1. สร้าง Project
   1. สร้าง project ใหม่
   2. Setup firebase ตาม slide powerpoint firestore.ppt ใน Ms teams
   3. สร้างไฟล์ใหม่ ที่ โฟลเดอร์ lib ชื่อ addbook.dart

// ignore\_for\_file: avoid\_print, deprecated\_member\_use

import 'package:flutter/material.dart';

import 'package:cloud\_firestore/cloud\_firestore.dart';

class AddBookPage extends StatefulWidget {

  const AddBookPage({Key? key}) : super(key: key);

  @override

  State createState() => \_AddBookPageState();

}

class \_AddBookPageState extends State<AddBookPage> {

  final \_form = GlobalKey<FormState>();

  final \_title = TextEditingController();

  final \_detail = TextEditingController();

  final \_price = TextEditingController();

  final store = FirebaseFirestore.instance;

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: AppBar(

        title: const Text('Add book'),

      ),

      body: Form(

        key: \_form,

        child: ListView(

          children: <Widget>[

            buildTitleField(),

            buildDetailField(),

            buildPriceField(),

            buildSaveButton()

          ],

        ),

      ),

    );

  }

  ElevatedButton buildSaveButton() {

    return ElevatedButton(

        child: const Text('Save'),

        onPressed: () async {

          if (\_form.currentState!.validate()) {

            print('save button press');

            Map<String, dynamic> data = {

              'title': \_title.text,

              'detail': \_detail.text,

              'price': double.parse(\_price.text)

            };

            try {

              DocumentReference ref =

                  await store.collection('books').add(data);

              print('save id = ${ref.id}');

              Navigator.pop(context);

            } catch (e) {

               ScaffoldMessenger.of(context).showSnackBar(

                SnackBar(

                  content: Text('Error $e'),

                ),

              );

            }

          } else {

             ScaffoldMessenger.of(context).showSnackBar(

              const SnackBar(

                content: Text('Please validate value'),

              ),

            );

          }

        });

  }

  TextFormField buildTitleField() {

    return TextFormField(

      controller: \_title,

      decoration: const InputDecoration(

        labelText: 'title',

        icon: Icon(Icons.book),

      ),

      validator: (value) => value!.isEmpty ? 'Please fill in title' : null,

    );

  }

  TextFormField buildDetailField() {

    return TextFormField(

      controller: \_detail,

      decoration: const InputDecoration(

        labelText: 'detail',

        icon: Icon(Icons.list),

      ),

      validator: (value) => value!.isEmpty ? 'Please fill in detail' : null,

    );

  }

  TextFormField buildPriceField() {

    return TextFormField(

        controller: \_price,

        keyboardType: TextInputType.number,

        decoration: const InputDecoration(

          labelText: 'price',

          icon: Icon(Icons.list),

        ),

        validator: (value) {

          if (value!.isEmpty) {

            return 'Please fill in price';

          } else {

            double price = double.parse(value);

            if (price < 0) {

              return 'Please fill in price';

            } else {

              return null;

            }

          }

        });

  }

}

1. สร้างไฟล์ใหม่ ที่ โฟลเดอร์ lib ชื่อ book.dart

import 'package:flutter/material.dart';

import 'package:cloud\_firestore/cloud\_firestore.dart';

class BookPage extends StatelessWidget {

  final store = FirebaseFirestore.instance;

   BookPage({Key? key}) : super(key: key);

  @override

  Widget build(BuildContext context) {

    return StreamBuilder(

      stream: store.collection('books').snapshots(),

      builder: (context, AsyncSnapshot<QuerySnapshot> snapshot) {

        return Scaffold(

          appBar: AppBar(

            title: const Text("Books"),

            actions: <Widget>[buildAddButton(context)],

          ),

          body: snapshot.hasData

              ? buildBookList(snapshot.data!)

              : const Center(

                  child: CircularProgressIndicator(),

                ),

        );

      },

    );

  }

  IconButton buildAddButton(context) {

    return IconButton(

        icon: const Icon(Icons.add),

        onPressed: () {

          print("add icon press");

          Navigator.pushNamed(context, '/addbook');

        });

  }

  ListView buildBookList(QuerySnapshot data) {

    return ListView.builder(

      itemCount: data.size,

      itemBuilder: (BuildContext context, int index) {

        var model = data.docs.elementAt(index);

        return ListTile(

          title: Text(model['title']),

          subtitle: Text(model['detail']),

          trailing: Text("${model['price']}"),

          onTap: () {

            print(model['title']);

          },

        );

      },

    );

  }

}

1. ลบข้อมูลในไฟล์ lib/main.dart แทนที่ด้วย code ด้านล่าง แก้ไข path ของ project ให้ถูกต้อง

import 'package:flutter/material.dart';

import 'package:lab\_9\_firestore/addbook.dart';

import 'package:lab\_9\_firestore/book.dart';

import 'package:firebase\_core/firebase\_core.dart';

void main() async {

  WidgetsFlutterBinding.ensureInitialized();

  await Firebase.initializeApp();

  runApp(MyApp());

}

class MyApp extends StatelessWidget {

  @override

  Widget build(BuildContext context) {

    return MaterialApp(

      title: 'Firestore Demo',

      theme: ThemeData(

        primarySwatch: Colors.blue,

      ),

      home: buildMaterialApp(),

    );

  }

}

MaterialApp buildMaterialApp() {

  return MaterialApp(

    title: 'FireStore Demo',

    theme: ThemeData(

      primarySwatch: Colors.amber,

    ),

    initialRoute: '/',

    routes: {

      '/': (context) => BookPage(),

      '/addbook': (context) => AddBookPage(),

    },

  );

}

* 1. Run program และ แสดงผลลัพธ์

|  |
| --- |
|  |

1. สร้างไฟล์ใหม่ ที่ โฟลเดอร์ lib ชื่อ showdetail.dart

// ignore\_for\_file: avoid\_print

import 'package:cloud\_firestore/cloud\_firestore.dart';

import 'package:flutter/material.dart';

class BookDetail extends StatefulWidget {

  final String \_idi; //if you have multiple values add here

  const BookDetail(this.\_idi, {Key? key})

      : super(key: key); //add also..example this.abc,this...

