



# Modul Praktikum **Pemrograman Mobile**



# FIREBASE FIRESTORE

## TUJUAN PEMBELAJARAN

- A. Mahasiswa memahami dan mengenal layanan-layanan firebase
- B. Mahasiswa memahami dan mengenal layanan firestore
- C. Mahasiswa dapat mengimplementasikan layanan firebase ke dalam sebuah aplikasi

## DASAR TEORI

### ● Firebase

The screenshot shows the official Firebase website. At the top, there's a navigation bar with links for 'Produk', 'Solusi', 'Harga', 'Dokumen', 'Komunitas', 'Dukungan', a search bar, and language options ('Bahasa Ing...'). Below the header, a large blue banner features the text 'Maksimalkan kualitas aplikasi Anda' and a subtext explaining that Firebase is a platform for app development and game optimization, supported by Google and used by millions of businesses worldwide. At the bottom of the banner are three buttons: 'Mulai', 'Coba demo', and 'Tonton video'. To the right of the banner, there's a decorative graphic of colorful geometric shapes (circles, triangles, stars) forming a path leading to a central heart icon.

Firebase adalah salah satu layanan yang disediakan dari Google. Firebase menyediakan beberapa layanan, tiga diantaranya yaitu:

1. **Auth**, digunakan untuk autentikasi login untuk kasus aplikasi yang memiliki fitur login. Auth menyediakan login menggunakan Email dan Password, atau Anonymous Login seperti Sign in with Google, Sign in with GitHub, dll.
2. **Firestore**, merupakan database non-relasional yang berbasis key-value.
3. **Storage**, untuk penyimpanan file.

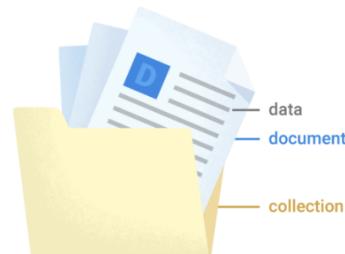


- **Cloud Firestore**

Cloud Firestore adalah salah satu layanan firebase yang disediakan dari Google. Firestore merupakan database yang memberikan kemudahan bagi developer untuk menyimpan data, melakukan sinkronisasi, dan mengambil data ke aplikasi mobile mereka dengan dukungan *offline* yang berkualitas.

Firestore tidak memiliki table atau row, Data disimpan dalam bentuk document. Document berisi data key-value seperti JSON dan dapat bercabang di dalamnya (memiliki object di dalam object). Satu atau lebih document disimpan dalam collection.

SQL	NoSQL “equivalent”
Table	Collection
Row	Document

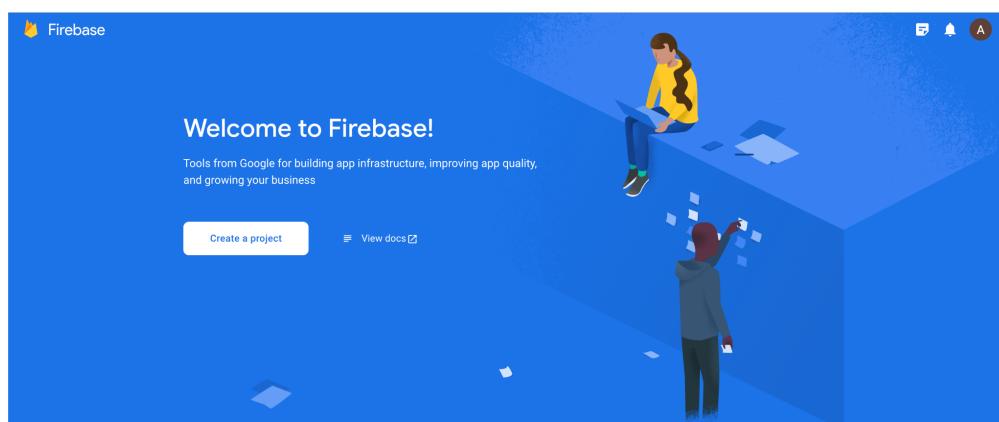


Collection dan document dibuat secara implisit di Cloud Firestore. Jika salah satunya masih belum ada, maka Cloud Firestore akan membuatnya ketika create data.

## Praktikum

- **Membuat Project Firebase**

Buka situs <https://console.firebaseio.google.com> dan pastikan akun Google telah login, kemudian klik *Create a project*.



Setelah itu lakukan beberapa hal di step selanjutnya:



1. Menentukan nama projek untuk mengatur projek di Firebase.
2. **(Optional)** Menentukan project ID. Ini bersifat unik dan tidak dapat diubah setelah projek dibuat.
3. Centang checkbox persetujuan penggunaan project
4. Continue

Let's start with a name for your project®

Project name

Pertemuan 9 Mobile

I accept the [Firebase terms](#)

I confirm that I will use Firebase exclusively for purposes relating to my trade, business, craft, or profession.

Continue

Pada step selanjutnya, **uncheck Enable Google Analytics** karena pada praktikum ini kita tidak menggunakan Google Analytics, **lalu Create project**.

Google Analytics for your Firebase project

Google Analytics is a free and unlimited analytics solution that enables targeting, reporting, and more in Firebase Crashlytics, Cloud Messaging, In-App Messaging, Remote Config, A/B Testing, and Cloud Functions.

Google Analytics enables:

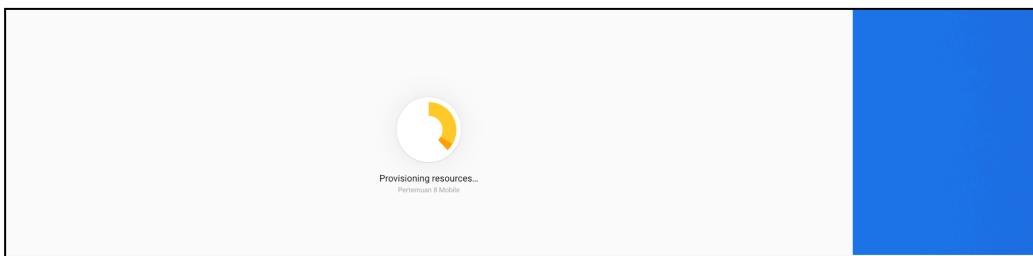
- Analytics
- User segmentation & targeting
- Firebase products
- Data collection

Enable Google Analytics for this project

Unc  
k

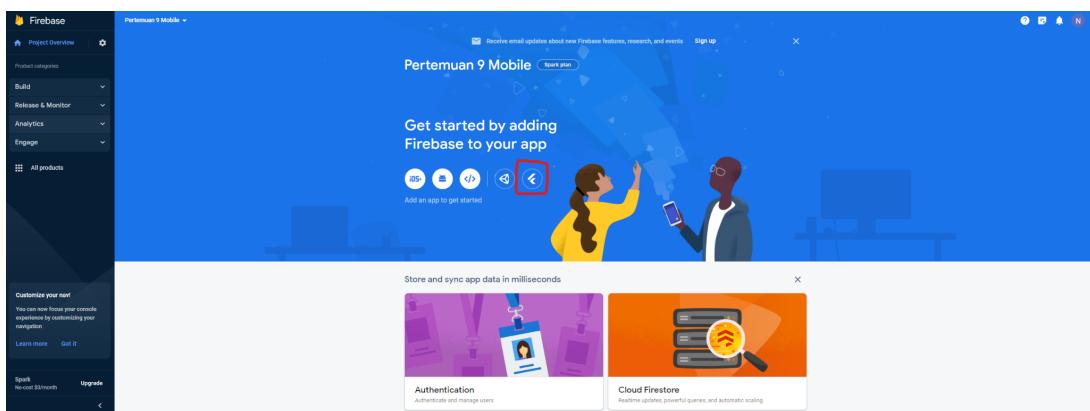
Create project

Tunggu hingga project selesai dibuat.



- **Setup Flutter Project dengan Firebase**

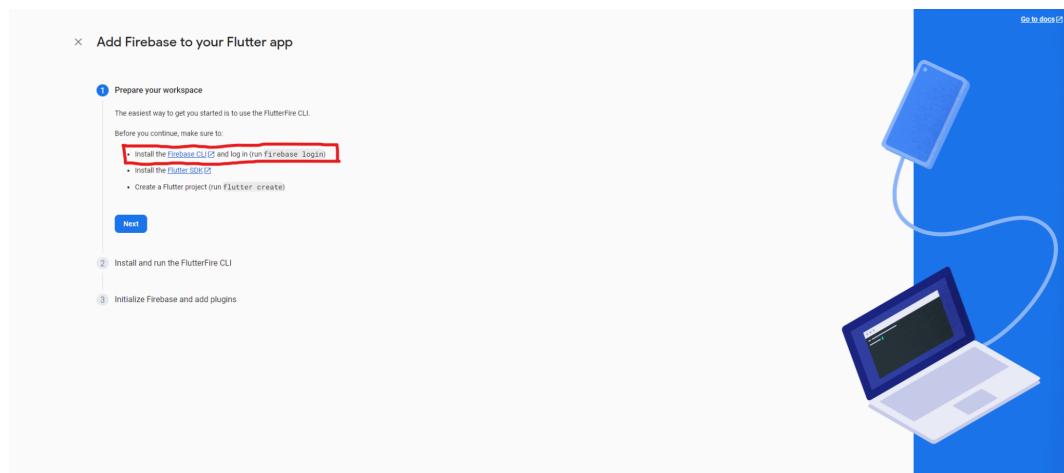
Setelah projek Firebase selesai dibuat maka akan diarahkan ke dashboard. Mari config Firebase ke aplikasi Flutter yang dibuat.





### a. Install Firebase CLI

Pada tahap ini, hal yang wajib kita lakukan adalah **Install Firebase CLI**. Klik link seperti pada gambar dan ikuti perintahnya.



Scroll halaman website hingga menemukan tutorial cara penginstalan **sesuai dengan OS perangkat anda** dan ikuti perintahnya. Berikut adalah tutorial penginstalan untuk Windows.

- ❖ Download **Firebase CLI for Windows** dengan meng-klik link seperti pada gambar berikut.

**Windows**

You can install the Firebase CLI for Windows using one of the following options:

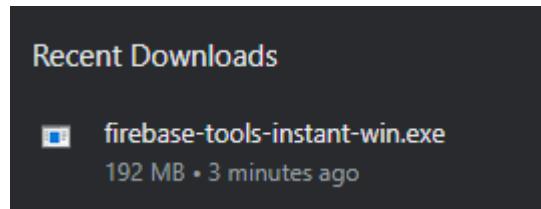
Option	Description	Recommended for...
standalone binary	Download the standalone binary for the CLI. Then, you can access the executable to open a shell where you can run the <code>firebase</code> command.	New developers  Developers not using or unfamiliar with <a href="#">Node.js</a>
npm	Use npm (the Node Package Manager) to install the CLI and enable the globally available <code>firebase</code> command.	Developers using <a href="#">Node.js</a>

**standalone binary**      **npm**

To download and run the binary for the Firebase CLI, follow these steps:

1. Download the [Firebase CLI binary for Windows](#).
2. Access the binary to open a shell where you can run the `firebase` command.
3. Continue to [log in and test the CLI](#).

