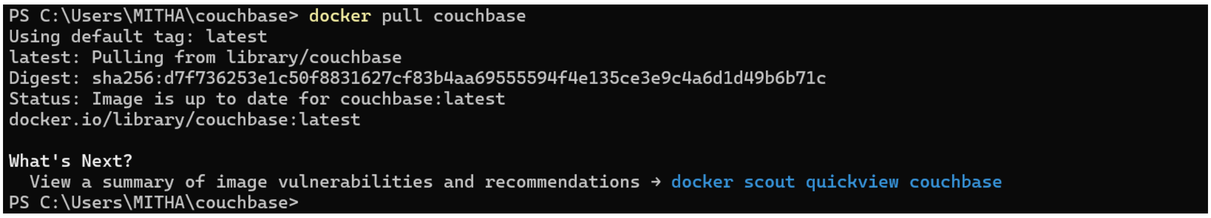
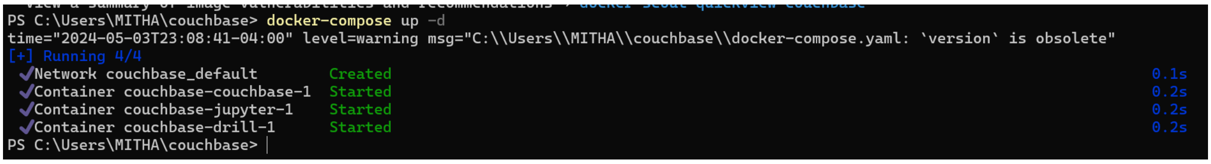
COUCHBASE :

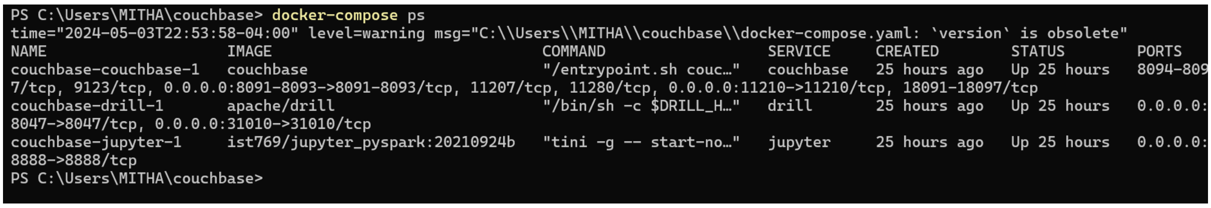
We have used docker to run the couchbase server locally and it is accessible by the port localhost:8091 .