  @override

  State createState() => \_BookDetailState();

}

class \_BookDetailState extends State<BookDetail> {

  final FirebaseFirestore \_firestore = FirebaseFirestore.instance;

  @override

  Widget build(BuildContext context) {

    String id = widget.\_idi;

    return StreamBuilder(

        stream: getBook(id),

        builder: (context, AsyncSnapshot<QuerySnapshot> snapshot) {

          return Scaffold(

            appBar: AppBar(

              title: const Text("Detail Books"),

            ),

            body: snapshot.hasData

                ? buildBookList(snapshot.data!)

                : const Center(

                    child: CircularProgressIndicator(),

                  ),

          );

        });

  }

  ListView buildBookList(QuerySnapshot data) {

    return ListView.builder(

      itemCount: data.docs.length,

      itemBuilder: (BuildContext context, int index) {

        var model = data.docs.elementAt(index);

        String a = model['detail'] + '  ' + model['price'].toString();

        return ListTile(

          title: Text(model['title']),

          //    subtitle: Text(model['detail'] + Text("${model['price']}")),

          subtitle: Text(a),

          trailing: ElevatedButton(

              child: const Text('Delete'),

              onPressed: () {

                print(model.id);

                //deleteValue(model.id);

                //Navigator.pop(context);

              }),

        );

      },

    );

  }

  //Future<void> deleteValue(String titleName) async {

  //  await \_firestore

  //      .collection('books')

  //      .doc(titleName)

  //      .delete()

  //      .catchError((e) {

  //    print(e);

  //  });

  //}

  Stream<QuerySnapshot> getBook(String titleName) {

    // Firestore \_firestore = Firestore.instance;

    return \_firestore

        .collection('books')

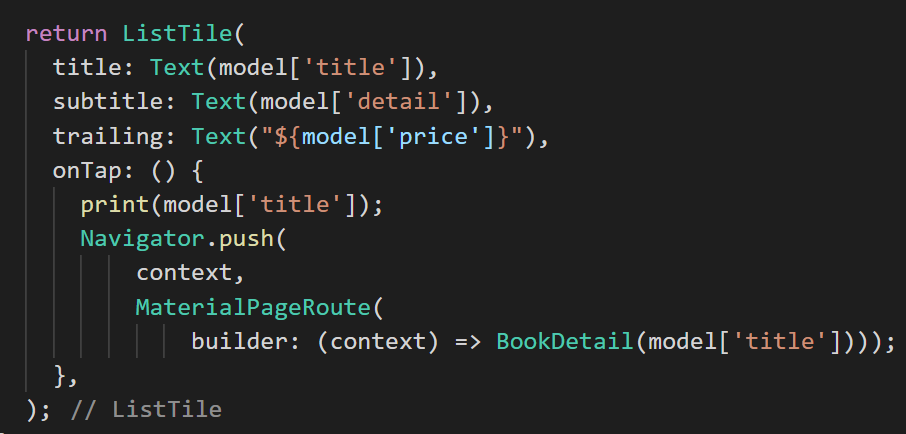
        .where('title', isEqualTo: titleName)

        .snapshots();

  }

}

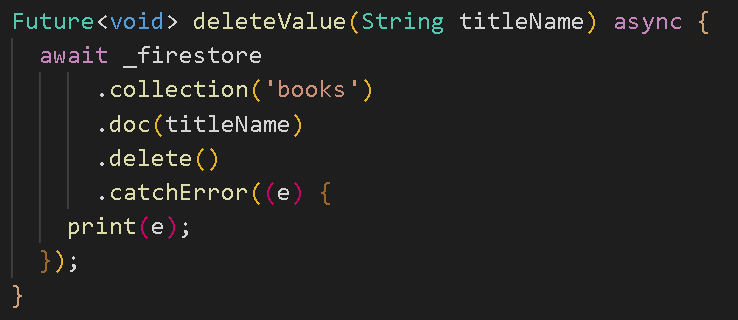
1. แก้ไขไฟล์ book.dart
   1. import 'package:YOUR\_PROJECT/showdetail.dart';
   2. เพิ่ม ontab() ของ ListTile ที่ class buildBookList เพื่อ เปลี่ยนหน้าไปยังหน้าแสดงรายละเอียดหนังสือ



* 1. Run program และ แสดงผลลัพธ์

|  |
| --- |
|  |

1. แก้ไขไฟล์ showdetail.dart
   1. เพิ่ม deleteValue ใน class \_BookDetailState เพื่อ ลบหนังสือเมื่อกด ปุ่ม delete



* 1. แก้ไข onPressed() ของ ElevatedButton ใน ListiTile ใน buildBookList class เอาเครื่องหมาย // ออก เพื่อเรียกใช้งาน deleteValue และกลับไปยังหน้าแรก
  2. Run program และ แสดงผลลัพธ์

|  |
| --- |
|  |

8. แก้ไขโปรแกรมโดยเพิ่ม ข้อมูล ผู้แต่ง author อีก 1 ข้อมูล โดยลบข้อมูลหนังสือเดิมทั้งหมดออกก่อน แล้วจึงแก้ไขโปรแกรมเพิ่ม author เข้าไป และแสดงข้อมูล ผู้แต่ง ในหน้า book และ showdetail ด้วย

8.1 Run program และ แสดงผลลัพธ์

|  |
| --- |
|  |