**Cloud Firestore**

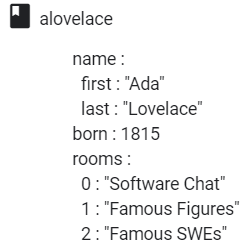
Google’s Cloud Firestore is a scalable, cloud database for mobile, web, and server development. It uses real-time listeners to keep your data synced and keeps apps running seamlessly regardless of any network connectivity issues.

**Structure**

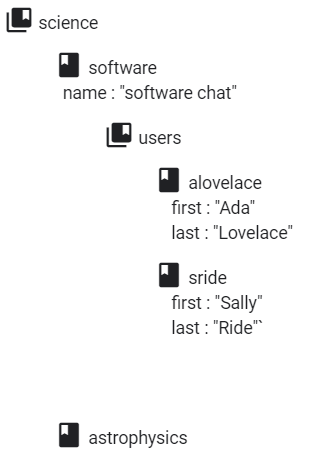
Firestore is a NoSQL database which stores data into documents which are arranged into collections. Queries can retrieve data at the document level and can be sorted or filtered for specific results. Each document contains sets of key-value pairs which map fields to their corresponding values. As shown below, a collection named “users” contains multiple users each represented by a document.



Firestore also allows you to nest objects within documents.



Subcollections are useful for growing lists and allow for specific collection group queries. However, they cannot be easily removed.



Cloud Firestore allows many data types to be stored including arrays, dates and times, geographical points, etc.

**SDKs/Libraries**

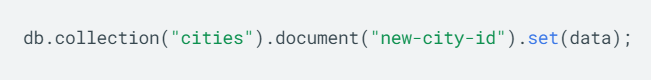
While the Cloud Firestore API can be accessed using direct HTTP calls, Firestore client libraries can be implemented to access the database.

Cloud Firestore supports Android SDKs where clients can connect directly to the database using Firebase Auth and Cloud Firestore security rules. Cloud Firestore also contains server client libraries which set up privileged server environments where users are authorized using Identity and Access Management (IAM). Server client libraries should be used for administrative database tasks or if you prefer an intermediary server between user and database. Lastly, Firebase Admin SDKs combine Google Cloud client libraries along with other client libraries and SDKs to initialize access to Firestore and other services from a single SDK.

**Adding/Removing Data**

There are three ways to write data to a Firestore database. Since each document within a collection needs a document identifier, either Firestore will automatically generate this identifier or it can be explicitly specified.

The first way of writing data is to specify the document identifier of the document you want to write to and set its data using the set() method.



The second way is to add a new document to a collection and let Firestore automatically generate the document identifier using the add() method.



The third way is to create an empty document and to assign data later while letting Firestore automatically generate the document identifier.

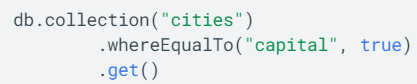
For deleting documents, the delete() method deletes the document but does not delete its subcollections.



Deleting specific fields from a document uses the FieldValue.delete() method. Deleting collections in Android is not recommended.

**Reading Data**

Retrieving data in Firestore can be done by calling a method or by setting a listener for real-time updates. Getting a single document can be done by calling the get() method on a document. You can also retrieve multiple documents from a collection by using the where() method to query all documents that match the parameters.



Firestore also provides query functionality that can be used with get() methods.



These queries include query operators such as less than, greater than, not equal to, etc. These operators can be chained together to create compound queries.



Collection group queries retrieve documents from a collection group instead of a single collection.



**Cloud Firestore REST API**

Accessing the Cloud Firestore using a REST API occurs through authentication with a Firebase Authentication token or a Google Identity OAuth 2.0 token. Firebase ID token requests are authorized using Firebase Security rules while Google Identity OAuth token requests are authorized using Identity and Access Management.