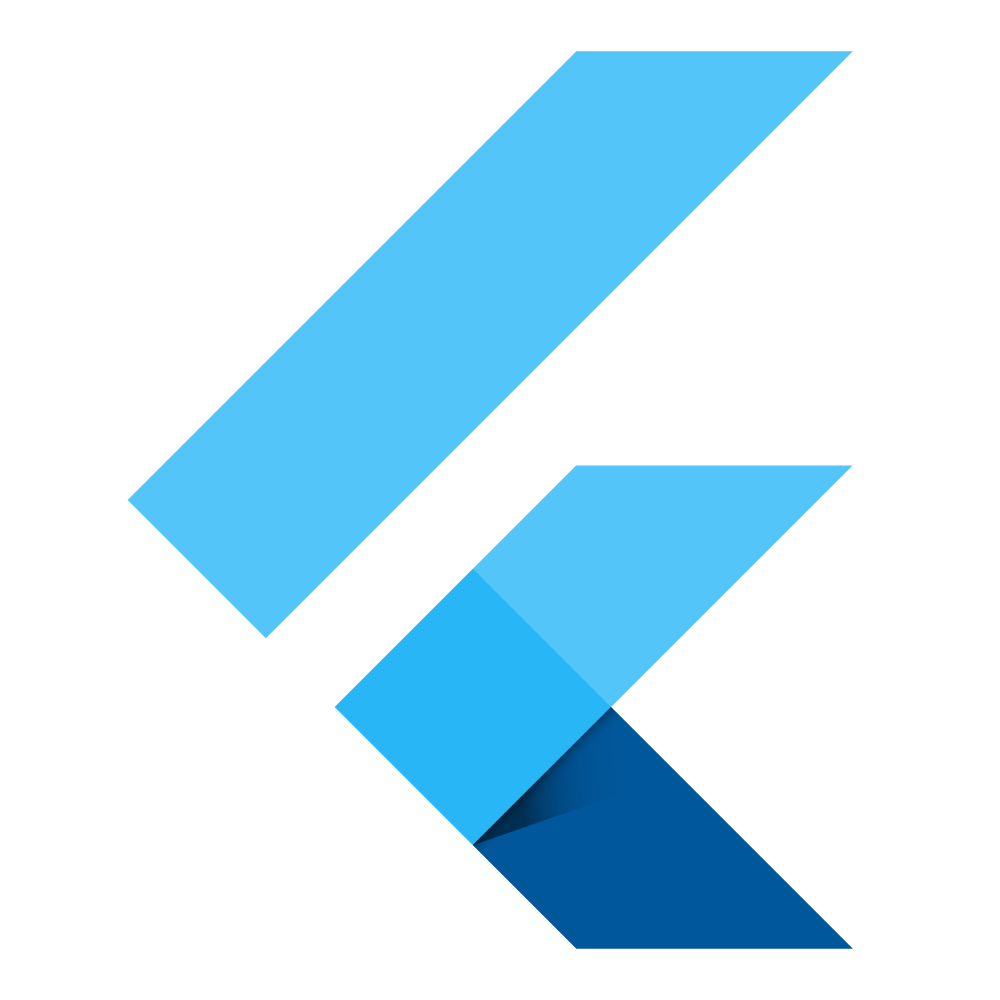


**FLUTTER BOOTCAMP**

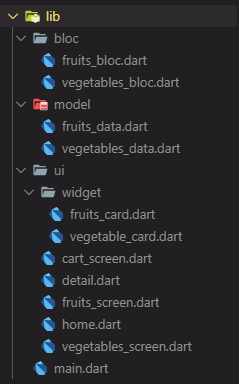
“FRESH CART APP”



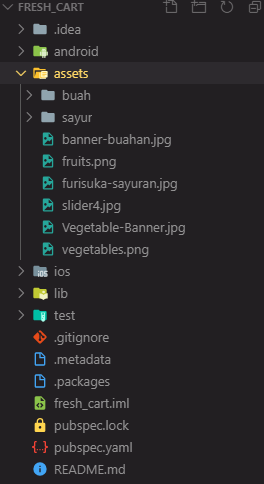
Writer and Development By :

***Muhammad Isrim YB***

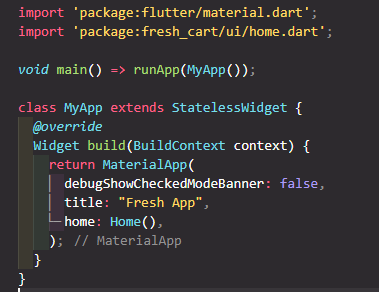
1. **Membuat Project Baru**
2. Buka Command Prompt ( CMD ) dan Arahkan ke direktori yang anda inginkan.
3. Tuliskan perintah **flutter create fresh\_app** dan tekan **enter**
4. Tunggu hingga proses selesai.
5. **Setting Struktur Project**
6. Setelah Membuat Project, kemudian masuk ke direktori project tersebut dengan menulis perintah **cd fresh\_app** dan tekan tombol **enter**
7. Buka Project ke Editor Visual Studio Code ( Recommended ) dengan menuliskan perintah **code .**
8. Tambahkan folder dan file sehingga **lib** menjadi seperti ini :



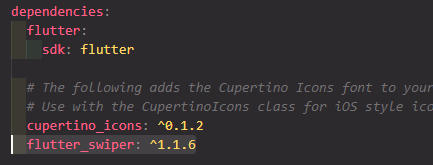
1. Setelah itu, download assets image yang sudah disediakan [disini](https://drive.google.com/open?id=1Hi96Tkz6qfKG1OHjwcMCIRAJJX9wwTvT) dengan klik kanan folder **assets** dan klik **download**
2. extract file yang telah di download, copy folder **assets** dan paste ke folder project **fresh\_cart** sehingga strukture project seperti ini :



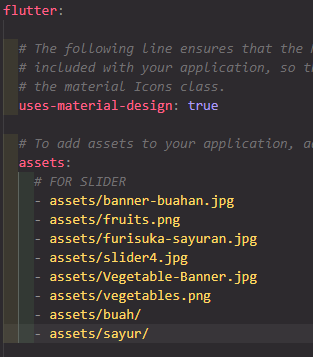
1. **Membuat Tampilan Home Screen**
2. Buka file **main.dart** di folder **lib** ubah codenya menjadi seperti berikut :



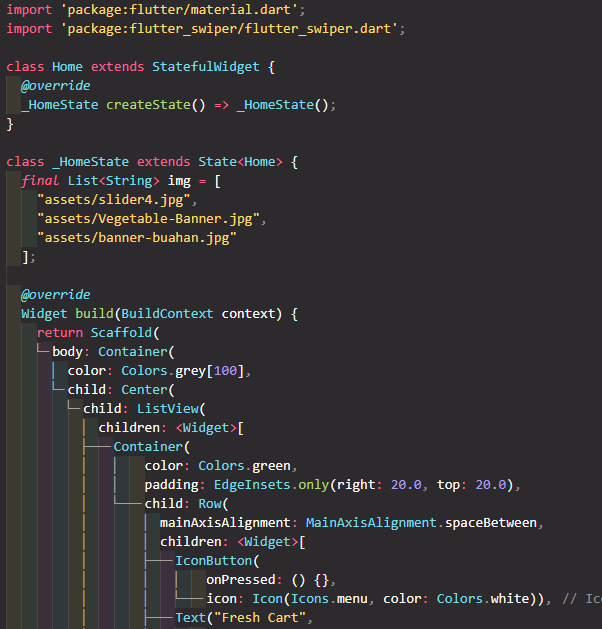
1. Buka file **pubspec.yaml** dan tambahkan dependencies **flutter\_swiper** :