- ❖ Tunggu hingga proses download selesai dan jalankan file yang telah terdownload.



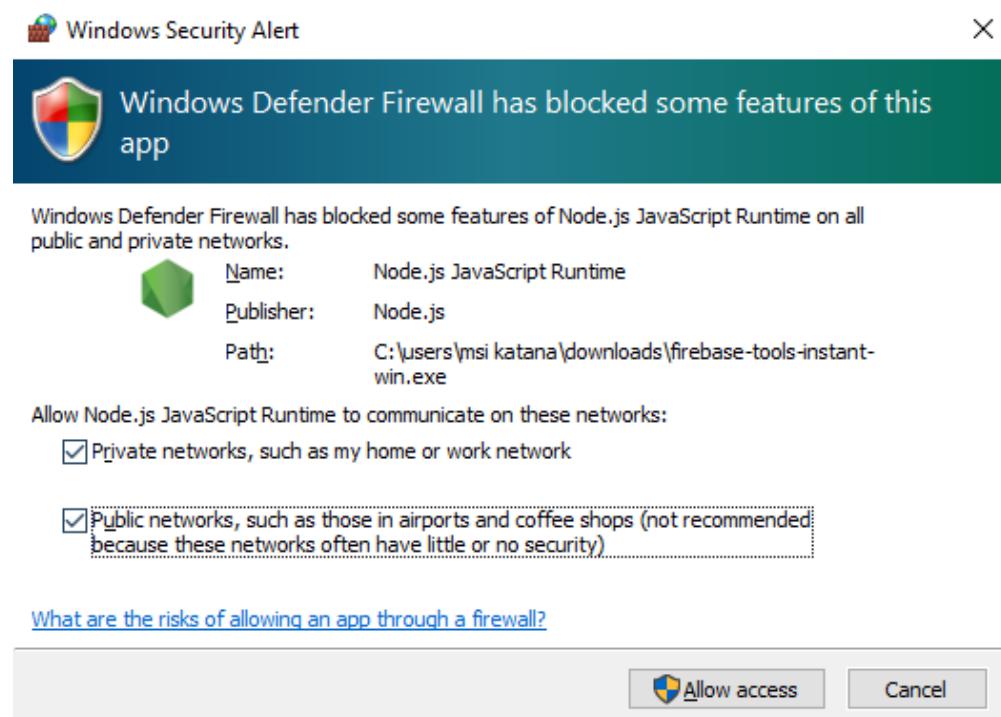
- ❖ Ikuti arahan Instalasi yang ada. bila ada konfirmasi masukkan “Y” lalu **ENTER**.



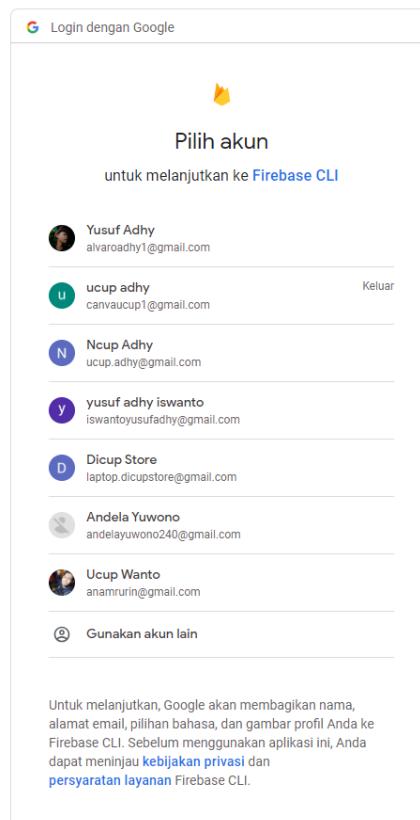
```
Firebase CLI
#####
# Welcome to...
#####
# Let's make sure your Firebase CLI is ready...
# Looks like your CLI needs to be set up.
# This may take a moment
# Setting up environment...-
```

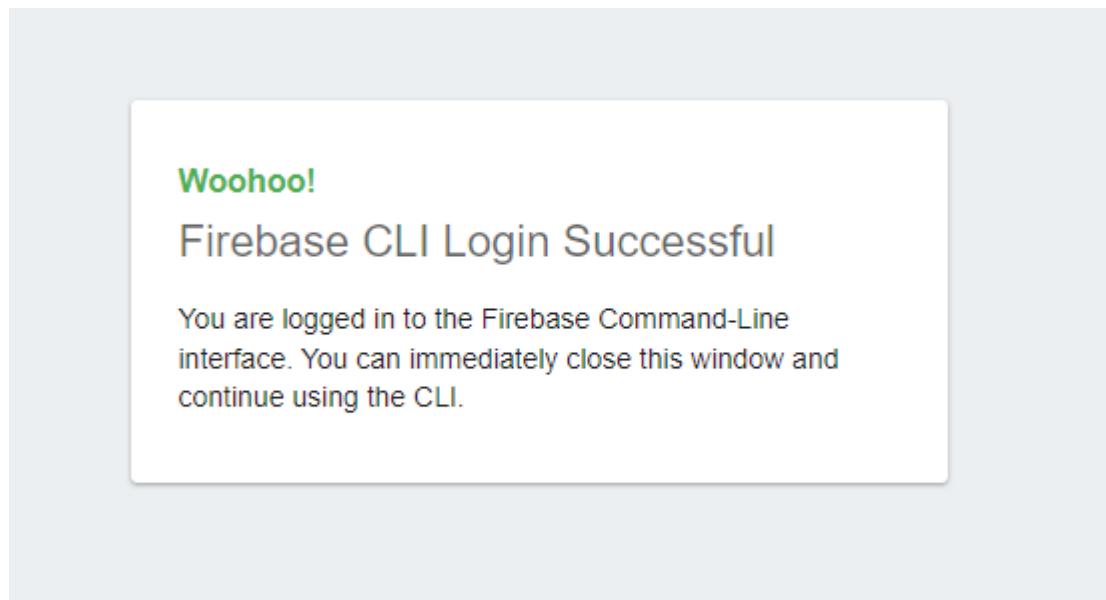
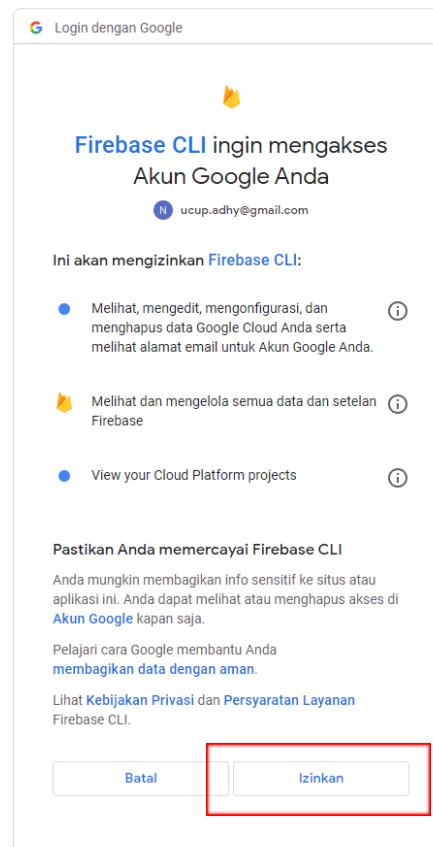
```
Select Firebase CLI
#####
# Welcome to ...
#####
# Let's make sure your Firebase CLI is ready...
# Looks like your CLI needs to be set up.
# This may take a moment
# Alright, your CLI is set up!
# Looks like you're not authenticated. Let's log in!
# Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our product
# Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to
# identify you.
# Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? (Y/n) Y
```

- ❖ Bila muncul popup seperti gambar berikut, centang semua checkbox dan klik **Allow Access**.



- ❖ Selanjutnya akan diarahkan ke halaman Login dan **silakan memilih akun yang ingin anda gunakan.**







- ❖ Kembali ke **Command Line Interface** dan tunggu hingga Instalasi sukses seperti pada gambar.

