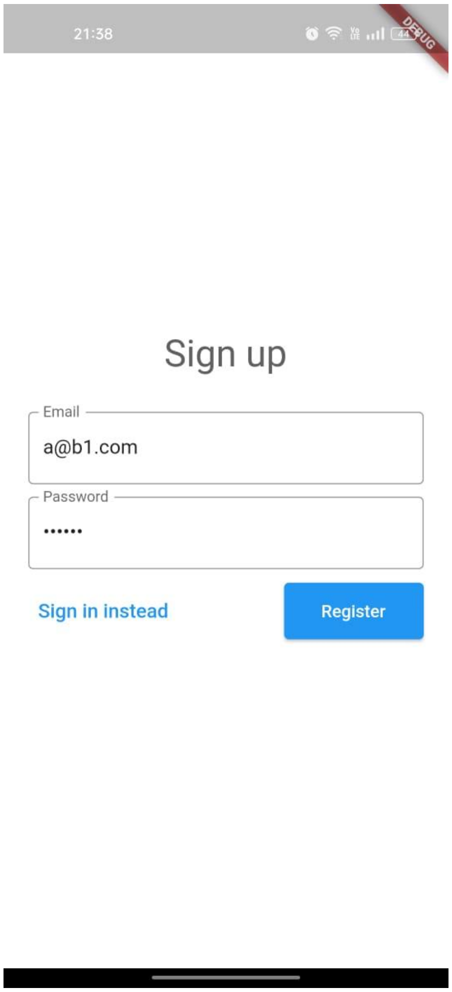
```
import 'package:flutter/material.dart';
import './auth.dart';

class Register extends StatefulWidget {
  final Function toggleView;
  Register({super.key, required
this.toggleView});

  @override
  State<Register> createState() =>
_RegisterState();
}

class _RegisterState extends
State<Register> {
  final _auth = AuthService();
  final _formKey = GlobalKey<FormState>();
  String email = '';
  String password = '';
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      body: Container(
        padding:
EdgeInsets.symmetric(horizontal: 20),
        child: Form(
          key: _formKey,
          child: Column(
            mainAxisAlignment:
MainAxisAlignment.center,
            crossAxisAlignment:
CrossAxisAlignment.center,
            children: [
              Text(
                'Sign up',
                style: TextStyle(
                  fontSize: 30,
                  fontWeight: FontWeight.w400,
                  color: Colors.grey[700],
                ),
              ),
              SizedBox(
                height: 30,
              ),
              TextFormField(
                validator: (value) {
                  if (value!.isEmpty) {
                    return "Field cannot be empty!";
                  } else if (!value.contains('@')) {
                    return "Enter valid Email ID!";
                  }
                },
                onChanged: (value) {
                  email = value;
                },
                decoration: InputDecoration(
                  label: Text('Email'),
                  border: OutlineInputBorder(),
                ),
              ),
              SizedBox(
                height: 10,
              ),
              TextFormField(
                validator: (value) {
                  if (value!.length < 1) {
                    return "Password is too short.";
                  }
                },
                onChanged: (value) {
                  password = value;
                },
                decoration: InputDecoration(
                  label: Text('Password'),
                  border: OutlineInputBorder(),
                ),
                obscureText: true,
              ),
              SizedBox(
                height: 10,
              ),
              Row(
                mainAxisAlignment:
MainAxisAlignment.spaceBetween,
                crossAxisAlignment:
CrossAxisAlignment.center,
                children: [
                  TextButton(
                    onPressed: () {
                      widget.toggleView();
                    },
                    child: Text(
                      'Sign in instead',
                      style: TextStyle(
                        color: Colors.blue,
                        fontSize: 16,
                      ),
                    ),
                  ),
                  ElevatedButton(
                    onPressed: () async {
                      if
(_formKey.currentState!.validate()) {
                        print(email);
                        print(password);
                        print(_auth.register(email,
password));
                      }
                    },
                    child: Text('Register'),
                    style:
ElevatedButton.styleFrom(
```

```
                      padding:
EdgeInsets.symmetric(horizontal: 30,
vertical: 15),
                    ),
                  ),
                ],
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

Output-



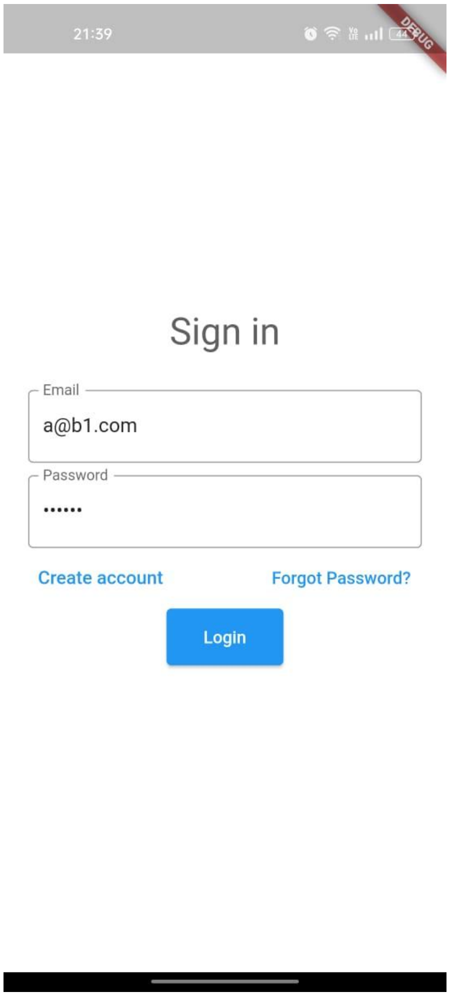




Sign up

Email: a@b1.com

Password: ••••••



Sign in

Email: a@b1.com

Password: ••••••

Conclusion -
We learned how to integrate firebase to our flutter app. We connected firebase to our application and enabled login and sign up using email and password.