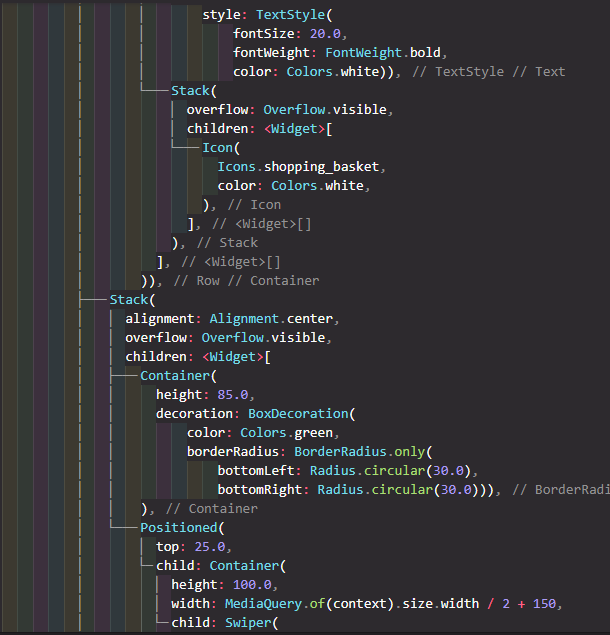
****

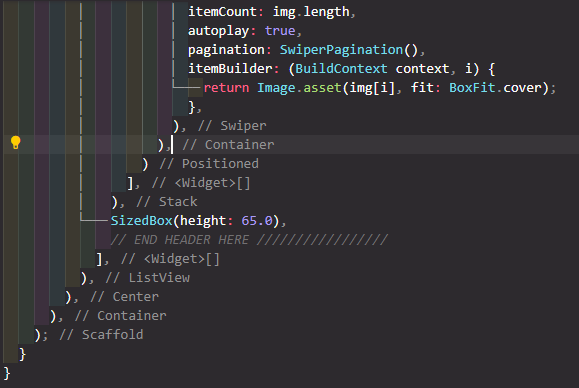
1. Buka file **pubspec.yaml** tambahkan assets gambar sebagai berikut :



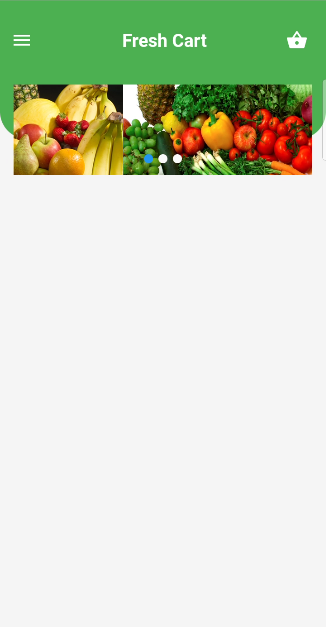
1. Buka file **home.dart** pada folder **ui** dan tuliskan codenya seperti berikut :



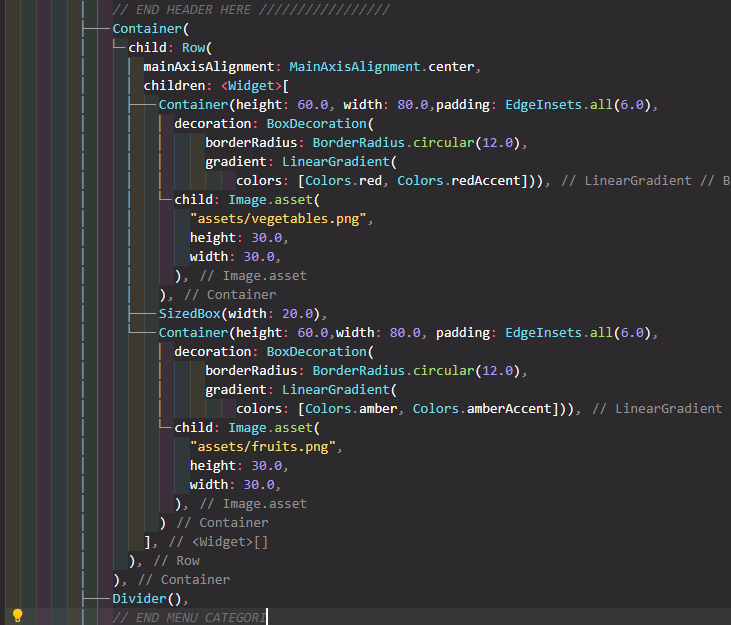




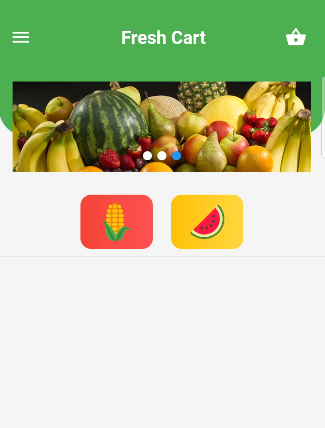
1. jalankan aplikasinya dengan cara menuliskan perintah **flutter run** pada terminal atau tekan **F5** pada keyboard, pastikan devices anda telah terhubung, dan hasil sementara seperti ini :



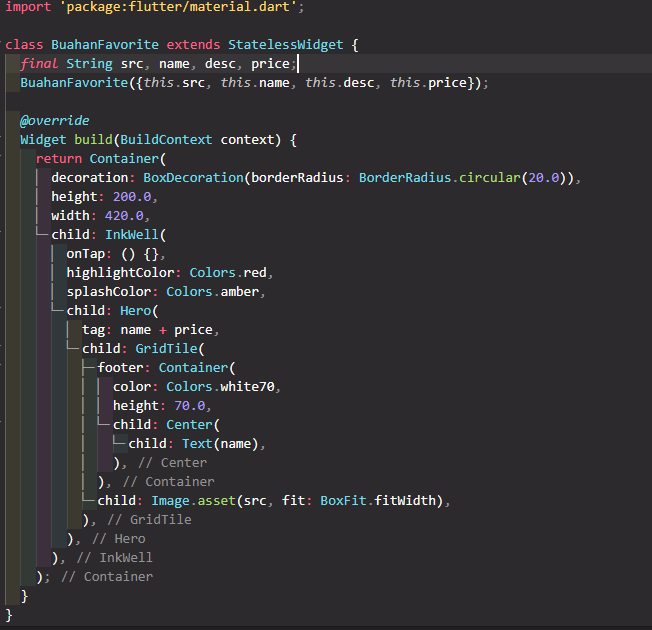
1. kembali lagi ke editor, buka lagi file **home.dart** pada folder **ui** dan tambahkan kode berikut di bawah widget **SizedBox()** tepatnya dibawah komentar **// END HEADER HERE ////////**

****

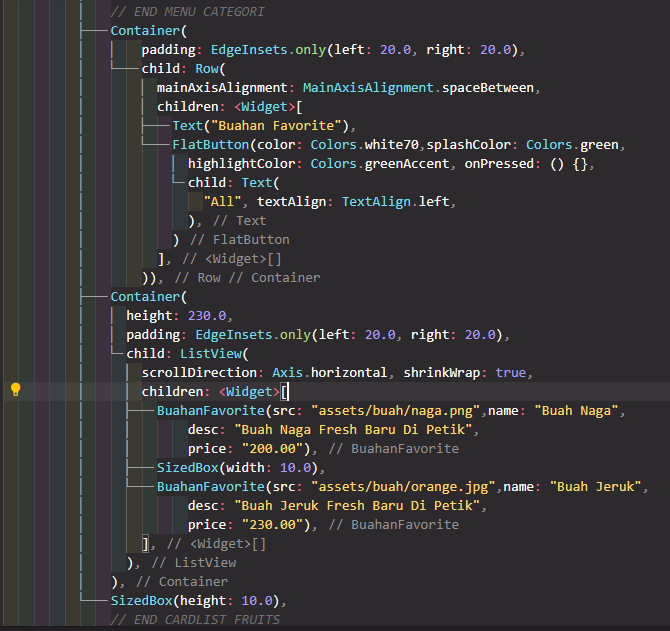
1. Jalankan kembali programmnya atau reload dengan menekan tombol **r** pada terminal dan hasil akan seperti berikut, terdapat 2 menu kartegori :



1. Kembali lagi ke editor, buka file **fruits\_card.dart** pada folder widget dan tambahkan codenya sebagai berikut :

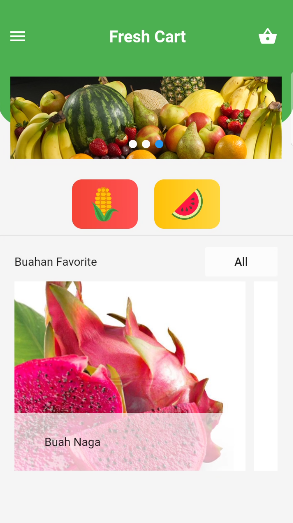


1. buka file **home.dart** tambahkan code tepat dibawah widget **Divider()** dan komentar **//END MENU CATEGORI //////////////**