The screenshot shows a command-line window titled 'C:\Users\MSI KATANA\Downloads\firebase-tools-instant-win.exe'. It displays the following text:

```
C:\Users\MSI KATANA\Downloads\firebase-tools-instant-win.exe
~ Let's make sure your Firebase CLI is ready...
~ Looks like your CLI needs to be set up.

~ This may take a moment
+ Alright, your CLI is set up!

~ Looks like you're not authenticated. Let's log in!
i Firebase optionally collects CLI and Emulator Suite usage and error reporting information to help improve our product
s. Data is collected in accordance with Google's privacy policy (https://policies.google.com/privacy) and is not used to
identify you.

? Allow Firebase to collect CLI and Emulator Suite usage and error reporting information? Yes
i To change your data collection preference at any time, run `firebase logout` and log in again.

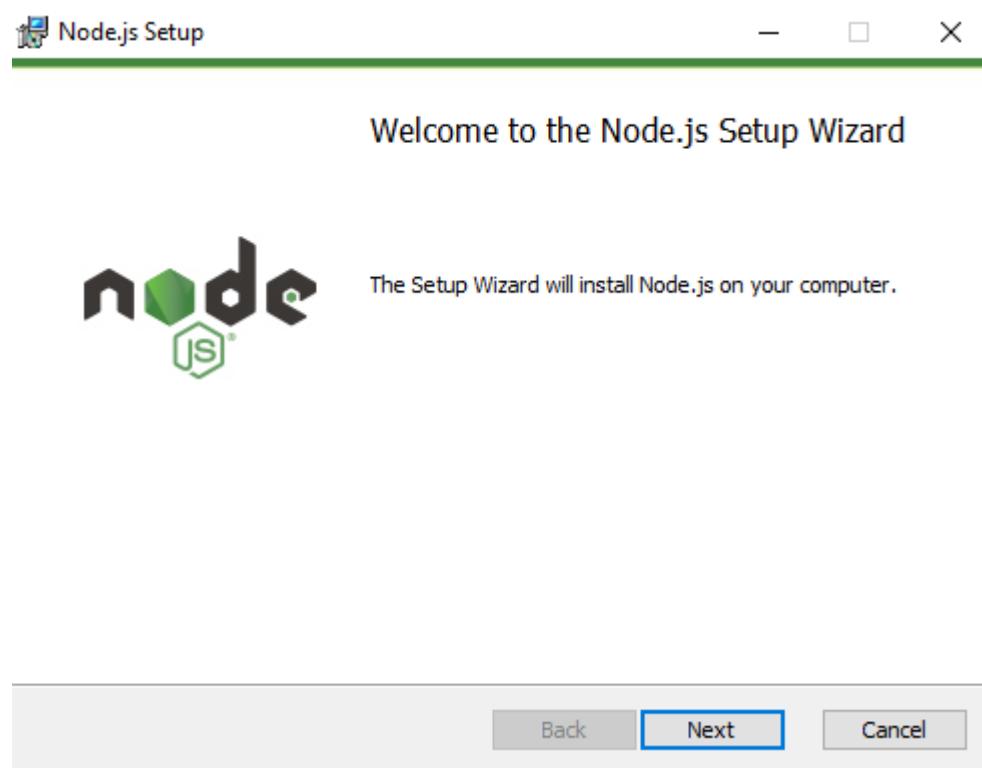
Visit this URL on this device to log in:
https://accounts.google.com/o/oauth2/auth?client\_id=563584335869-fgrhgm47bqnekijs18b5pr03ho849e6.apps.googleusercontent.com&scope=email%20openid%20https%3A%2Fwww.googleapis.com%2Fauth%2Fcldplatformprojects.readonly%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Ffirebase%20https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fcld-cloud-platform&response\_type=code&state=940150834&redirect\_uri=http%3A%2F%2Flocalhost%3A9005

Waiting for authentication...
+ Success! Logged in as ucup.adhy@gmail.com

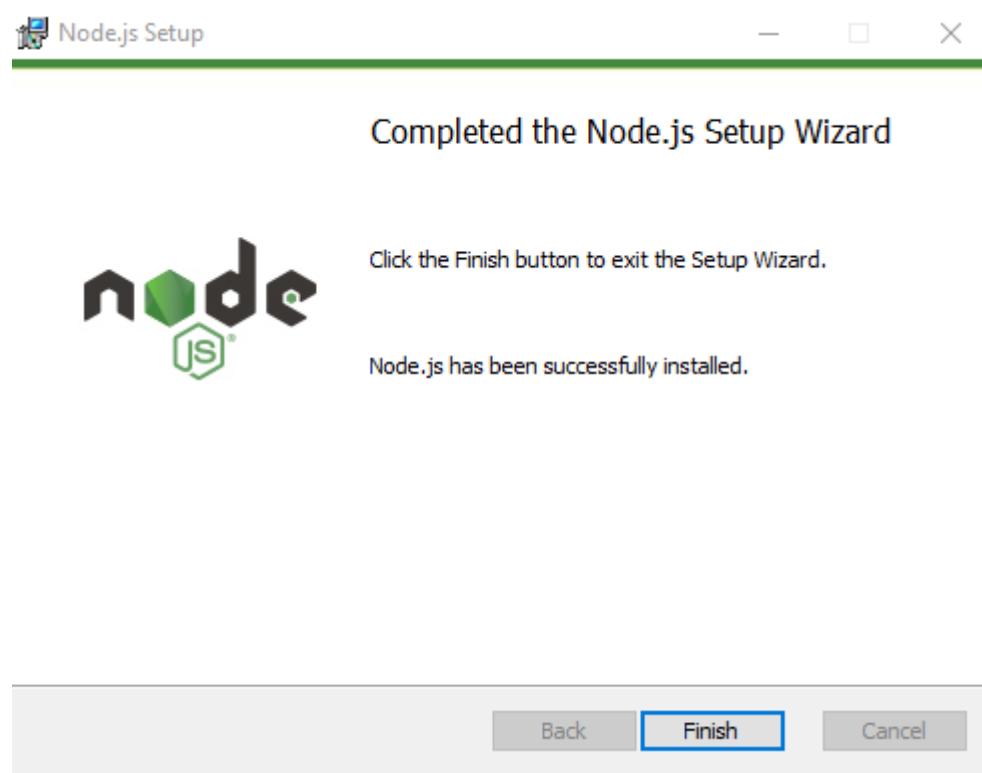
+ You can now use the 'firebase' or 'npm' commands!
~ For more help see https://firebase.google.com/docs/cli
```

### b. Install Node.js

- ❖ Download dan Install Node.js pada link berikut <https://nodejs.org/en>. Download versi yang recomended saja. Pada Setup Instalasi kli **Next** terus dan **Install**.



jika sudah selesai klik **finish**.



### c. Install Firebase Tools

- ❖ Buka CMD lalu jalankan perintah **npm install -g firebase-tools**.

```
C:\Users\MSI KATANA>npm install -g firebase-tools
(c) Microsoft Corporation. All rights reserved.

C:\Users\MSI KATANA>npm install -g firebase-tools
npm WARN deprecated @npmcli/move-file@2.0.1: This functionality has been moved to @npmcli/fs
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated uid@3.4.0: Please upgrade to version 7 or higher. Older versions may use Math.random() in certain
circumstances, which is known to be problematic. See https://v8.dev/blog/math-random for details.

added 660 packages in 3m
54 packages are looking for funding
  run `npm fund` for details
npm notice New minor version of npm available! 10.1.0 -> 10.2.3
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.2.3
npm notice Run npm install -g npm@10.2.3 to update!
npm notice

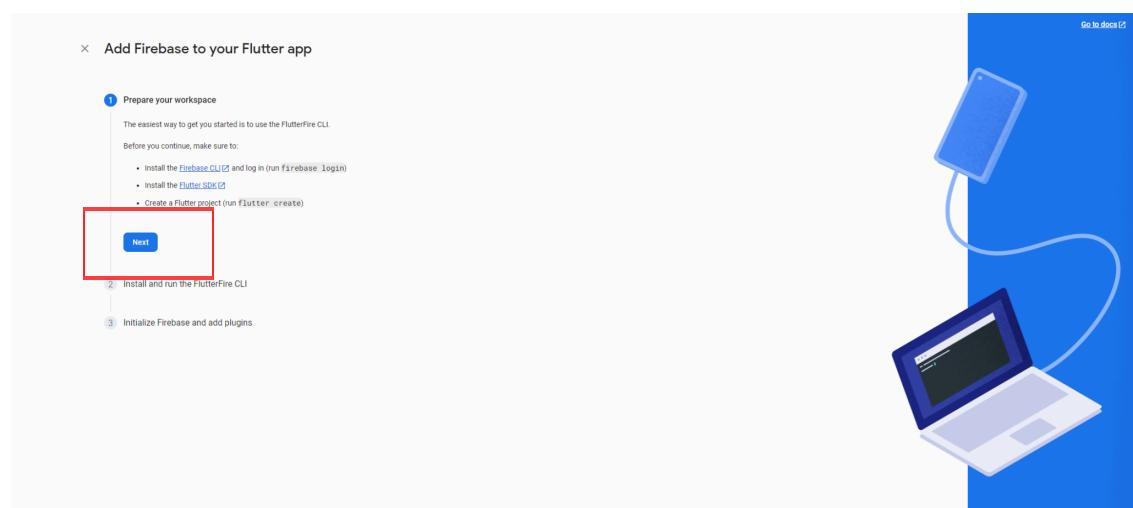
C:\Users\MSI KATANA>
```



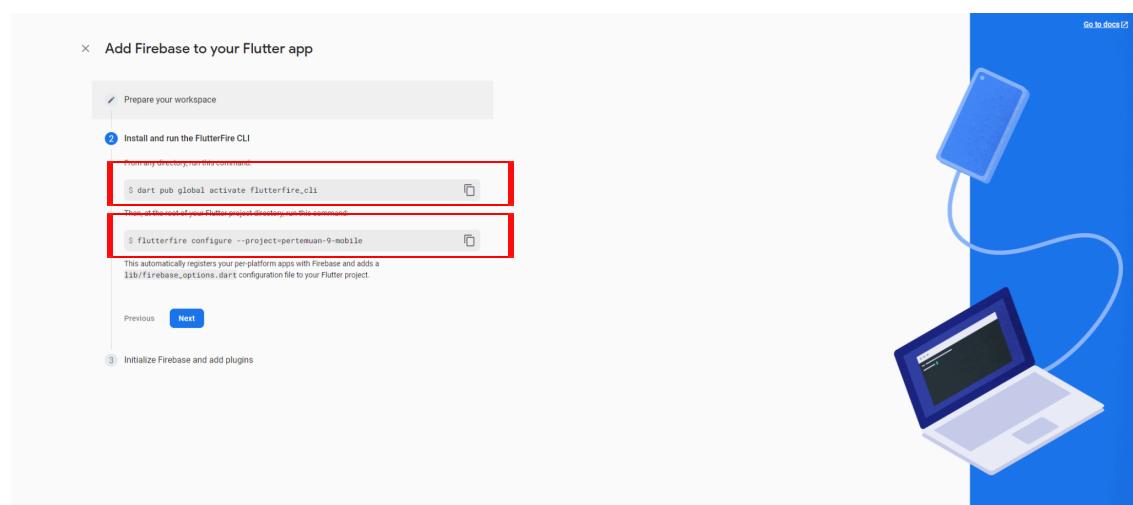
- ❖ Tambahkan **C:\Users\(Nama-PC-Anda)\AppData\Local\Pub\Cache\bin** kedalam PATH baru pada System environment variable.

#### d. Konfigurasi Flutterfire CLI.

- ❖ Setelah melakukan Intalasi selesai silakan kembali ke halaman website Firebase, lalu Klik **Next**.



- ❖ Ikuti langkah pada Step 2 yaitu menjalankan kode pertama pada gambar. **Simpan kedua perintah berikut kedalam notepad agar pengerojan menjadi lebih mudah.**



- ❖ Buka CMD, lalu jalankan perintah pertama di direktori manapun dan tunggu hingga selesai. **Bila ada warning**, Silahkan masukkan direktori yang ada di bagian warning kedalam **SYSTEM PATH**, lalu tutup dan buka lagi CMD anda dan jalankan ulang perintah pertama tadi.

