**FIREBASE CHEAT SHEET FOR FLUTTER**

1. ADDING DATA

async {  
 Navigator.*of*(context).pop();  
 await \_project.collection("project").document(myController.text).setData({  
 "project\_name": myController.text,  
 "project\_id": 50,  
 "project\_code": randomAlphaNumeric(8),  
 "created\_at": DateTime.now(),  
 "created\_by": Variables.*currentEmail*,  
 });

1. FETCHING DATA

\_note.collection("notes")  
 .where("notes\_id", isEqualTo: Variables.*currentNote*).snapshots().listen((data)=>data.documents.forEach((doc) {  
 setState(() {  
 Variables.*currentNoteTitle*=doc["notes\_title"];  
 Variables.*currentNoteContent*=doc["project\_notes"];

}

1. UPDATING or EDITING DATA

await \_note.collection("notes").document(Variables.*currentNote*).updateData({  
 "notes\_title": title,  
 "project\_notes": text,  
 "project\_name": Variables.*currentProject*,  
 "updated\_at": DateTime.now(),  
 "updated\_by": Variables.*currentEmail*,  
 "category": Variables.*notesPage*,  
});

OR

onPressed: () async{  
 await \_project  
 .collection("project")  
 .where("project\_code", isEqualTo: myController.text).snapshots().listen((data)=>data.documents.forEach((doc) {  
 setState(() {  
 S=doc["project\_name"];  
 print(S);  
 \_project.collection("project").document(S).updateData({"collaborators": FieldValue.*arrayUnion*([Variables.*currentEmail*])});  
 });  
 }));  
 Navigator.*of*(context).pop();  
},

1. DISPLAYING DATA

StreamBuilder(  
 stream: \_project  
 .collection("project")  
 .where("collaborators",arrayContains: Variables.*currentEmail*)  
 .snapshots(),  
 builder: (context, snapshot) {  
 if (!snapshot.hasData) {  
 return Text(  
 'No Data...',  
 );  
 } else {  
 var lists = snapshot.data.documents;  
 return ListView.builder(  
 shrinkWrap: true,  
 itemCount: lists.length,  
 itemBuilder: (BuildContext context, int index) {  
 return card(lists[index]["project\_name"]);  
 });  
 }  
 },  
),