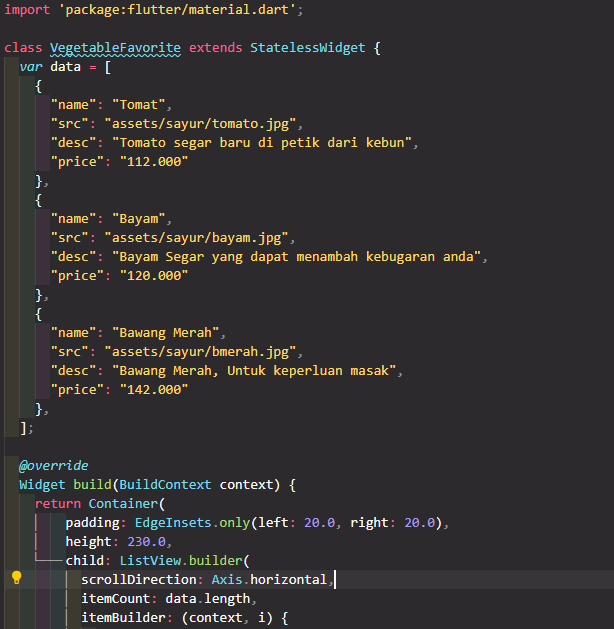


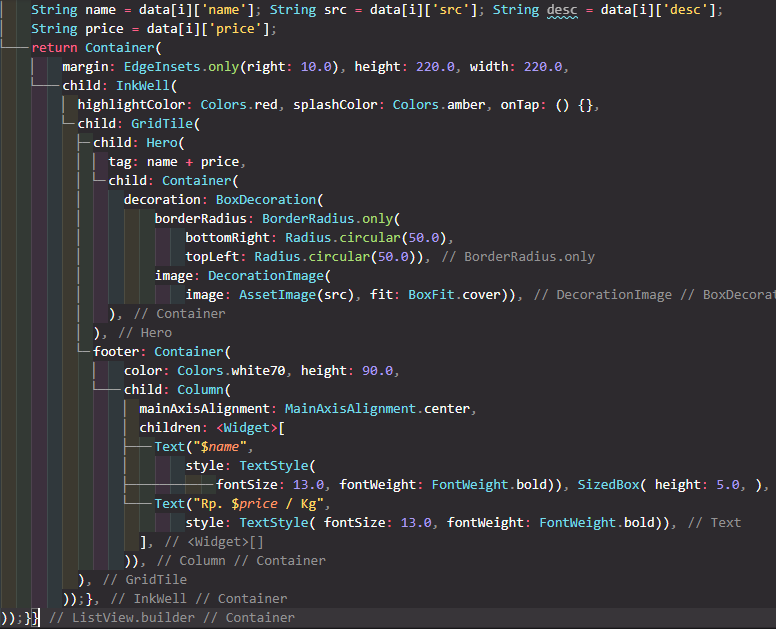
Apabila ada **error** pada **BuahanFavorite** klik icon lampu dan pilih **import 'package:fresh\_cart/ui/widget/fruits\_card.dart';**

1. Running Aplikasinya atau tekan **r** pada terminal dan hasil seperti berikut :

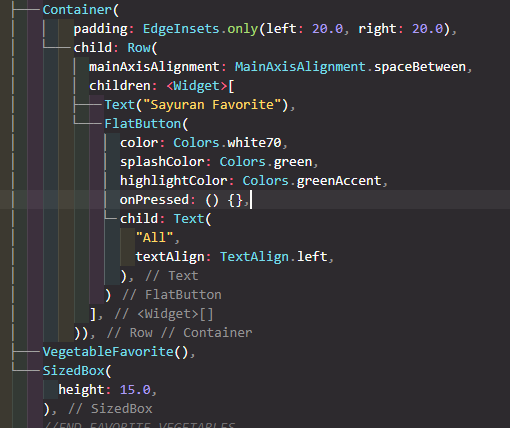


1. Kembali lagi ke editor, buka file **vegetables\_card.dart** pada folder **widget** dan tambahkan codenya sebagai berikut :





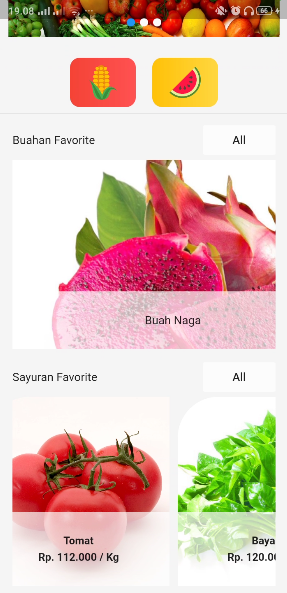
1. Kembali Ke **home.dart**, tambahkan code berikut untuk membuat list dari sayuran, tepatnya di bawah widget **SizedBox()** dan dibawah komentar /**/END CARDLIST FRUITS**

****

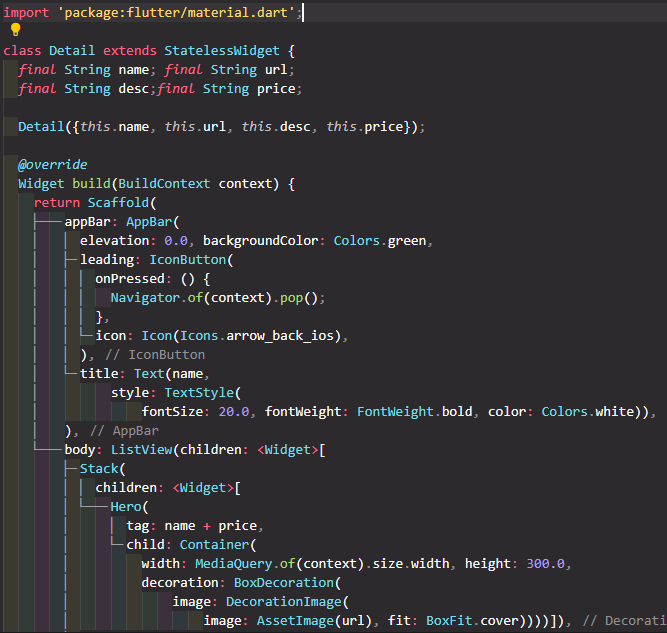
apabila terjadi error pada **VegetableFavorite()** klik icon lampu pada samping kiri pesan error dan pilih **import**

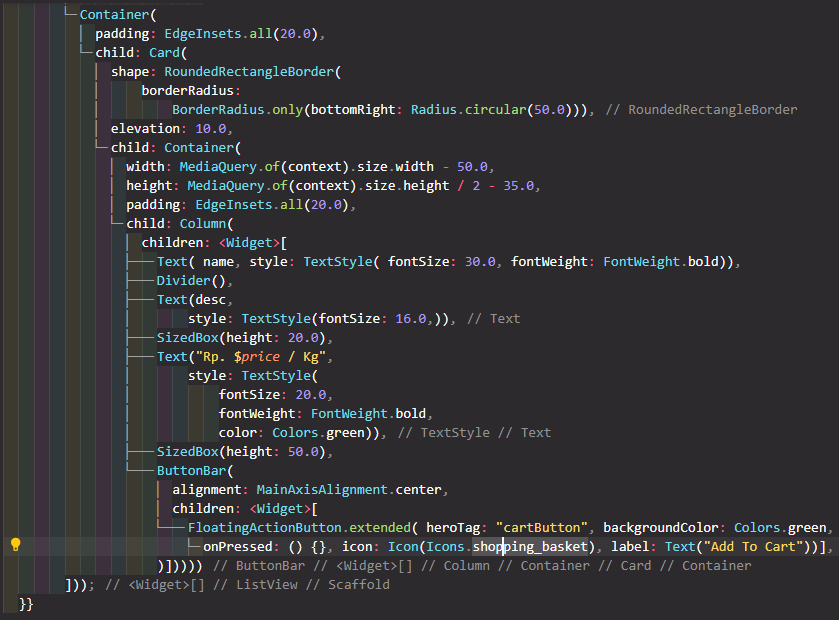
**'package:fresh\_cart/ui/widget/vegetable\_card.dart';**