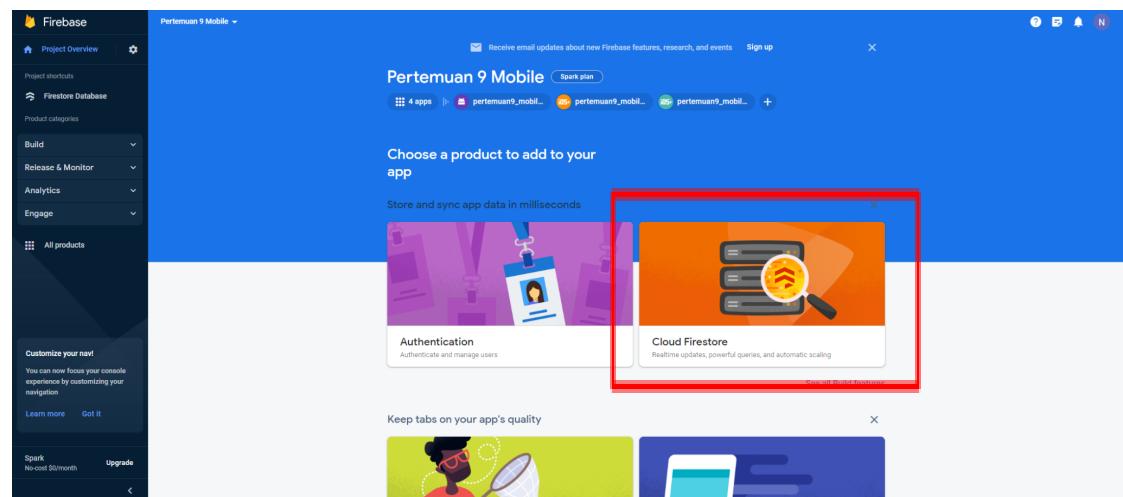
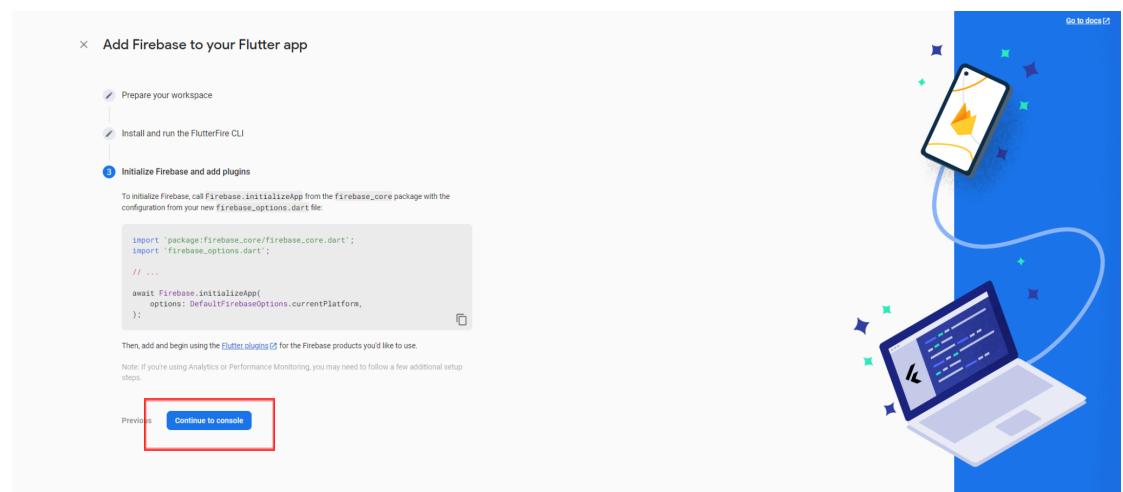
```
C:\Users\MSI KATANA>dart pub global activate flutterfire_cli
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\MSI KATANA>dart pub global activate flutterfire_cli
Package flutterfire_cli is currently active at version 0.2.7.
The package flutterfire_cli is already activated at newest available version.
To recompile executables, first run `dart pub global deactivate flutterfire_cli`.
Installed executable flutterfire.
Activated flutterfire_cli 0.2.7.

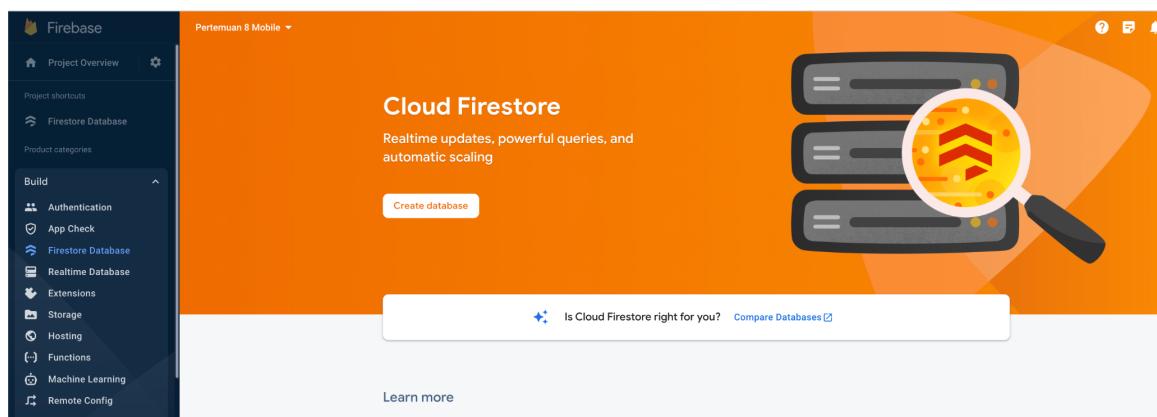
C:\Users\MSI KATANA>
```

#### e. Create Database Firestore

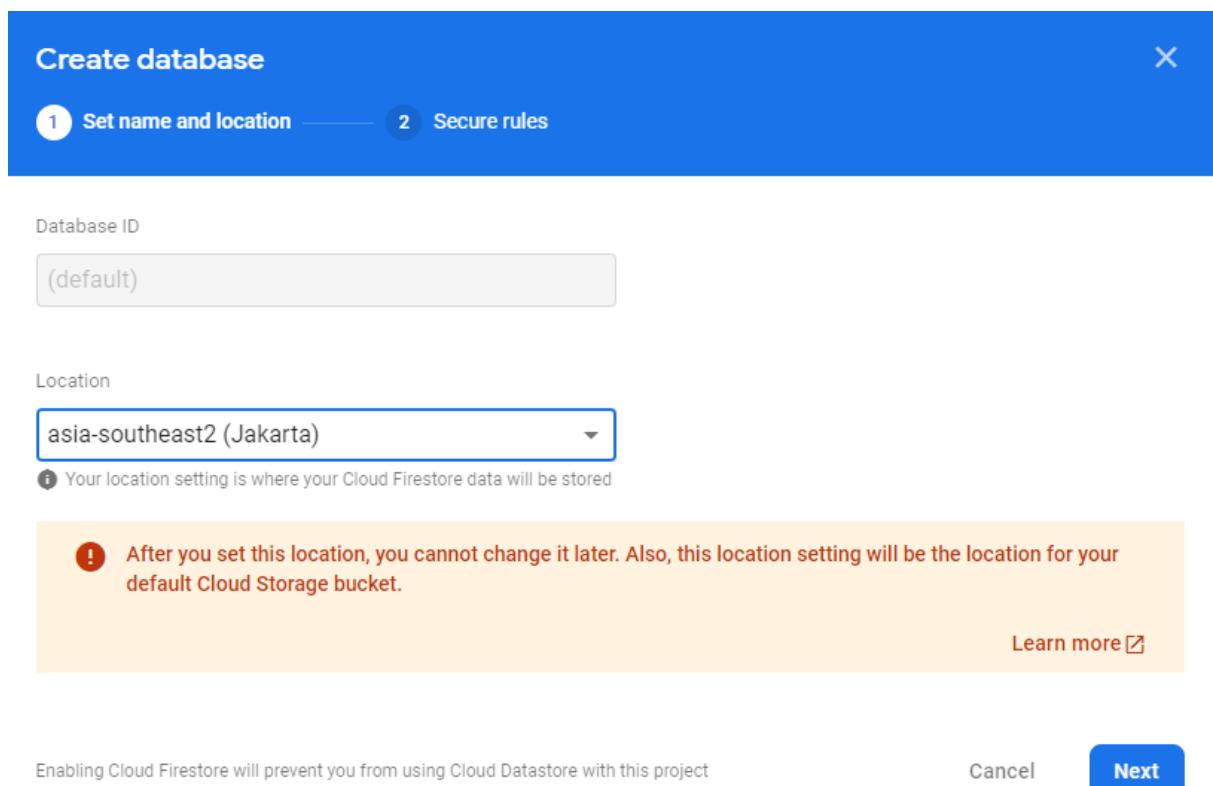
- ❖ Kembali ke website dan klik **Continue to Console**, lalu pilih **Cloud Firestore**.



❖ **Pilih Create Database.**



- ❖ Selanjutnya pilih lokasi Cloud Firestore. Mari pilih lokasi terdekat yaitu **Jakarta (asia-southeast2)**. Lalu klik **Next**.



- ❖ Untuk rule Firestore, karena hanya digunakan untuk test, maka pilih **Start in test mode**. Perlu diingat bahwa dengan menggunakan **test**



**mode**, rule harus diubah kembali setelah 30 hari. Lalu klik **Enable** dan tunggu hingga proses selesai.

The screenshot shows the 'Create database' wizard with the second step, 'Secure rules', selected. It includes instructions to define data structure and write security rules, a link to learn more, and two mode options: 'Start in production mode' (disabled) and 'Start in test mode' (selected). The 'test mode' section contains a note about default security rules allowing anyone to view, edit, and delete data for 30 days. A warning message at the bottom states that enabling Cloud Firestore will prevent using Cloud Datastore, with 'Cancel' and 'Enable' buttons.

After you define your data structure, you will need to write rules to secure your data.  
[Learn more](#)

Start in **production mode**  
Your data is private by default. Client read/write access will only be granted as specified by your security rules.

Start in **test mode**  
Your data is open by default to enable quick setup. However, you must update your security rules within 30 days to enable long-term client read/write access.

```
rules_version = '2';

service cloud.firestore {
    match /databases/{database}/documents {
        match /{document=**} {
            allow read, write: if
                request.time < timestamp.date(2023, 12, 3);
        }
    }
}
```

**!** The default security rules for test mode allow anyone with your database reference to view, edit and delete all data in your database for the next 30 days

Enabling Cloud Firestore will prevent you from using Cloud Datastore with this project

Cancel **Enable**

- ❖ Bila sudah tampil seperti pada gambar berikut, maka **database siap digunakan** pada **Project Flutter** anda.

The screenshot shows the Firebase Cloud Firestore dashboard. The left sidebar lists various services like Project Overview, Firestore Database, and Analytics. The main area displays the Cloud Firestore interface with tabs for Data, Rules, Indexes, Usage, and Extensions. A prominent message says 'Your database is ready to go. Just add data.' with a 'Start collection' button. Navigation icons for Panel view and Query builder are visible.