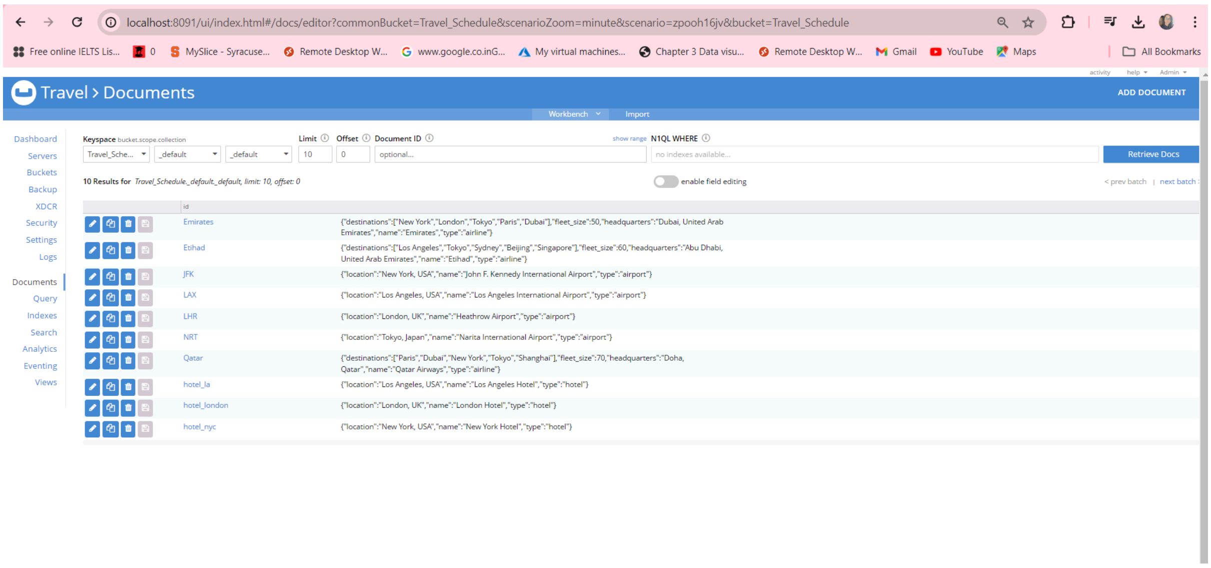
```
PS C:\Users\MITHA\couchbase> docker pull couchbase
Using default tag: latest
latest: Pulling from library/couchbase
Digest: sha256:d7f736253e1c50f8831627cf83b4aa69555594f4e135ce3e9c4a6d1d49b6b71c
Status: Image is up to date for couchbase:latest
docker.io/library/couchbase:latest

What's Next?
  View a summary of image vulnerabilities and recommendations → docker scout quickview couchbase
PS C:\Users\MITHA\couchbase>
```



```
  View a summary of image vulnerabilities and recommendations → docker scout quickview couchbase
PS C:\Users\MITHA\couchbase> docker-compose up -d
time="2024-05-03T23:08:41-04:00" level=warning msg="C:\\Users\\MITHA\\couchbase\\docker-compose.yaml: `version` is obsolete"
[+] Running 4/4
 ✔Network couchbase_default         Created                                                                  0.1s
 ✔Container couchbase-couchbase-1   Started                                                                  0.2s
 ✔Container couchbase-jupyter-1     Started                                                                  0.2s
 ✔Container couchbase-drill-1       Started                                                                  0.2s
PS C:\Users\MITHA\couchbase>
```



```
PS C:\Users\MITHA\couchbase> docker-compose ps
time="2024-05-03T22:53:58-04:00" level=warning msg="C:\\Users\\MITHA\\couchbase\\docker-compose.yaml: `version` is obsolete"
NAME                     IMAGE                                COMMAND                  SERVICE     CREATED          STATUS          PORTS
couchbase-couchbase-1    couchbase                            "/entrypoint.sh couc…"   couchbase   25 hours ago     Up 25 hours     8094-809
7/tcp, 9123/tcp, 0.0.0.0:8091-8093->8091-8093/tcp, 11207/tcp, 11280/tcp, 0.0.0.0:11210->11210/tcp, 18091-18097/tcp
couchbase-drill-1        apache/drill                         "/bin/sh -c $DRILL_H…"   drill       25 hours ago     Up 25 hours     0.0.0.0:
8047->8047/tcp, 0.0.0.0:31010->31010/tcp
couchbase-jupyter-1      ist769/jupyter_pyspark:20210924b     "tini -g -- start-no…"   jupyter     25 hours ago     Up 25 hours     0.0.0.0:
8888->8888/tcp
PS C:\Users\MITHA\couchbase>
```



Keyspace bucket.scope.collection: Travel_Sche... _default _default
Limit: 10
Offset: 0
Document ID: optional...
show range
N1QL WHERE: no indexes available...

10 Results for *Travel_Schedule._default._default, limit: 10, offset: 0*

id	
Emirates	{"destinations":["New York","London","Tokyo","Paris","Dubai"],"fleet_size":50,"headquarters":"Dubai, United Arab Emirates","name":"Emirates","type":"airline"}
Etihad	{"destinations":["Los Angeles","Tokyo","Sydney","Beijing","Singapore"],"fleet_size":60,"headquarters":"Abu Dhabi, United Arab Emirates","name":"Etihad","type":"airline"}
JFK	{"location":"New York, USA","name":"John F. Kennedy International Airport","type":"airport"}
LAX	{"location":"Los Angeles, USA","name":"Los Angeles International Airport","type":"airport"}
LHR	{"location":"London, UK","name":"Heathrow Airport","type":"airport"}
NRT	{"location":"Tokyo, Japan","name":"Narita International Airport","type":"airport"}
Qatar	{"destinations":["Paris","Dubai","New York","Tokyo","Shanghai"],"fleet_size":70,"headquarters":"Doha, Qatar","name":"Qatar Airways","type":"airline"}
hotel_la	{"location":"Los Angeles, USA","name":"Los Angeles Hotel","type":"hotel"}
hotel_london	{"location":"London, UK","name":"London Hotel","type":"hotel"}
hotel_nyc	{"location":"New York, USA","name":"New York Hotel","type":"hotel"}

Here we have created Bucket called Travel and keyspace as Travel_Schedule.

For performing crud operations in couchbase we have used query service. In the query service , we have N1QL Query Language which is similar to SQL . For our Trip planner Project we used this N1QL query for performing basic CRUD Operations in Couchbase.N1QL is the query language used in Couchbase to query and manipulate JSON data stored in the database. We have created 4 documents in total which are Airlines , Airports, Routes and Hotels with each document uniquely identified by the key.