1. Jalankan atau reload aplikasinya, terdapat list sayuran secara horizontal dan hasilnya akan seperti ini :

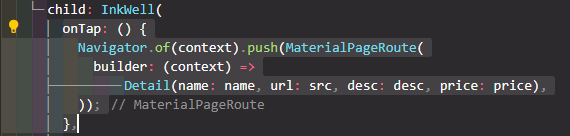


1. **Membuat Tampilan Detail**
2. Buka file **detail.dart** pada folder **ui,** tuliskan kode berikut :

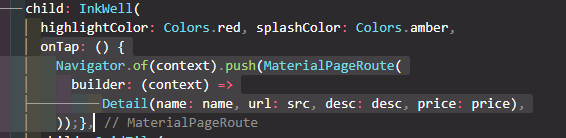




1. Untuk dapat menampilkan halaman detail, kita perlu menambahkan navigator dari item card fruits dan vegetables, untuk menambahkan navigator di fruits item, tambahkan kode berikut pada file **fruits\_card.dart** tepatnya pada widget **ontap()** yang ada di dalam widget **InkWell**



dan juga pada file **vegetables\_card.dart** sebagai berikut



apabila error pada pemanggilan class Detail, klik Detail dan akan muncul icon lampu, klik icon lampu tersebut dan pilih

**import ‘package:fresh\_cart/ui/detail.dart’**

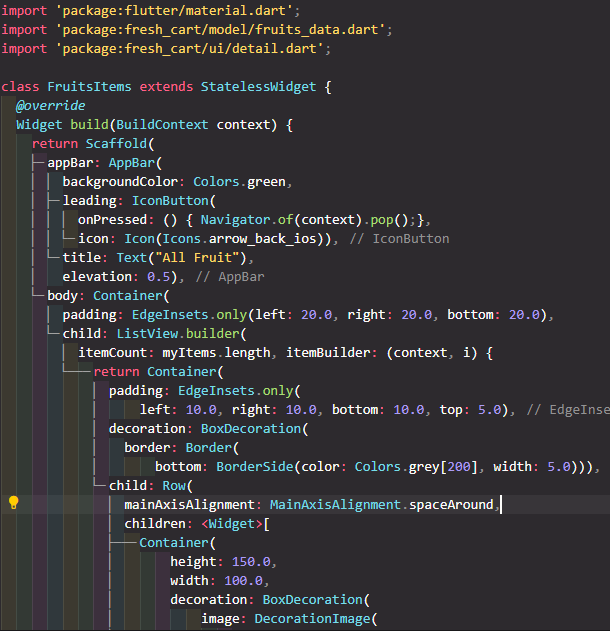
1. Jika sudah, save dan running atau reload programnya, klik salah satu item pada list favorite buahan atau sayuran dan hasilnya akan seperti dibawah ini

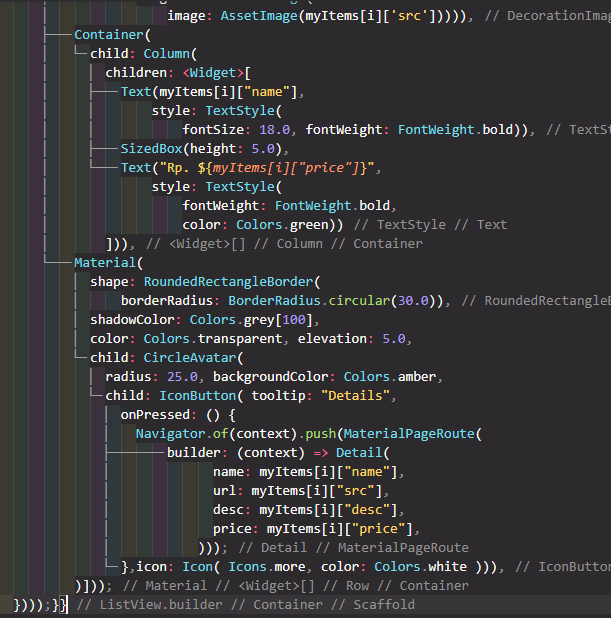


1. **Membuat Tampilan List Fruits Screen**
2. Buka file **fruits\_data.dart** pada folder **model** dan tuliskan codenya sebagai berikut

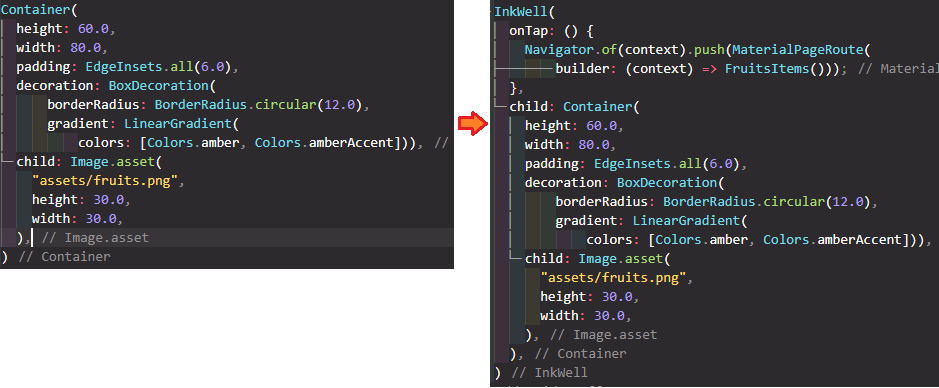


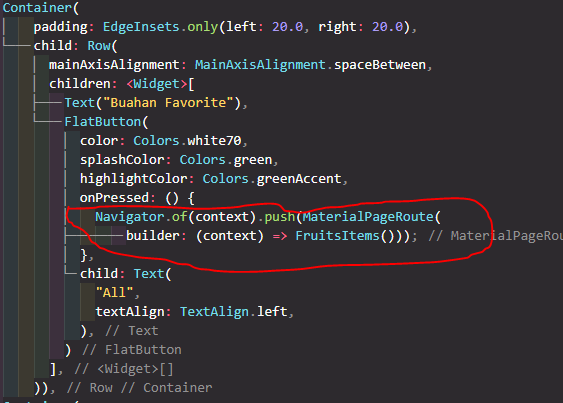
1. Tambahkan juga code berikut di file **fruits\_screen.dart** pada folder **ui**





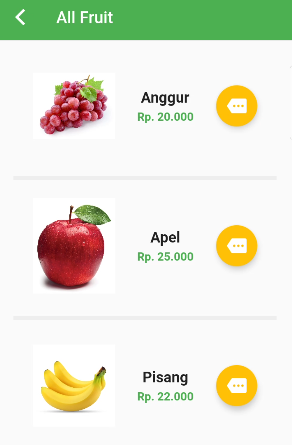
1. Tambahkan Widget **InkWell** dan Navigator pada Menu Categori dan Tombol All di file **home.dart** pada folder **ui**

****

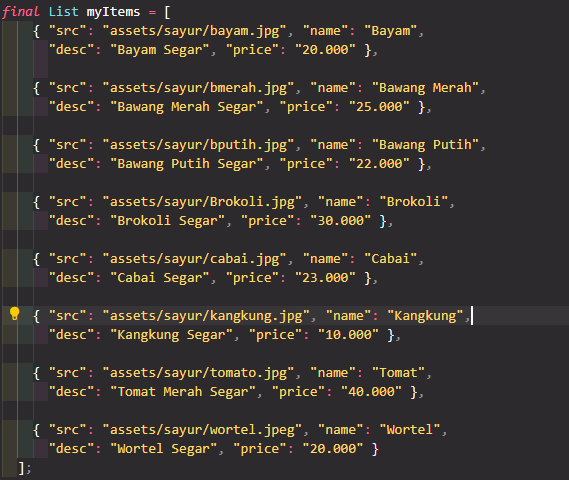


apabila mendapatkan error pada **FruitsItems()**, klik icon lampu di samping kiri dan pilih **import ‘package:/fresh\_cart/ui/fruits\_screen.dart’**