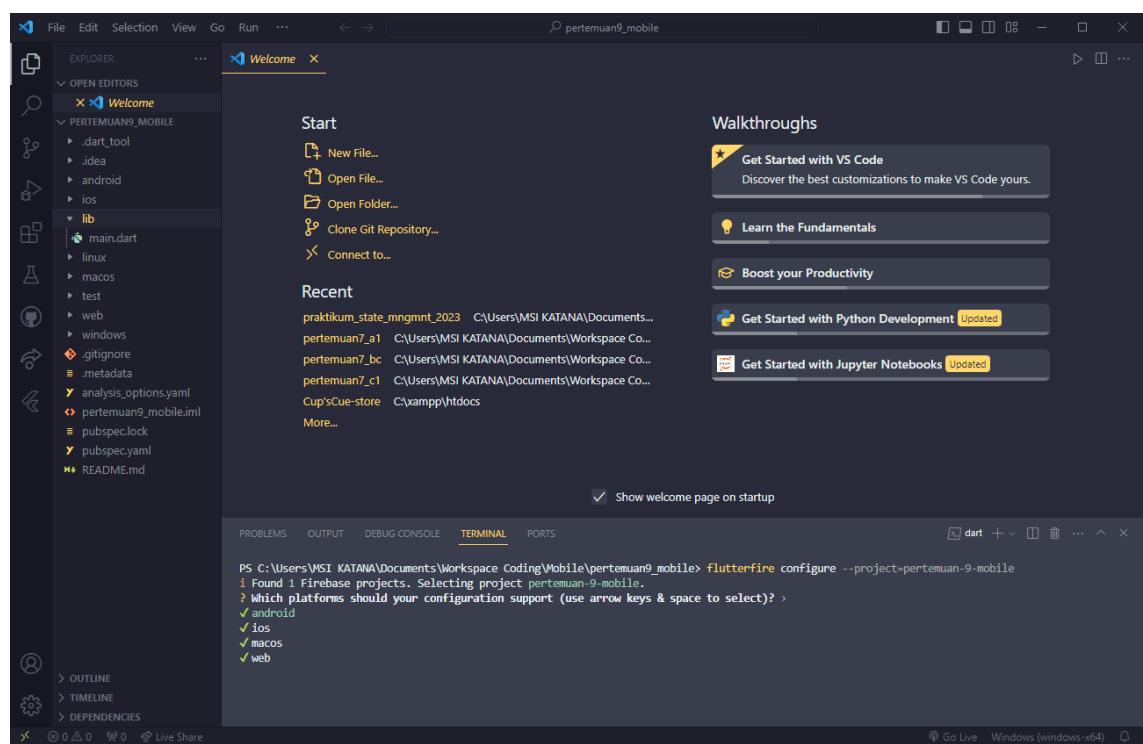


## f. Konfigurasi pada Project Flutter

Silakan **download** projek flutter berikut ini :

[https://drive.google.com/drive/folders/1v\\_ftxJwgWaaN6ZOaQFoAh5lalgQpq7kX?usp=sharing](https://drive.google.com/drive/folders/1v_ftxJwgWaaN6ZOaQFoAh5lalgQpq7kX?usp=sharing)

- ❖ Langkah selanjutnya adalah membuka **terminal** pada **projek flutter** yang telah didownload dan jalankan perintah kedua yang ada di website.



- ❖ Pilih **android** dan tekan enter. dan tunggu hingga proses register berhasil.



The screenshot shows the VS Code interface with the terminal tab active. The terminal window displays the following command and its execution:

```
PS C:\Users\MSI KATANA\Documents\Workspace Coding\Mobile\pertemuan9_mobile> flutterfire configure --project=pertemuan9-mobile
i Found 1 Firebase projects. Selecting project pertemuan9-mobile.
✓ Which platforms should your configuration support (use arrow keys & space to select)? - android, ios, macos, web
i Firebase android app com.example.pertemuan9_mobile is not registered on Firebase project pertemuan9-mobile.
i Registered a new Firebase android app on Firebase project pertemuan9-mobile.
i Firebase ios app com.example.pertemuan9_mobile is not registered on Firebase project pertemuan9-mobile.
i Registered a new Firebase ios app on Firebase project pertemuan9-mobile.
i Firebase macos app com.example.pertemuan9_mobile.RunnerTests is not registered on Firebase project pertemuan9-mobile.
i Registered a new Firebase macos app on Firebase project pertemuan9-mobile.
i Firebase web app pertemuan9_mobile (web) is not registered on Firebase project pertemuan9-mobile.
i Registered a new Firebase web app on Firebase project pertemuan9-mobile.

Firebase configuration file lib.firebaseio_options.dart generated successfully with the following Firebase apps:

Platform  Firebase App Id
web      1:589468901782:web:3e9cb39d1e2d467a3c3db7
android  1:589468901782:android:988ec06affab0f973c3db7
ios       1:589468901782:ios:a27c138fa03959f23c3db7
macos     1:589468901782:ios:4588ca20b325269f3c3db7

Learn more about using this file and next steps from the documentation:
> https://firebase.google.com/docs/flutter/setup
PS C:\Users\MSI KATANA\Documents\Workspace Coding\Mobile\pertemuan9_mobile>
```

- ❖ Buka **firestore\_options.dart**, lalu lakukan quick fix dan pilih **Add 'firebase\_core' to dependencies.**



```
// File generated by FlutterFire CLI.
// ignore_for_file: Lines_longer_than_80_chars, avoid_classes_with_only_static_members
import 'package:firebase_core/firebase_core.dart' show FirebaseOptions;
Quick Fix
Ignore 'depend_on_referenced_packages' for this line
Add 'depend_on_referenced_packages' for the whole file
Add 'firebase_core' to dependencies
More Actions...
Convert to double quoted string
Convert to multiline string
<.../>
class DefaultFirebaseOptions {
    static FirebaseOptions get currentPlatform {
        if (kIsWeb) {
            return web;
        }
        switch (defaultTargetPlatform) {
            case TargetPlatform.android:
                return android;
            case TargetPlatform.iOS:
                return ios;
            case TargetPlatform.macOS:
                return macos;
            case TargetPlatform.windows:
                throw UnsupportedError(
                    'DefaultFirebaseOptions have not been configured for windows - '
                    'you can reconfigure this by running the FlutterFire CLI again.');
        }
    }
}
```

- ❖ Tambahkan dependencies **cloud\_firestore** kedalam **pubspec.yaml**. Salin **cloud\_firestore** pada website <https://pub.dev/>. Pastikan indentasinya benar.



```
main.dart pubspec.yaml MyHomePage.dart 1 LihatData.dart
pubspec.yaml
version: 1.0.0+1
environment:
  sdk: '>=3.1.0 <4.0.0'

dependencies:
  cloud_firestore: ^4.12.2 ←
  flutter:
    sdk: flutter

cupertino_icons: ^1.0.2
firebase_core: ^2.21.0

dev_dependencies:
  flutter_test:
    sdk: flutter

# The "flutter_lints" package below contains a set of recommended lints to
# encourage good coding practices. The lint set provided by the package is
# activated in the 'analysis_options.yaml' file located at the root of your
```

- ❖ Buka File **build.gradle** pada direktori **android/app** lalu ubah **minSdkVersion** menjadi **21**. **Hati-hati! didalam projek terdapat beberapa build.gradle, jangan sampai salah direktori.**



The screenshot shows the Android Studio interface with the following details:

- File Bar:** File, Edit, Selection, View, Go, Run, Terminal, Help.
- Toolbar:** Includes icons for Explorer, Open Editors, Editor, Timeline, and Dependencies.
- Left Sidebar (OPEN EDITORS):**
  - main.dart.lib
  - build.gradle android...
  - MyHomePage.dart.lib
  - pubspec.yaml
- Central Area (Editor):** The build.gradle file for the android/app module. The file content is as follows:

```
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68

kotlinOptions {
    jvmTarget = '1.8'
}

sourceSets {
    main.java.srcDirs += 'src/main/kotlin'
}

defaultConfig {
    // TODO: Specify your own unique Application ID (https://developer.android.com/studio/build/application-id.html).
    applicationId "com.example.pertemuan9_test"
    // You can update the following values to match your application needs.
    // For more information, see: https://docs.flutter.dev/deployment/android#reviewing-the-gradle-build-configuration.
    minSdkVersion 21
    targetSdkVersion flutter.targetSdkVersion
    versionCode flutterVersionCode.toInt()
    versionName flutterVersionName
}

buildTypes {
    release {
        // TODO: Add your own signing config for the release build.
        // Signing with the debug keys for now, so `flutter run --release` works.
        signingConfig signingConfigs.debug
    }
}

flutter {
    source '../..'
}

dependencies {
```

A red arrow points to the line `minSdkVersion 21`.

- ❖ Ubah **void main** yang ada di file **main.dart** menjadi seperti berikut :
    - Pada file lib/main.dart, ubah codingan pada fungsi main menjadi seperti dibawah. Hal-hal yang perlu diperhatikan adalah:
    - **await Firebase.initializeApp();** ditambahkan untuk inisialisasi Firebase. Karena tipe datanya Future, maka diberi awalan await.
    - **void main()** perlu diberi tambahan **async** karena adanya fungsi Future agar dapat berjalan secara asinkron.
    - Jika terdapat perintah yang dijalankan sebelum **runApp(MyApp());** tambahkan perintah **WidgetsFlutterBinding.ensureInitialized();** pada awalan fungsi main().

```
void main() async {  
    WidgetsFlutterBinding.ensureInitialized();  
    await Firebase.initializeApp(  
        options: DefaultFirebaseOptions.currentPlatform);  
    runApp(MyApp());  
}
```



```
options: DefaultFirebaseOptions.currentPlatform,  
);  
runApp(const MyApp());  
}
```