CRUD OPERATIONS IN COUCHBASE:

1. CREATE:

AIRLINE Document:

Documents created for each airline with the airline's name as the key.

- **"type"**: Specifies the type of entity (in this case, "airline").
- **"name"**: The name of the airline.
- **"headquarters"**: The location of the airline's headquarters.
- **"fleet_size"**: The number of aircraft in the airline's fleet.
- **"destinations"**: A list of destinations served by the airline.

```
INSERT INTO `Travel_Schedule` (KEY, VALUE)
VALUES
  ('Emirates',
  {
    "type": "airline",
    "name": "Emirates",
    "headquarters": "Dubai, United Arab Emirates",
    "fleet_size": 50,
    "destinations": ["New York", "London", "Tokyo", "Paris", "Dubai"]
  }),

  ('Etihad',
  {
    "type": "airline",
    "name": "Etihad",
    "headquarters": "Abu Dhabi, United Arab Emirates",
    "fleet_size": 60,
    "destinations": ["Los Angeles", "Tokyo", "Sydney", "Beijing", "Singapore"]
  }),

  ('Qatar',
  {
    "type": "airline",
    "name": "Qatar Airways",
    "headquarters": "Doha, Qatar",
    "fleet_size": 70,
    "destinations": ["Paris", "Dubai", "New York", "Tokyo", "Shanghai"]
  });
```

ROUTE Document:

Documents created for each flight route with a unique key representing the route.

- **"type"**: Specifies the type of entity (in this case, "route").
- **"airline"**: The name of the airline operating the route.
- **"departure_airport"**: The code or name of the departure airport.
- **"arrival_airport"**: The code or name of the arrival airport.
- **"distance"**: The distance of the route in kilometers.
- **"flight_time"**: The duration of the flight.

```
INSERT INTO `Travel_Schedule` (KEY, VALUE)
VALUES
  ('nyc_london',
  {
    "type": "route",
    "airline": "Emirates",
    "departure_airport": "JFK",
    "arrival_airport": "LHR",
    "distance": 5555,
    "flight_time": "07:30"
}),

('la_tokyo',
{
    "type": "route",
    "airline": "Etihad",
    "departure_airport": "LAX",
    "arrival_airport": "NRT",
    "distance": 8800,
    "flight_time": "11:00"
}),

('paris_dubai',
{
    "type": "route",
    "airline": "Qatar",
    "departure_airport": "CDG",
    "arrival_airport": "DXB",
    "distance": 5400,
    "flight_time": "06:45"
});
```

HOTEL Document:

Documents created for each hotel with the Hotel name as the key.

- **"type"**: Specifies the type of entity (in this case, "hotel").
- **"hotel"**: The name of the hotel.
- **location:** Hotel location

```
INSERT INTO `Travel_Schedule` (KEY, VALUE)

VALUES
```

```
   ('hotel_nyc',
   {
"type": "hotel",
"name": "New York Hotel",
"location": "New York, USA"
   }),

   ('hotel_london',
   {
"type": "hotel",
"name": "London Hotel",
"location": "London, UK"
   }),

   ('hotel_la',
   {
"type": "hotel",
"name": "Los Angeles Hotel",
"location": "Los Angeles, USA"
   }),

   ('hotel_tokyo',
   {
"type": "hotel",

"name": "Tokyo Hotel",

"location": "Tokyo, Japan"

   });
```

Airport Document:

Four airports are being added: JFK, LHR, LAX, and NRT.

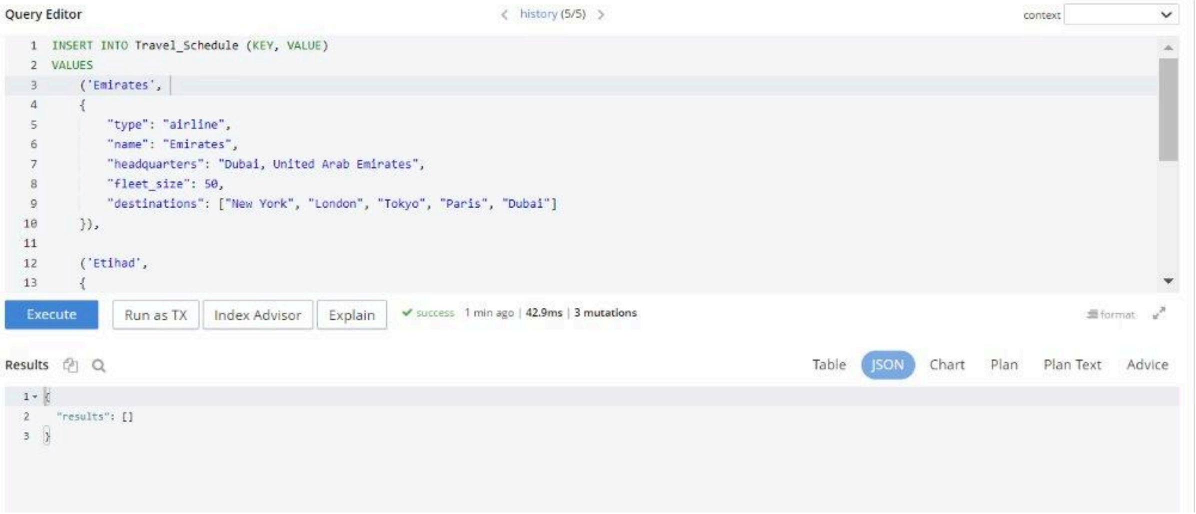
Each airport document contains attributes such as:

"type": Indicates the entity type, in this case, "airport".

"name": The name of the airport.

"location": The location of the airport, including city and country

```
INSERT INTO `Travel_Schedule` (KEY, VALUE)

VALUES

  ('JFK',

  {

    "type": "airport",

    "name": "John F. Kennedy International Airport",

    "location": "New York, USA"

  }),


  ('LHR',

  {

    "type": "airport",

    "name": "Heathrow Airport",

    "location": "London, UK"

  }),
('LAX',
{
   "type": "airport",
   "name": "Los Angeles International Airport",
   "location": "Los Angeles, USA"
}),

('NRT',
{
   "type": "airport",
   "name": "Narita International Airport",
   "location": "Tokyo, Japan"
});
```



```
INSERT INTO Travel_Schedule (KEY, VALUE)
VALUES
    ('Emirates',
    {
        "type": "airline",
        "name": "Emirates",
        "headquarters": "Dubai, United Arab Emirates",
        "fleet_size": 50,
        "destinations": ["New York", "London", "Tokyo", "Paris", "Dubai"]
    }),

    ('Etihad',
    {
```

✓ success 1 min ago | 42.9ms | 3 mutations

```
{
  "results": []
}
```

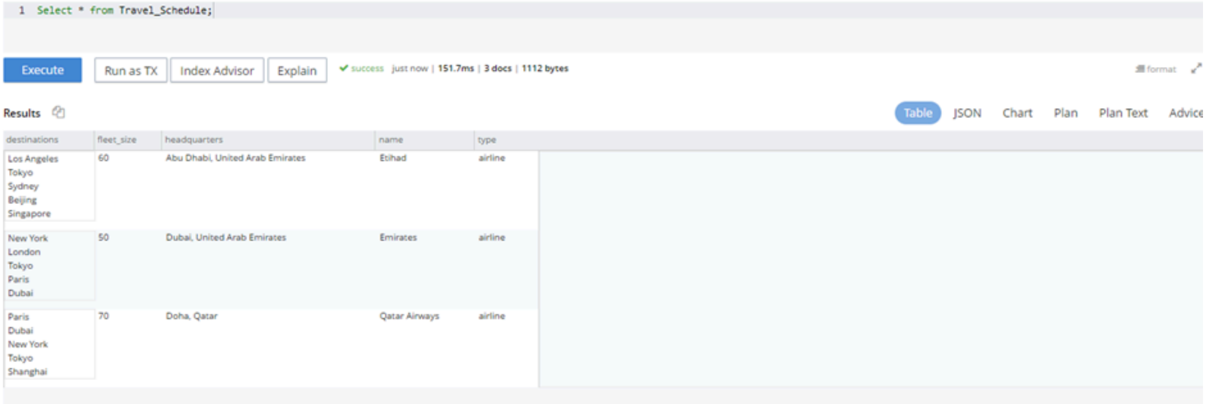
2. READ:

- Each SELECT query follows the same structure, specifying the Travel_Schedule bucket and filtering documents based on their type attribute.

- The WHERE clause restricts the result set to documents of a specific type (e.g., "airline", "route", "airport", "hotel").

- By executing these queries, we can retrieve subsets of documents from the Travel_Schedule bucket based on their entity type, facilitating easy access to relevant data for various travel management purposes.