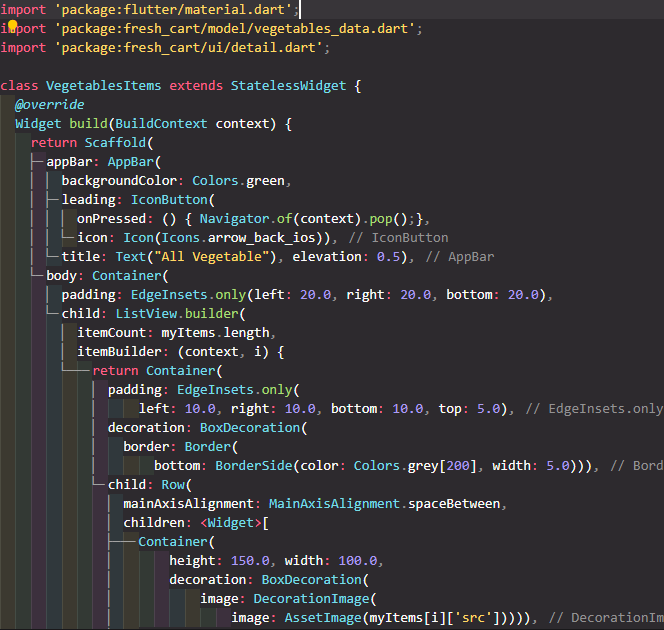
1. Save, Reload Aplikasinya, dan ketika anda menekan icon buah atau menekan tombol All pada Buahan favorite, maka akan tampil semua list dari buah, seperti berikut :



1. **Membuat Tampilan List Vegetables Screen**
2. Buka file **vegetables\_data.dart** pada folder **model** dan tambahkan kodenya sebagai berikut

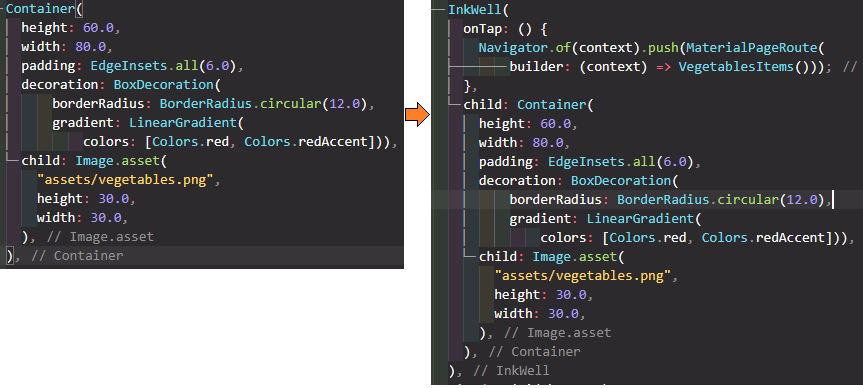


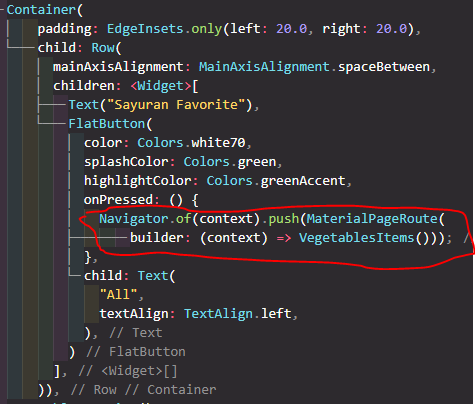
1. Buka File **vegetables\_screen.dart** pada folder **ui** dan tambahkan kode berikut





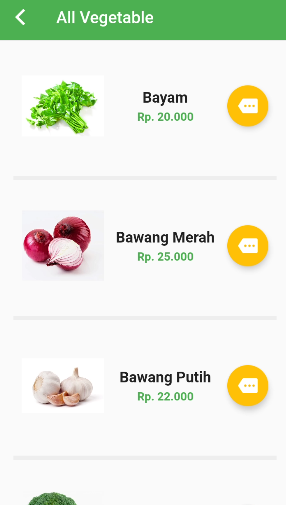
1. Buka file **home.dart** untuk menambahkan widget **inkWell()** pada widget Container dan Navigator() pada menu All





apabila mendapatkan error pada **VegetablesItems()**, klik icon lampu di samping kiri dan pilih **import ‘package:/fresh\_cart/ui/fruits\_screen.dart’**

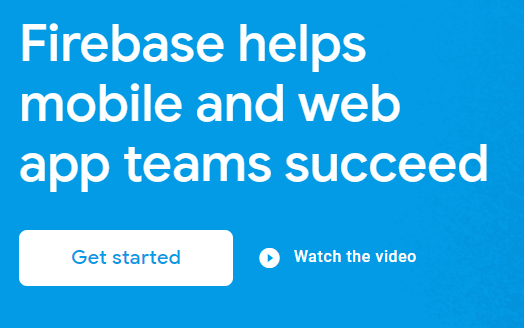
1. Save, Reload Aplikasinya, dan ketika anda menekan icon Sayur atau menekan tombol All pada Sayuran favorite, maka akan tampil semua list dari sayur, seperti berikut :



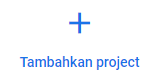
1. **Membuat Akun Firebase**
2. Buka web firebase [disini](https://firebase.google.com)
3. klik menu **sign in** di pojok kanan atas



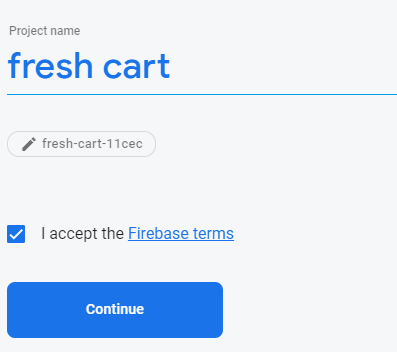
1. login menggunakan akun **gmail** anda
2. Selamat akun **Firebase** anda berhasil dibuat
3. **Konfigurasi Firebase di Project Flutter**
4. Setelah Berhasil login menggunakan akun gmail
5. klik menu **Get Started** pada halaman depan webnya



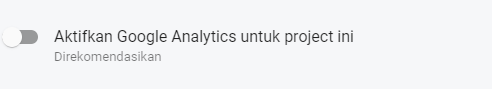
1. klik menu **tambahkan project**

****

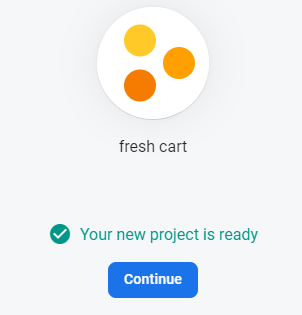
1. isi nama projectnya dan klik tombol **lanjutkan** atau **continue**

****

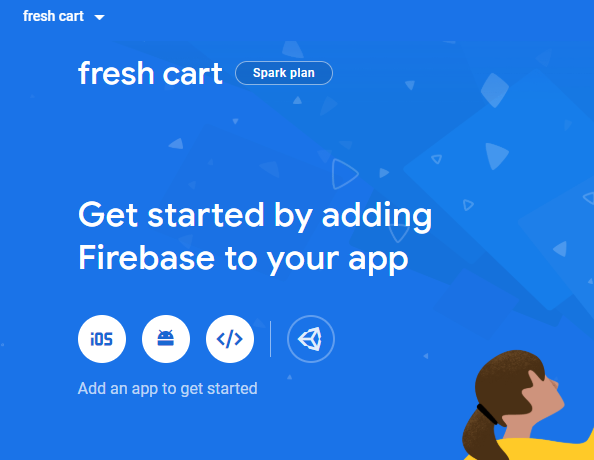
1. Matikan Konfigurasi **google Analytics** karena kita membutuhkannya di project ini



1. Klik Button **Buat Project** dan tunggu proses hingga selesai
2. Setelah proses selesai klik **lanjutkan** atau **continue**

****

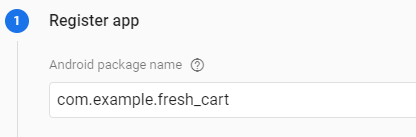
1. Setelah itu akan muncul tampilan overview seperti ini



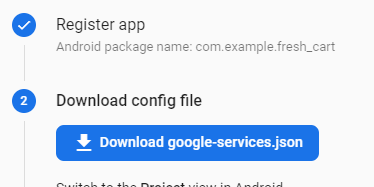
1. klik icon **android **
2. kembali ke editor, buka file **AndroidManifest.xml** pada direktori **android/app/src/main** dan copy isi packagenya



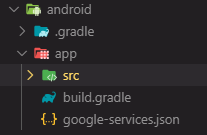
1. kembali ke project firebase dan paste atau tuliskan isi package tersebut pada android package name, seperti ini



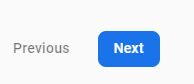
1. klik **register app**
2. klik download **google-service.json**