The screenshot shows the VS Code interface with the main.dart file open in the center editor pane. The code implements Firebase initialization and defines the MyApp widget. The left sidebar shows the project structure with files like firebase\_options.dart, lib/main.dart, and pubspec.yaml.

```
File Edit Selection View Go Run ... pertemuan9_mobile ...  
EXPLORER lib > main.dart x ...  
OPEN EDITORS lib > main.dart lib ...  
PERTEMUAN9_MOBILE .dart_tool .idea android ios lib firebase_options.dart lib main.dart lib linux macos test web windows flutter-plugins flutter-plugins-dependencies ignore metadata analysis_options.yaml pertemuan9_mobile.iml pubspec.lock pubspec.yaml README.md ...  
main.dart x lib > main.dart > ...  
1 import 'package:firebase_core/firebase_core.dart';  
2 import 'package:flutter/material.dart';  
3 import 'package:pertemuan9_mobile.firebaseio_options.dart';  
4  
Run | Debug | Profile  
5 void main() async {  
6 WidgetsFlutterBinding.ensureInitialized();  
7 await Firebase.initializeApp(  
8 options: DefaultFirebaseOptions.currentPlatform,  
9 );  
10 runApp(const MyApp());  
11 }  
12  
13 class MyApp extends StatelessWidget {  
14 const MyApp({super.key});  
15  
// This widget is the root of your application.  
16 @override  
17 Widget build(BuildContext context) {  
18 return MaterialApp(  
19 title: 'Flutter Demo',  
20 theme: ThemeData(  
21 // This is the theme of your application.  
22 //  
23 // TRY THIS: Try running your application with "flutter run". You'll see  
// the application has a blue toolbar. Then, without quitting the app,  
// try changing the seedColor in the colorScheme below to Colors.green  
// and then invoke "hot reload" (save your changes or press the "hot  
// reload" button in a Flutter-supported IDE, or press "r" if you used  
// the command line to start the app).  
24 //  
25 // Notice that the counter didn't reset back to zero; the application  
26 // state is not lost during the reload. To reset the state, use hot  
27 // reload.  
28 //  
29 //  
30 //  
31 //  
32 //  
33 //  
34 //  
35 //  
36 //  
37 //  
38 //  
39 //  
40 //  
41 //  
42 //  
43 //  
44 //  
45 //  
46 //  
47 //  
48 //  
49 //  
50 //  
51 //  
52 //  
53 //  
54 //  
55 //  
56 //  
57 //  
58 //  
59 //  
60 //  
61 //  
62 //  
63 //  
64 //  
65 //  
66 //  
67 //  
68 //  
69 //  
70 //  
71 //  
72 //  
73 //  
74 //  
75 //  
76 //  
77 //  
78 //  
79 //  
80 //  
81 //  
82 //  
83 //  
84 //  
85 //  
86 //  
87 //  
88 //  
89 //  
90 //  
91 //  
92 //  
93 //  
94 //  
95 //  
96 //  
97 //  
98 //  
99 //  
100 //  
101 //  
102 //  
103 //  
104 //  
105 //  
106 //  
107 //  
108 //  
109 //  
110 //  
111 //  
112 //  
113 //  
114 //  
115 //  
116 //  
117 //  
118 //  
119 //  
120 //  
121 //  
122 //  
123 //  
124 //  
125 //  
126 //  
127 //  
128 //  
129 //  
130 //  
131 //  
132 //  
133 //  
134 //  
135 //  
136 //  
137 //  
138 //  
139 //  
140 //  
141 //  
142 //  
143 //  
144 //  
145 //  
146 //  
147 //  
148 //  
149 //  
150 //  
151 //  
152 //  
153 //  
154 //  
155 //  
156 //  
157 //  
158 //  
159 //  
160 //  
161 //  
162 //  
163 //  
164 //  
165 //  
166 //  
167 //  
168 //  
169 //  
170 //  
171 //  
172 //  
173 //  
174 //  
175 //  
176 //  
177 //  
178 //  
179 //  
180 //  
181 //  
182 //  
183 //  
184 //  
185 //  
186 //  
187 //  
188 //  
189 //  
190 //  
191 //  
192 //  
193 //  
194 //  
195 //  
196 //  
197 //  
198 //  
199 //  
200 //  
201 //  
202 //  
203 //  
204 //  
205 //  
206 //  
207 //  
208 //  
209 //  
210 //  
211 //  
212 //  
213 //  
214 //  
215 //  
216 //  
217 //  
218 //  
219 //  
220 //  
221 //  
222 //  
223 //  
224 //  
225 //  
226 //  
227 //  
228 //  
229 //  
230 //  
231 //  
232 //  
233 //  
234 //  
235 //  
236 //  
237 //  
238 //  
239 //  
240 //  
241 //  
242 //  
243 //  
244 //  
245 //  
246 //  
247 //  
248 //  
249 //  
250 //  
251 //  
252 //  
253 //  
254 //  
255 //  
256 //  
257 //  
258 //  
259 //  
260 //  
261 //  
262 //  
263 //  
264 //  
265 //  
266 //  
267 //  
268 //  
269 //  
270 //  
271 //  
272 //  
273 //  
274 //  
275 //  
276 //  
277 //  
278 //  
279 //  
280 //  
281 //  
282 //  
283 //  
284 //  
285 //  
286 //  
287 //  
288 //  
289 //  
290 //  
291 //  
292 //  
293 //  
294 //  
295 //  
296 //  
297 //  
298 //  
299 //  
300 //  
301 //  
302 //  
303 //  
304 //  
305 //  
306 //  
307 //  
308 //  
309 //  
310 //  
311 //  
312 //  
313 //  
314 //  
315 //  
316 //  
317 //  
318 //  
319 //  
320 //  
321 //  
322 //  
323 //  
324 //  
325 //  
326 //  
327 //  
328 //  
329 //  
330 //  
331 //  
332 //  
333 //  
334 //  
335 //  
336 //  
337 //  
338 //  
339 //  
340 //  
341 //  
342 //  
343 //  
344 //  
345 //  
346 //  
347 //  
348 //  
349 //  
350 //  
351 //  
352 //  
353 //  
354 //  
355 //  
356 //  
357 //  
358 //  
359 //  
360 //  
361 //  
362 //  
363 //  
364 //  
365 //  
366 //  
367 //  
368 //  
369 //  
370 //  
371 //  
372 //  
373 //  
374 //  
375 //  
376 //  
377 //  
378 //  
379 //  
380 //  
381 //  
382 //  
383 //  
384 //  
385 //  
386 //  
387 //  
388 //  
389 //  
390 //  
391 //  
392 //  
393 //  
394 //  
395 //  
396 //  
397 //  
398 //  
399 //  
400 //  
401 //  
402 //  
403 //  
404 //  
405 //  
406 //  
407 //  
408 //  
409 //  
410 //  
411 //  
412 //  
413 //  
414 //  
415 //  
416 //  
417 //  
418 //  
419 //  
420 //  
421 //  
422 //  
423 //  
424 //  
425 //  
426 //  
427 //  
428 //  
429 //  
430 //  
431 //  
432 //  
433 //  
434 //  
435 //  
436 //  
437 //  
438 //  
439 //  
440 //  
441 //  
442 //  
443 //  
444 //  
445 //  
446 //  
447 //  
448 //  
449 //  
450 //  
451 //  
452 //  
453 //  
454 //  
455 //  
456 //  
457 //  
458 //  
459 //  
460 //  
461 //  
462 //  
463 //  
464 //  
465 //  
466 //  
467 //  
468 //  
469 //  
470 //  
471 //  
472 //  
473 //  
474 //  
475 //  
476 //  
477 //  
478 //  
479 //  
480 //  
481 //  
482 //  
483 //  
484 //  
485 //  
486 //  
487 //  
488 //  
489 //  
490 //  
491 //  
492 //  
493 //  
494 //  
495 //  
496 //  
497 //  
498 //  
499 //  
500 //  
501 //  
502 //  
503 //  
504 //  
505 //  
506 //  
507 //  
508 //  
509 //  
510 //  
511 //  
512 //  
513 //  
514 //  
515 //  
516 //  
517 //  
518 //  
519 //  
520 //  
521 //  
522 //  
523 //  
524 //  
525 //  
526 //  
527 //  
528 //  
529 //  
530 //  
531 //  
532 //  
533 //  
534 //  
535 //  
536 //  
537 //  
538 //  
539 //  
540 //  
541 //  
542 //  
543 //  
544 //  
545 //  
546 //  
547 //  
548 //  
549 //  
550 //  
551 //  
552 //  
553 //  
554 //  
555 //  
556 //  
557 //  
558 //  
559 //  
560 //  
561 //  
562 //  
563 //  
564 //  
565 //  
566 //  
567 //  
568 //  
569 //  
570 //  
571 //  
572 //  
573 //  
574 //  
575 //  
576 //  
577 //  
578 //  
579 //  
580 //  
581 //  
582 //  
583 //  
584 //  
585 //  
586 //  
587 //  
588 //  
589 //  
590 //  
591 //  
592 //  
593 //  
594 //  
595 //  
596 //  
597 //  
598 //  
599 //  
600 //  
601 //  
602 //  
603 //  
604 //  
605 //  
606 //  
607 //  
608 //  
609 //  
610 //  
611 //  
612 //  
613 //  
614 //  
615 //  
616 //  
617 //  
618 //  
619 //  
620 //  
621 //  
622 //  
623 //  
624 //  
625 //  
626 //  
627 //  
628 //  
629 //  
630 //  
631 //  
632 //  
633 //  
634 //  
635 //  
636 //  
637 //  
638 //  
639 //  
640 //  
641 //  
642 //  
643 //  
644 //  
645 //  
646 //  
647 //  
648 //  
649 //  
650 //  
651 //  
652 //  
653 //  
654 //  
655 //  
656 //  
657 //  
658 //  
659 //  
660 //  
661 //  
662 //  
663 //  
664 //  
665 //  
666 //  
667 //  
668 //  
669 //  
670 //  
671 //  
672 //  
673 //  
674 //  
675 //  
676 //  
677 //  
678 //  
679 //  
680 //  
681 //  
682 //  
683 //  
684 //  
685 //  
686 //  
687 //  
688 //  
689 //  
690 //  
691 //  
692 //  
693 //  
694 //  
695 //  
696 //  
697 //  
698 //  
699 //  
700 //  
701 //  
702 //  
703 //  
704 //  
705 //  
706 //  
707 //  
708 //  
709 //  
710 //  
711 //  
712 //  
713 //  
714 //  
715 //  
716 //  
717 //  
718 //  
719 //  
720 //  
721 //  
722 //  
723 //  
724 //  
725 //  
726 //  
727 //  
728 //  
729 //  
730 //  
731 //  
732 //  
733 //  
734 //  
735 //  
736 //  
737 //  
738 //  
739 //  
740 //  
741 //  
742 //  
743 //  
744 //  
745 //  
746 //  
747 //  
748 //  
749 //  
750 //  
751 //  
752 //  
753 //  
754 //  
755 //  
756 //  
757 //  
758 //  
759 //  
760 //  
761 //  
762 //  
763 //  
764 //  
765 //  
766 //  
767 //  
768 //  
769 //  
770 //  
771 //  
772 //  
773 //  
774 //  
775 //  
776 //  
777 //  
778 //  
779 //  
770 //  
771 //  
772 //  
773 //  
774 //  
775 //  
776 //  
777 //  
778 //  
779 //  
780 //  
781 //  
782 //  
783 //  
784 //  
785 //  
786 //  
787 //  
788 //  
789 //  
790 //  
791 //  
792 //  
793 //  
794 //  
795 //  
796 //  
797 //  
798 //  
799 //  
800 //  
801 //  
802 //  
803 //  
804 //  
805 //  
806 //  
807 //  
808 //  
809 //  
8010 //  
8011 //  
8012 //  
8013 //  
8014 //  
8015 //  
8016 //  
8017 //  
8018 //  
8019 //  
8020 //  
8021 //  
8022 //  
8023 //  
8024 //  
8025 //  
8026 //  
8027 //  
8028 //  
8029 //  
8030 //  
8031 //  
8032 //  
8033 //  
8034 //  
8035 //  
8036 //  
8037 //  
8038 //  
8039 //  
8040 //  
8041 //  
8042 //  
8043 //  
8044 //  
8045 //  
8046 //  
8047 //  
8048 //  
8049 //  
8050 //  
8051 //  
8052 //  
8053 //  
8054 //  
8055 //  
8056 //  
8057 //  
8058 //  
8059 //  
8060 //  
8061 //  
8062 //  
8063 //  
8064 //  
8065 //  
8066 //  
8067 //  
8068 //  
8069 //  
8070 //  
8071 //  
8072 //  
8073 //  
8074 //  
8075 //  
8076 //  
8077 //  
8078 //  
8079 //  
8080 //  
8081 //  
8082 //  
8083 //  
8084 //  
8085 //  
8086 //  
8087 //  
8088 //  
8089 //  
8090 //  
8091 //  
8092 //  
8093 //  
8094 //  
8095 //  
8096 //  
8097 //  
8098 //  
8099 //  
80100 //  
80101 //  
80102 //  
80103 //  
80104 //  
80105 //  
80106 //  
80107 //  
80108 //  
80109 //  
80110 //  
80111 //  
80112 //  
80113 //  
80114 //  
80115 //  
80116 //  
80117 //  
80118 //  
80119 //  
80120 //  
80121 //  
80122 //  
80123 //  
80124 //  
80125 //  
80126 //  
80127 //  
80128 //  
80129 //  
80130 //  
80131 //  
80132 //  
80133 //  
80134 //  
80135 //  
80136 //  
80137 //  
80138 //  
80139 //  
80140 //  
80141 //  
80142 //  
80143 //  
80144 //  
80145 //  
80146 //  
80147 //  
80148 //  
80149 //  
80150 //  
80151 //  
80152 //  
80153 //  
80154 //  
80155 //  
80156 //  
80157 //  
80158 //  
80159 //  
80160 //  
80161 //  
80162 //  
80163 //  
80164 //  
80165 //  
80166 //  
80167 //  
80168 //  
80169 //  
80170 //  
80171 //  
80172 //  
80173 //  
80174 //  
80175 //  
80176 //  
80177 //  
80178 //  
80179 //  
80180 //  
80181 //  
80182 //  
80183 //  
80184 //  
80185 //  
80186 //  
80187 //  
80188 //  
80189 //  
80190 //  
80191 //  
80192 //  
80193 //  
80194 //  
80195 //  
80196 //  
80197 //  
80198 //  
80199 //  
80100 //  
80101 //  
80102 //  
80103 //  
80104 //  
80105 //  
80106 //  
80107 //  
80108 //  
80109 //  
80110 //  
80111 //  
80112 //  
80113 //  
80114 //  
80115 //  
80116 //  
80117 //  
80118 //  
80119 //  
80120 //  
80121 //  
80122 //  
80123 //  
80124 //  
80125 //  
80126 //  
80127 //  
80128 //  
80129 //  
80130 //  
80131 //  
80132 //  
80133 //  
80134 //  
80135 //  
80136 //  
80137 //  
80138 //  
80139 //  
80140 //  
80141 //  
80142 //  
80143 //  
80144 //  
80145 //  
80146 //  
80147 //  
80148 //  
80149 //  
80150 //  
80151 //  
80152 //  
80153 //  
80154 //  
80155 //  
80156 //  
80157 //  
80158 //  
80159 //  
80160 //  
80161 //  
80162 //  
80163 //  
80164 //  
80165 //  
80166 //  
80167 //  
80168 //  
80169 //  
80170 //  
80171 //  
80172 //  
80173 //  
80174 //  
80175 //  
80176 //  
80177 //  
80178 //  
80179 //  
80180 //  
80181 //  
80182 //  
80183 //  
80184 //  
80185 //  
80186 //  
80187 //  
80188 //  
80189 //  
80190 //  
80191 //  
80192 //  
80193 //  
80194 //  
80195 //  
80196 //  
80197 //  
80198 //  
80199 //  
80100 //  
80101 //  
80102 //  
80103 //  
80104 //  
80105 //  
80106 //  
80107 //  
80108 //  
80109 //  
80110 //  
80111 //  
80112 //  
80113 //  
80114 //  
80115 //  
80116 //  
80117 //  
80118 //  
80119 //  
80120 //  
80121 //  
80122 //  
80123 //  
80124 //  
80125 //  
80126 //  
80127 //  
80128 //  
80129 //  
80130 //  
80131 //  
80132 //  
80133 //  
80134 //  
80135 //  
80136 //  
80137 //  
80138 //  
80139 //  
80140 //  
80141 //  
80142 //  
80143 //  
80144 //  
80145 //  
80146 //  
80147 //  
80148 //  
80149 //  
80150 //  
80151 //  
80152 //  
80153 //  
80154 //  
80155 //  
80156 //  
80157 //  
80158 //  
80159 //  
80160 //  
80161 //  
80162 //  
80163 //  
80164 //  
80165 //  
80166 //  
80167 //  
80168 //  
80169 //  
80170 //  
80171 //  
80172 //  
80173 //  
80174 //  
80175 //  
80176 //  
80177 //  
80178 //  
80179 //  
80180 //  
80181 //  
80182 //  
80183 //  
80184 //  
80185 //  
80186 //  
80187 //  
80188 //  
80189 //  
80190 //  
80191 //  
80192 //  
80193 //  
80194 //  
80195 //  
80196 //  
80197 //  
80198 //  
80199 //  
80100 //  
80101 //  
80102 //  
80103 //  
80104 //  
80105 //  
80106 //  
80107 //  
80108 //  
80109 //  
80110 //  
80111 //  
80112 //  
80113 //  
80114 //  
80115 //  
80116 //  
80117 //  
80118 //  
80119 //  
80120 //  
80121 //  
80122 //  
80123 //  
80124 //  
80125 //  
80126 //  
80127 //  
80128 //  
80129 //  
80130 //  
80131 //  
80132 //  
80133 //  
80134 //  
80135 //  
80136 //  
80137 //  
80138 //  
80139 //  
80140 //  
80141 //  
80142 //  
80143 //  
80144 //  
80145 //  
80146 //  
80147 //  
80148 //  
80149 //  
80150 //  
80151 //  
80152 //  
80153 //  
80154 //  
80155 //  
80156 //  
80157 //  
80158 //  
80159 //  
80160 //  
80161 //  
80162 //  
80163 //  
80164 //  
80165 //  
80166 //  
80167 //  
80168 //  
80169 //  
80170 //  
80171 //  
80172 //  
80173 //  
80174 //  
80175 //  
80176 //  
80177 //  
80178 //  
80179 //  
80180 //  
80181 //  
80182 //  
80183 //  
80184 //  
80185 //  
80186 //  
80187 //  
80188 //  
80189 //  
80190 //  
80191 //  
80192 //  
80193 //  
80194 //  
80195 //  
80196 //  
80197 //  
80198 //  
80199 //  
80100 //  
80101 //  
80102 //  
80103 //  
80104 //  
80105 //  
80106 //  
80107 //  
80108 //  
80109 //  
80110 //  
80111 //  
80112 //  
80113 //  
80114 //  
80115 //  
80116 //  
80117 //  
80118 //  
80119 //  
80120 //  
80121 //  
80122 //  
80123 //  
80124 //  
80125 //  
80126 //  
80127 //  
80128 //  
80129 //  
80130 //  
80131 //  
80132 //  
80133 //  
80134 //  
80135 //  
80136 //  
80137 //  
80138 //  
80139 //  
80140 //  
80141 //  
80142 //  
80143 //  
80144 //  
80145 //  
80146 //  
80147 //  
80148 //  
80149 //  
80150 //  
80151 //  
80152 //  
80153 //  
80154 //  
80155 //  
80156 //  
80157 //  
80158 //  
80159 //  
80160 //  
80161 //  
80162 //  
80163 //  
80164 //  
80165 //  
80166 //<br
```



- **firestore.collection("users")** merujuk ke collection users pada Firestore yang akan berisi data-data nama dan umur. Jika belum ada maka otomatis dibuat saat menambahkan data.

```
FirebaseFirestore firestore = FirebaseFirestore.instance;
CollectionReference dataDiri = firestore.collection("data_diri");
```

```
File Edit Selection View Go Run Terminal Help
lib > main.dart MyHomePage.dart 1 > _MyHomePageState > build
  void initState() {
    super.initState();
    _ctrlNama.addListener(_printLatestNama);
    _ctrlUsia.addListener(_printLatestUsia);
  }

  void _printLatestNama() {
    print('Nama : ${_ctrlNama.text}');
  }

  void _printLatestUsia() {
    print('Usia : ${_ctrlUsia.text}');
  }

  @override
  Widget build(BuildContext context) {
    ...
    FirebaseFirestore firestore = FirebaseFirestore.instance;
    CollectionReference dataDiri = firestore.collection("data_diri");

    return SafeArea(
      child: Scaffold(
        appBar: AppBar(
          title: Text("CRUD Firebase"),
          backgroundColor: Colors.deepPurple,
          actions: [
            TextButton(
              onPressed: () {
                Navigator.push(
                  context,
                  MaterialPageRoute(

```

Cari widget **ElevatedButton** untuk Add Data, tambahkan perintah untuk menambah data ke colecion users dalam bentuk object berisi key-value name dan age, serta mengosongkan input dari TextField. Jika



input umur gagal diubah dalam bentuk integer (menjadi null) maka key age diisi value -1.

```
dataDiri.add({  
    'nama' : _ctrlNama.text,  
    'usia' : int.tryParse(_ctrlUsia.text) ?? -1,  
});  
_ctrlNama.text = "";  
_ctrlUsia.text = "";
```

```
84     child: Padding(  
85         padding: const EdgeInsets.all(20),  
86         child: ListView(  
87             children: [  
88                 TextField(  
89                     decoration: InputDecoration(  
90                         border: OutlineInputBorder(),  
91                         labelText: 'Nama',  
92                     ), // InputDecoration  
93                     controller: _ctrlNama,  
94                 ), // TextField  
95                 SizedBox(height: 10),  
96                 TextField(  
97                     decoration: InputDecoration(  
98                         border: OutlineInputBorder(),  
99                         labelText: 'Usia',  
100                    ), // InputDecoration  
101                    controller: _ctrlUsia,  
102                ), // TextField  
103                SizedBox(height: 10),  
104                ElevatedButton(  
105                    onPressed: () {  
106                        dataDiri.add(  
107                            'nama' : _ctrlNama.text,  
108                            'usia' : int.tryParse(_ctrlUsia.text) ?? -1,  
109                        );  
110                        _ctrlNama.text = "";  
111                        _ctrlUsia.text = "";  
112  
113                        ScaffoldMessenger.of(context).showSnackBar(notifTambah);  
114                    },  
115                    child: Text("Submit"),  
116                ), // ElevatedButton  
117            ],  
118        ), // ListView  
119    ), // Padding  
120    ), // Center  
121    ), // Scaffold  
122 ); // SafeArea  
123 }  
124 }  
125 }
```

## b. Read Data

### i. Menampilkan List Data



Untuk menampilkan data dari Firestore, terdapat dua cara yang dapat dilakukan. Cara pertama adalah mengambil kumpulan data **SEKALI** menggunakan widget **FutureBuilder()** di dalam children dari **ListView()**. Hal-hal yang perlu diperhatikan adalah:

- Parameter future diisi dengan method future yang akan dijalankan dan kali ini diisi dengan reference dari collection users.
- Karena **users.get()** mengembalikan **Future<QuerySnapshots>** maka **FutureBuilder** nantinya akan berisi **QuerySnapshots**.
- Method builder akan membuat widget berdasarkan status dari future tadi. Developer dapat menentukan widget yang akan dikembalikan jika proses future telah beres maupun belum.