****

1. copy file yang telah di download dan pastekan pada direktori **android/app**

****

1. Kembali ke web firebase dan klik **next**

****

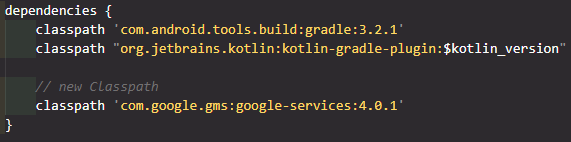
1. setelah itu akan muncul tampilan seperti ini

****

1. copy classpath pada dependencies dengan cara mengklik icon salin di samping kanan code



1. kembali ke editor, dan pastekan pada file **build.gradle** pada direktori **android** ganti versi menjadi **4.0.1**

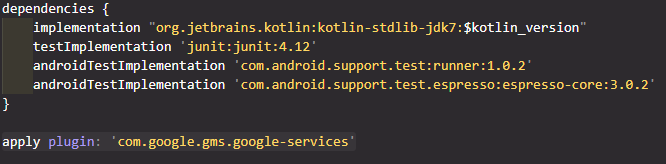
****

kenapa kita mengganti ke versi yang lebih lama, karena package cloud\_firestore pada flutter belum support dengan google services yang terbaru

1. Setelah itu, kembali ke web firebase dan copy code pada apply plugin service berikut



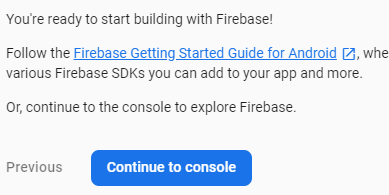
1. kembali ke editor, Pastekan pada pada file **build.gradle** pada direktori **android** paling bawah



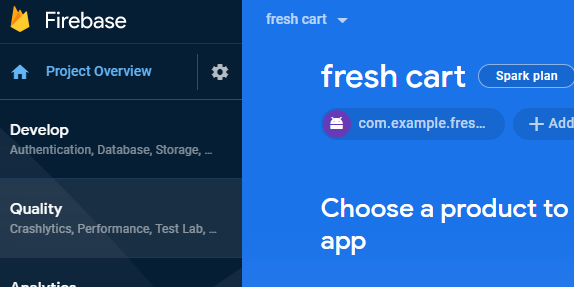
1. Setelah itu kembali ke web firebase console dan klik tombol **next**

****

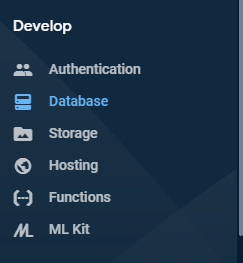
1. dan terakhir, klik **continue to console**



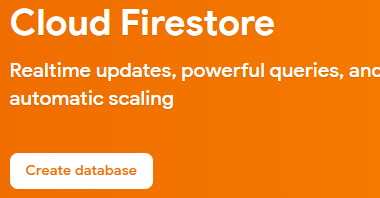
1. Selamat anda telah dapat membuat project dengan firebase, selanjutnya kita akan membuat database pada console firebase.



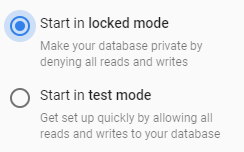
1. Untuk membuat database pada firebase, klik menu **Develop** dan klik pilihan **Database**

****

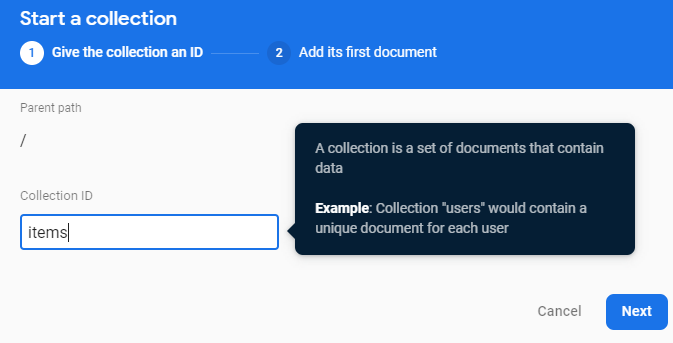
1. Kita akan membuat sample data pada database kita, untuk membuatnya klik **create database**

****

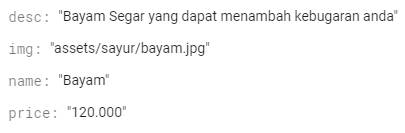
1. pilih **start in locked mode** dan klik tombol **next**



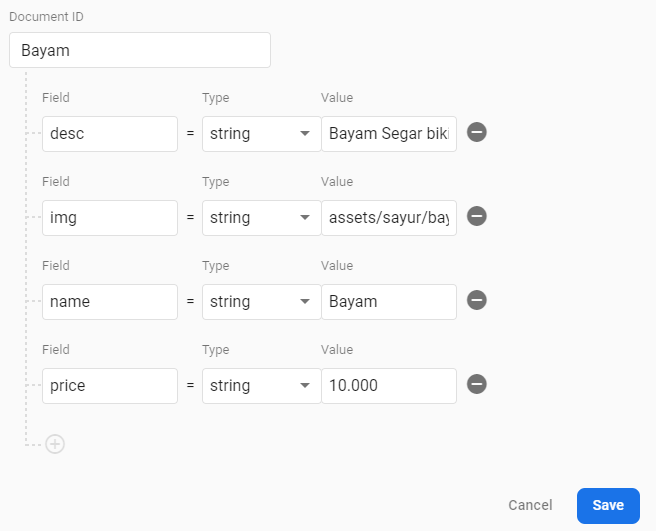
1. Setelah itu klik tombol **done** dan tunggu proses hingga selesai
2. Setelah proses selesai, pilih **start collection** 
3. Setelah itu, isi collection ID nya dan klik next



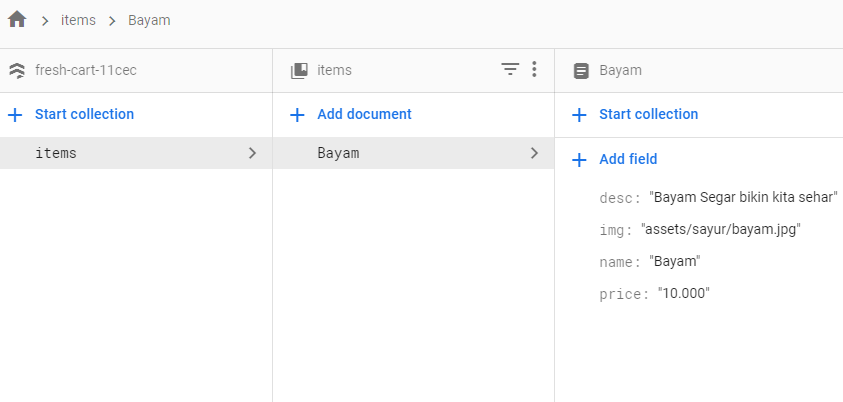
1. isi document ID nya, dokumen ID yang di maksud adalah nama item yang bakalan di panggil ke project kita menjadi sebuah product baik sayur ataupun buah, dan jangan isi juga fieldnya dengan nilai value berikut



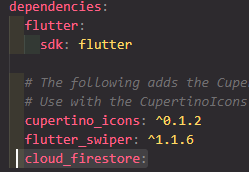
isi value dari documentnya dengan value di atas dan klik tombol **save**



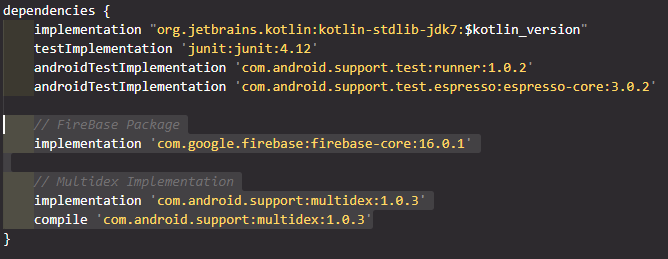
1. Setelah disave, seharusnya datanya akan muncul seperti ini

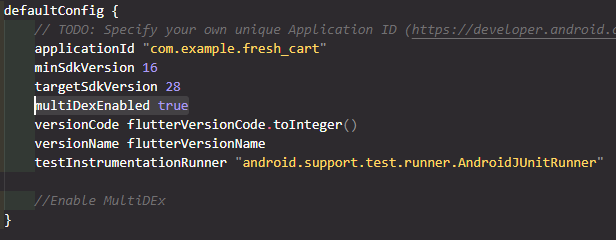


1. **Membuat Tampilan MyCart Screen**
2. kembali ke editor, buka file **pubspec.yaml** dan masukkan denpendencies **cloud\_firestore** dan save filenya

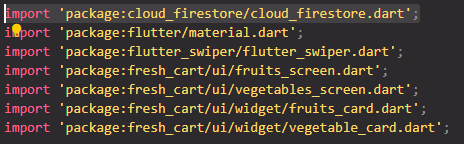
****

1. buka file **build.gradle** pada direktori **android/app** tambahkan code package firestore dan multidex berikut dalam tag dependencies serta tambahkan multidex: true pada defaultConfig

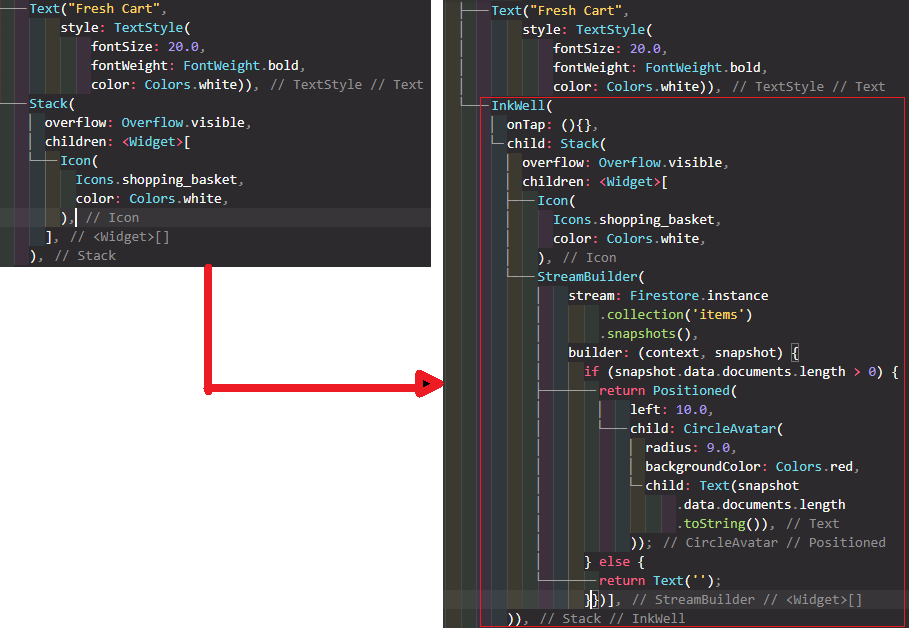




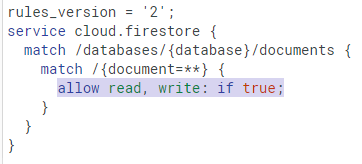
1. buka file **home.dart** pada direktori **ui** dan import package firestore paling atas



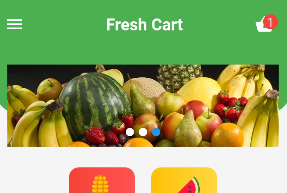
dan modifikasi kode berikut pada file **home.dart**

****

1. buka kembali web firebase console, klik tab **rules,** kemudian ganti **allow read, write: if false;** menjadi **allow read, write: if true;**

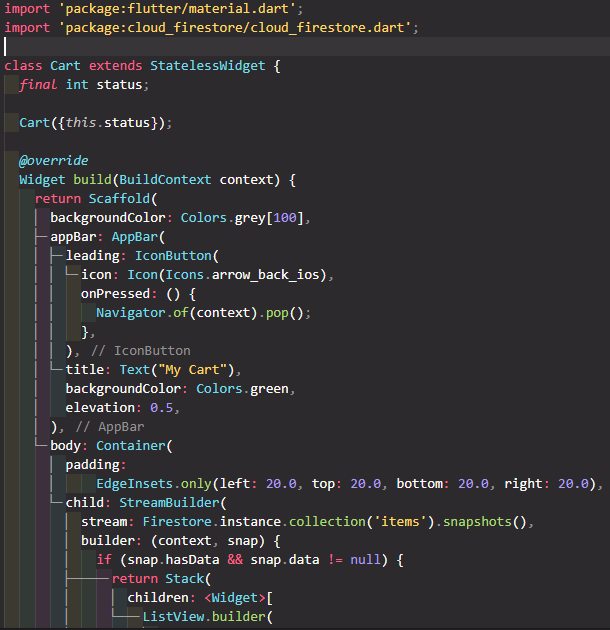
****

1. Setelah di rubah, Tekan tombol **publish**
2. Jalankan aplikasi, dan hasil akan seperti dibawah ini, dimana ada angka 1 diatas icon cart, kenapa ada 1? karena di table kita terdapat 1 **document / item** yaitu **Bayam** yang telah di tambahkan tadi



1. Sekarang kita akan membuat halaman cart, ketika icon cart pada pojok kanan di klik, maka akan muncul halaman list cart yang ada pada table firebase kita.

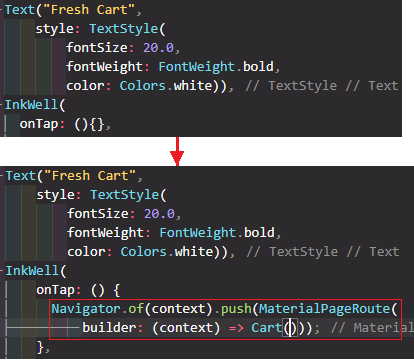
Buka file **cart\_screen.dart** pada folder **ui** kemudian tuliskan kode berikut







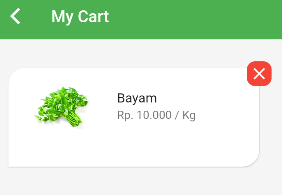
1. Setelah itu, buka kembali file **home.dart** dan modifikasi kode berikut



apabila terjadi error pada class **Cart()** klik icon lampu dan pilih **import ‘package:fresh\_cart/ui/cart\_screen.dart’** atau import code berikut



1. Jalankan atau Reload Aplikasinya, dan ketika menekan tombol **cart/keranjang** pada pojok kanan aplikasi, maka akan muncul tampilan list cart data dari firebase, seperti ini



Selamat, anda berhasil membuat list cart yang langsung mengambil data dari firebase atau dapat dikatakan **read** data.

1. **Membuat Fungsi Add Data ke Firebase**

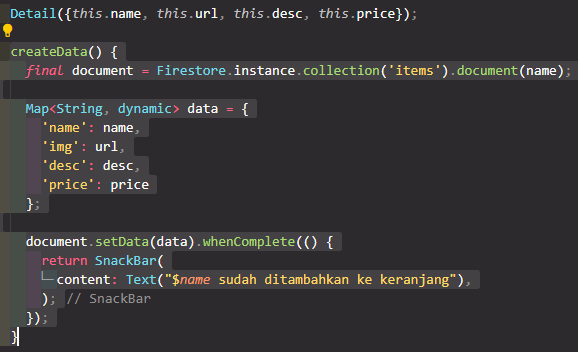
Di step ini kita akan membuat item yang dapat di tambahkan di tables/collections firebase kita, jadi logika dasarnya adalah ketika kita menekan item maka kita akan ke halaman detail, nah di halaman detail terdapat tombol add to cart, yang nantinya berfungsi untuk menambahkan item ke dalam table items yang ada pada firebase.