Dalam hal ini, jika snapshot belum memiliki data atau masih dalam proses mengambil data, maka akan menampilkan **Text('Loading...')**. Jika snapshot memiliki data, maka aplikasi akan menampilkan list data dari collection **users**.

```
FutureBuilder<QuerySnapshot>(  
    future: dataDiri.get(),  
    builder: (_, snapshot) {  
        return (snapshot.hasData)  
            ? Column(  
                children: snapshot.data!.docs  
                    .map(  
                        (e) => dataTile(  
                            e.get('nama'), e.get('usia').toString()),  
                    )  
                    .toList(),  
            )  
            : Text('Loading...');  
    },  
,
```



```
lib > LihatData.dart > LihatDataState > build
128   ),
129   ); // IconButton
130 );
131
132 @override
133 Widget build(BuildContext context) {
134   FirebaseFirestore firestore = FirebaseFirestore.instance;
135   CollectionReference dataDiri = firestore.collection("data_diri");
136
137   return SafeArea(
138     child: Scaffold(
139       appBar: AppBar(
140         title: Text("Halaman Tampil Data"),
141         backgroundColor: Colors.deepPurple,
142       ), // AppBar
143       body: Center(
144         child: Padding(
145           padding: const EdgeInsets.all(20),
146           child: FutureBuilder<QuerySnapshot>(
147             future: dataDiri.get(),
148             builder: (_, snapshot) {
149               if (snapshot.hasData)
150                 return Column(
151                   children: snapshot.data!.docs
152                     .map(
153                       (e) => dataTile(
154                         e.get('nama'), e.get('usia').toString(),
155                         )
156                     .toList(),
157                   );
158                 );
159               else
160                 return Text('Loading...');
161             },
162           ), // FutureBuilder
163           ), // Padding
164           ), // Center
165         ); // Scaffold
166       ); // SafeArea
167     }
168   )
169 }
```

Cara kedua untuk menampilkan data dari Firestore adalah menggunakan widget **StreamBuilder()** dimana data ditampilkan secara real time selama terkoneksi dengan internet.

Hal-hal yang perlu diperhatikan adalah:

- **StreamBuilder()** menggunakan parameter stream daripada menggunakan future, dengan value berupa snapshot dari reference dari collection.

```
StreamBuilder<QuerySnapshot>(
  stream: dataDiri.snapshots(),
  builder: (_, snapshot) {
    return (snapshot.hasData)
      ? Column(
        children: snapshot.data!.docs
          .map(
            (e) => dataTile(
```



```
        e.get('nama'), e.get('usia').toString()),
    )
    .toList(),
)
: Text('Loading...');
},
),
```

```
File Edit Selection View Go Run Terminal Help
main.dart MyHomePage.dart LihatData.dart
lib > LihatData.dart > _LihatDataState > build
128     ), // IconButton
129     ); // ListTile
130   }
131
132   @override
133   Widget build(BuildContext context) {
134     FirebaseFirestore firestore = FirebaseFirestore.instance;
135     CollectionReference dataDiri = firestore.collection("data_diri");
136
137     return SafeArea(
138       child: Scaffold(
139         appBar: AppBar(
140           title: Text("Halaman Tampil Data"),
141           backgroundColor: Colors.deepPurple,
142         ), // AppBar
143         body: Center(
144           child: Padding(
145             padding: const EdgeInsets.all(20),
146             child: StreamBuilder<QuerySnapshot>(
147               stream: dataDiri.snapshots(),
148               builder: (_, snapshot) {
149                 return (snapshot.hasData)
150                   ? Column(
151                     children: snapshot.data!.docs
152                     .map(
153                       (e) => ListTile(
154                         e.get('nama'), e.get('usia').toString(),
155                         )
156                         .toList(),
157                     ) // Column
158                     : Text('Loading...');
159                   },
160                 ), // StreamBuilder
161               ), // Padding
162             ), // Center
163           ), // Scaffold
164         ); // SafeArea
165       }
166     }
167
```

## ii. Mengambil Satu Data

Mengambil satu data dari collection, digunakan snapshot yang diambil dari stream berdasarkan ID document nya. Stream akan



mengembalikan **DocumentSnapshot**, maka **StreamBuilder** juga bertipe **DocumentSnapshot**.

```
StreamBuilder<DocumentSnapshot>(
    stream:
    dataDiri.doc('bN8Av9mge1DDKCwdZ9QX').snapshots(),
    builder: (_, snapshot) {
        return (snapshot.hasData)
            ? dataTile(
                snapshot.data!.get('nama'),
                snapshot.data!.get('usia').toString(),
            )
            : Text('Loading...');
    },
),
```

The screenshot shows a code editor with a dark theme. The file being edited is `LihatData.dart`. The code is identical to the one shown in the previous slide, demonstrating the use of `StreamBuilder` to listen to a Firestore collection and display its data as `DocumentSnapshot` objects.

```
File Edit Selection View Go Run Terminal Help
lib > LihatData.dart > _LihatDataState > build
130 }
131
132 @override
133 Widget build(BuildContext context) {
134     FirebaseFirestore firestore = FirebaseFirestore.instance;
135     CollectionReference dataDiri = firestore.collection("data_diri");
136
137     return SafeArea(
138         child: Scaffold(
139             appBar: AppBar(
140                 title: Text("Halaman Tampil Data"),
141                 backgroundColor: Colors.deepPurple,
142             ), // AppBar
143             body: Center(
144                 child: Padding(
145                     padding: const EdgeInsets.all(20),
146                     child: StreamBuilder<DocumentSnapshot>(
147                         stream: dataDiri.doc('bN8Av9mge1DDKCwdZ9QX').snapshots(),
148                         builder: (_, snapshot) {
149                             return (snapshot.hasData)
150                                 ? dataTile(
151                                     snapshot.data!.get('nama'),
152                                     snapshot.data!.get('usia').toString(),
153                                 )
154                                 : Text('Loading...');
155                         },
156                     ), // StreamBuilder
157                     ), // Padding
158                     ), // Center
159                     ), // Scaffold
160                 ); // SafeArea
161             }
162 }
```

### iii. Sortir dan Filter List Data



List data dari Firestore dapat disortir atau difilter sebelum ditampilkan pada aplikasi, agar dapat sortir list data berdasarkan key data pada documents, maka value dari stream dapat diganti seperti dibawah ini.

- ❖ Sortir list data secara descending, maka tambahkan parameter descending pada orderBy.

```
StreamBuilder<QuerySnapshot>(  
    stream: dataDiri.orderBy('nama', descending:  
true).snapshots(),  
    builder: (_, snapshot) {  
        return (snapshot.hasData)  
            ? Column(  
                children: snapshot.data!.docs  
                    .map(  
                        (e) => dataTile(  
                            e.get('nama'), e.get('usia').toString()),  
                        )  
                    .toList(),  
                )  
            : Text('Loading...');  
    },  
,
```

- ❖ Filter list data maka tambahkan fungsi **where()** pada snapshot yang dipanggil.

```
StreamBuilder<QuerySnapshot>(  
    stream: dataDiri.where('usia', isGreaterThan:  
24).snapshots(),  
    builder: (_, snapshot) {  
        return (snapshot.hasData)  
            ? Column(  
                children: snapshot.data!.docs
```



```
.map(  
    (e) => dataTile(  
        e.get('nama'), e.get('usia').toString()),  
    )  
    .toList(),  
)  
: Text('Loading...');  
},  
,
```

### c. Update Data

Pada proses update ambil satu document berdasarkan ID nya, lalu panggil fungsi update dengan parameter object dari data yang baru.

```
dataDiri.doc(id).update({usia: 15});
```

### Penerapan pada Projek

Ubahlah source code pada **LihatData.dart** menjadi seperti berikut (**Perhatikan Source Code yang Bold**) :

```
StreamBuilder<QuerySnapshot>(  
    stream: dataDiri.snapshots(),  
    builder: (_, snapshot) {  
        return (snapshot.hasData)  
            ? Column(  
                children: snapshot.data!.docs  
                    .map(  
                        (e) => dataTile(  
                            e.get('nama'), e.get('usia').toString(),e.id),  
                        )  
                    .toList(),  
            )  
            : Text('Loading...');  
    },
```



```
 ),
```

```
ListTile dataTile(String nama, String usia , String id) {  
    return ListTile(  
        title: Text("Nama : " + nama),  
        subtitle: Text("Usia : " + usia + " tahun"),  
        onTap: () {  
            editData(context,nama, usia, id);  
        },  
        tileColor: Color.fromARGB(255, 222, 193, 238),  
        shape: RoundedRectangleBorder(  
            borderRadius: BorderRadius.circular(10),  
        ),  
        leading: IconButton(  
            onPressed: () {  
                hapusData(context);  
            },  
            icon: Icon(Icons.delete),  
        ),  
    );  
}
```

```
Future<dynamic> editData(BuildContext context,String nama, String  
usia, String id) {  
    FirebaseFirestore firestore = FirebaseFirestore.instance;  
    CollectionReference dataDiri = firestore.collection("data_diri");  
    _ctrlNama.text = nama;  
    _ctrlUsia.text = usia;  
    return showDialog(  
        context: context,  
        builder: (context) {  
            return AlertDialog(  
                title: Text("Edit Data"),  
                content: Container(  
                    height: 140,
```



```
child: Column(  
    children: [  
        TextField(  
            decoration: InputDecoration(  
                border: OutlineInputBorder(),  
                labelText: 'Nama',  
            ),  
            controller: _ctrlNama,  
        ),  
        SizedBox(height: 10),  
        TextField(  
            decoration: InputDecoration(  
                border: OutlineInputBorder(),  
                labelText: 'Usia',  
            ),  
            controller: _ctrlUsia,  
        ),  
    ],  
,  
),  
actions: [  
    TextButton(  
        onPressed: () {  
            Navigator.of(context).pop();  
        },  
        child: Text("Cancel"),  
    ),  
    TextButton(  
        onPressed: () {  
            dataDiri.doc(id).update({  
                "nama" : _ctrlNama.text.toString(),  
                "usia" : int.tryParse(_ctrlUsia.text),  
            });  
            Navigator.of(context).pop();  
            ScaffoldMessenger.of(context).showSnackBar(notifEdit);  
        },  
        child: Text("Save"),  
    ),  
],  
);  
,
```



```
 );  
 }
```

#### d. Delete Data

Pada proses delete ambil satu document berdasarkan ID nya, lalu panggil fungsi delete.

```
users.doc(id).delete();
```

#### Penerapan Pada Projek

Ubahlah source code pada **LihatData.dart** menjadi seperti berikut (**Perhatikan Source Code yang Bold**) :

```
ListTile dataTile(String nama, String usia , String id) {  
    return ListTile(  
        title: Text("Nama : " + nama),  
        subtitle: Text("Usia : " + usia + " tahun"),  
        onTap: () {  
            editData(context,nama, usia, id);  
        },  
        tileColor: Color.fromRGBO(255, 222, 193, 238),  
        shape: RoundedRectangleBorder(  
            borderRadius: BorderRadius.circular(10),  
        ),  
        leading: IconButton(  
            onPressed: () {  
                hapusData(context, id);  
            },  
            icon: Icon(Icons.delete),  
        ),  
    );  
}
```



```
Future<dynamic> hapusData(BuildContext context, String id) {
    FirebaseFirestore firestore = FirebaseFirestore.instance;
    CollectionReference dataDiri = firestore.collection("data_diri");
    return showDialog(
        context: context,
        builder: (context) {
            return AlertDialog(
                title: Text("Hapus Data"),
                content: Text("Anda yakin ingin menghapus data ini ?"),
                actions: [
                    TextButton(
                        onPressed: () {
                            Navigator.of(context).pop();
                        },
                        child: Text("Cancel"),
                    ),
                    TextButton(
                        onPressed: () {
                            dataDiri.doc(id).delete();
                            Navigator.of(context).pop();
                            ScaffoldMessenger.of(context).showSnackBar(notifHapus);
                        },
                        child: Text("Yes"),
                    ),
                ],
            );
        });
}
```