1. Buka file **detail.dart** pada direktori **ui,** kemudian modifikasi beberapa kode didalamnya,

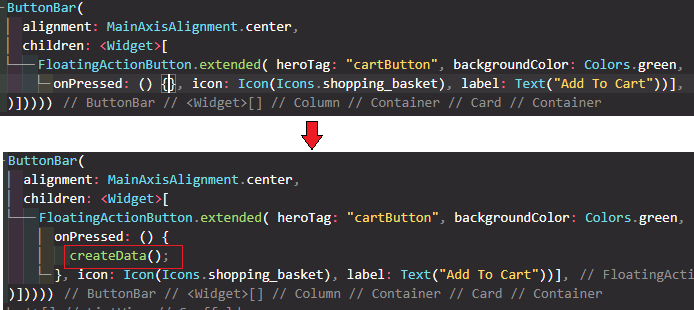
import package firestore :

****

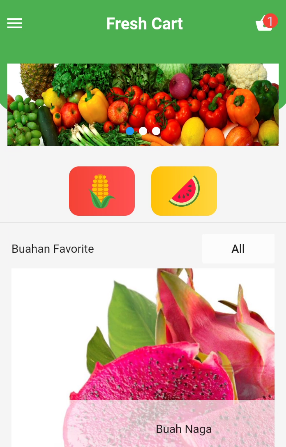
buat fungsi **createData()** tepatnya dibawah class **Detail() :**

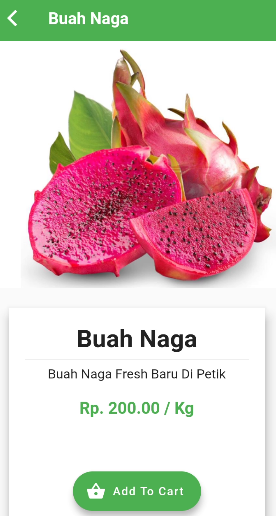
****

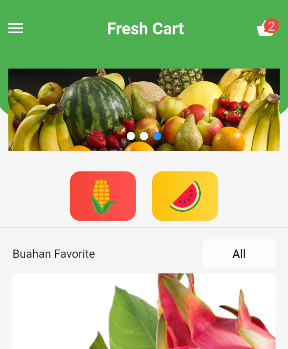
modifikasi kode pada button **add to cart** untuk memanggil fungsi **createData(),** yang nantinya apabila tombol tersebut di tekan maka akan masuk ke table firebase kita :

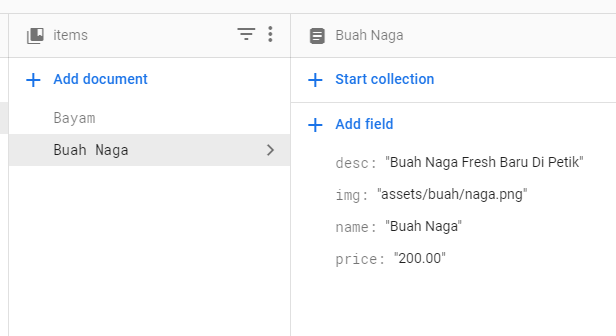


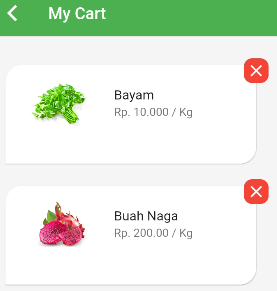
1. Running atau Restart aplikasi, dan ketika aplikasi sudah jalan, klik salah satu item yang ada dan akan di direct ke halaman **detail**, dihalaman **detail** klik tombol **add to cart** dan data akan disimpan ke table firebase kita, untuk mengetahuinya kita dapat membuka kembali console firebase pada browser anda untuk melihat apakah data sudah masuk atau belum, dan juga nomor pada icon **cart** akan bertambah











1. **Membuat Fungsi Delete Data ke Firebase**

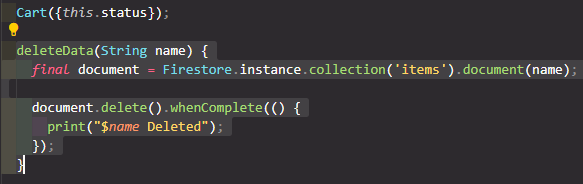
Di step ini, kita akan membuat fungsi delete data dari firebase, logikanya ketika kita mengklik tombol close/x item pada halaman list cart, maka datanya akan terhapus dari list dan juga dari tabel items yang ada pada firebase kita.

1. buka file **cart\_screen.dart** pada folder **ui** dan modifikasi beberapa kode didalamnya

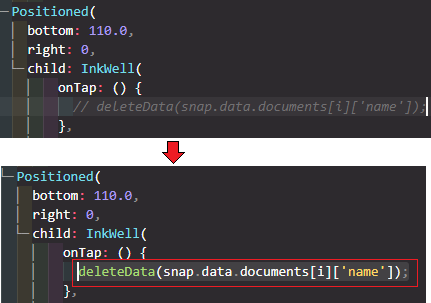
menambahkan atau mengimport package cloud\_firestore :



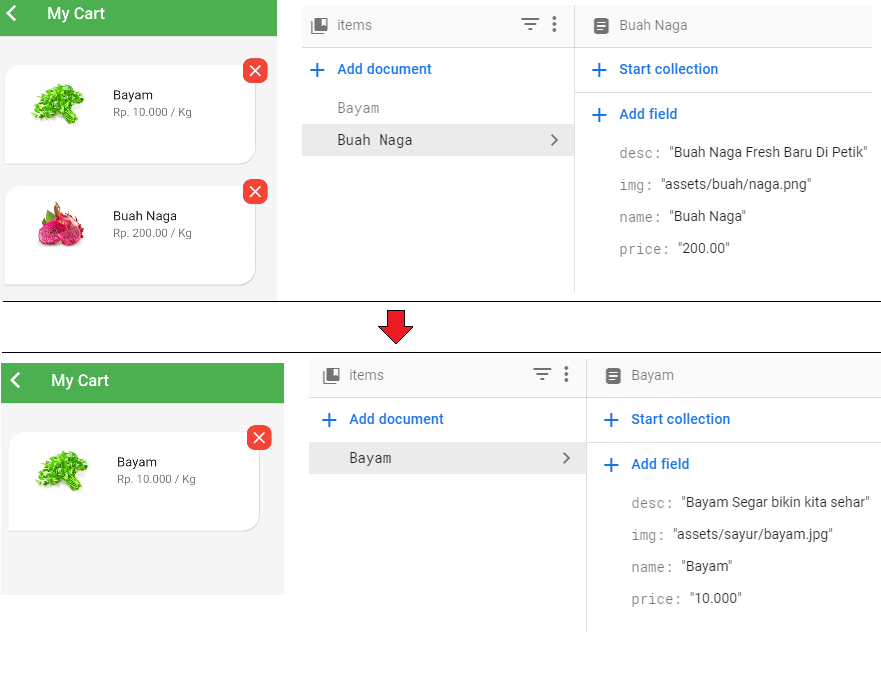
menambahkan fungsi **deleteData()** tepat dibawah class Constructor **Cart() :**

****

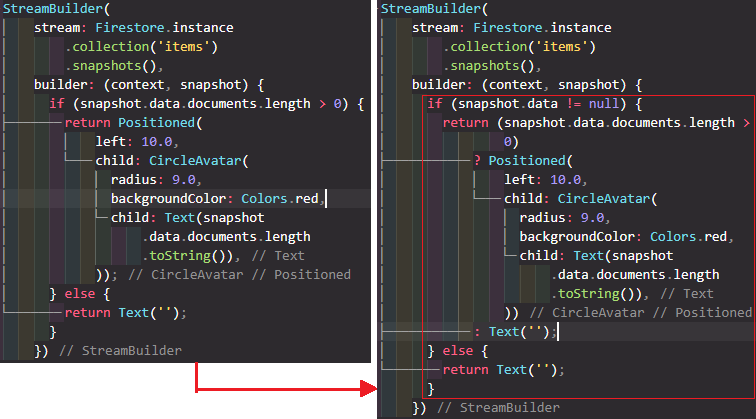
modifikasi tombol **x** untuk memanggil fungsi **deleteData()**, yang nantinya ketika tombol tersebut ditekan, maka data tersebut akan terhapus :



1. Running atau Restart aplikasi, dan ketika aplikasi sudah jalan, tekan tombol **x** salah satu item yang ada pada halaman **cart\_screen**, maka data tersebut akan terhapus pda list dan juga pada table di firebase kita, untuk mengetahuinya kita dapat membuka kembali console firebase pada browser anda untuk melihat apakah data sudah terhapus atau belum, dan juga nomor pada icon **cart** akan berkurang.



Selamat anda berhasil membuat fungsi untuk menghapus data, yang artinya aplikasi kita telah selesai, namun ada beberapa kode yang akan disempurnakan untuk menghandle error,



Selamat Aplikasi Sudah Aman dan dapat Digunakan dengan baik :)

1. **Push Ke Github ( Optional / Challenge )**

1. Buat Akun github[**disini**](https://github.com/)
2. Buat Repository dengan nama **dsc\_freshCart**
3. push project tersebut ke **github**