SELECT * FROM Travel_Schedule WHERE type = "airline";

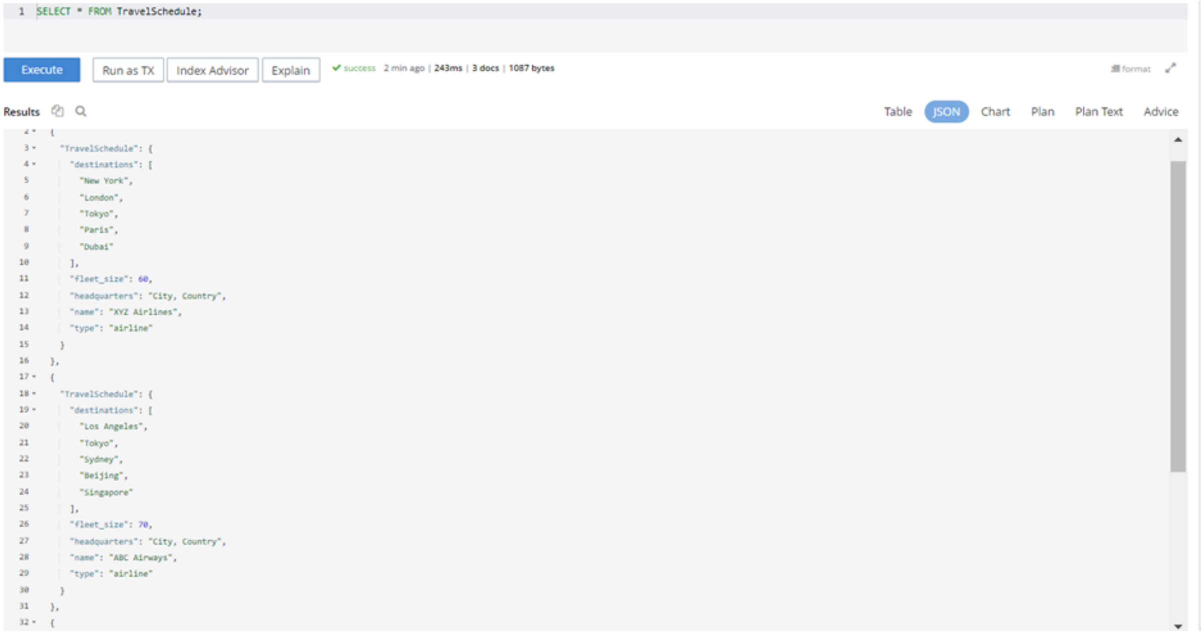


```
Select * from Travel_Schedule;
```

✓ success just now | 151.7ms | 3 docs | 1112 bytes

Results

destinations	fleet_size	headquarters	name	type
Los Angeles Tokyo Sydney Beijing Singapore	60	Abu Dhabi, United Arab Emirates	Etihad	airline
New York London Tokyo Paris Dubai	50	Dubai, United Arab Emirates	Emirates	airline
Paris Dubai New York Tokyo Shanghai	70	Doha, Qatar	Qatar Airways	airline



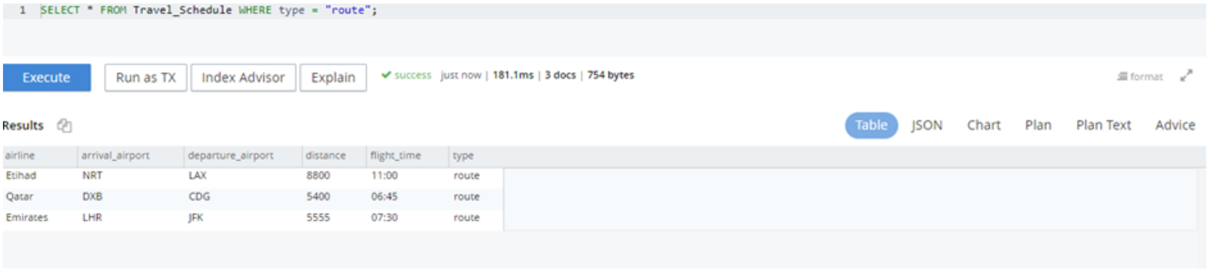
```
SELECT * FROM TravelSchedule;
```

✓ success 2 min ago | 243ms | 3 docs | 1087 bytes

Results

```
  {
    "TravelSchedule": {
      "destinations": [
        "New York",
        "London",
        "Tokyo",
        "Paris",
        "Dubai"
      ],
      "fleet_size": 60,
      "headquarters": "City, Country",
      "name": "XYZ Airlines",
      "type": "airline"
    }
  },
  {
    "TravelSchedule": {
      "destinations": [
        "Los Angeles",
        "Tokyo",
        "Sydney",
        "Beijing",
        "Singapore"
      ],
      "fleet_size": 70,
      "headquarters": "City, Country",
      "name": "ABC Airways",
      "type": "airline"
    }
  },
  {
```

SELECT * FROM Travel_Schedule WHERE type = "route";

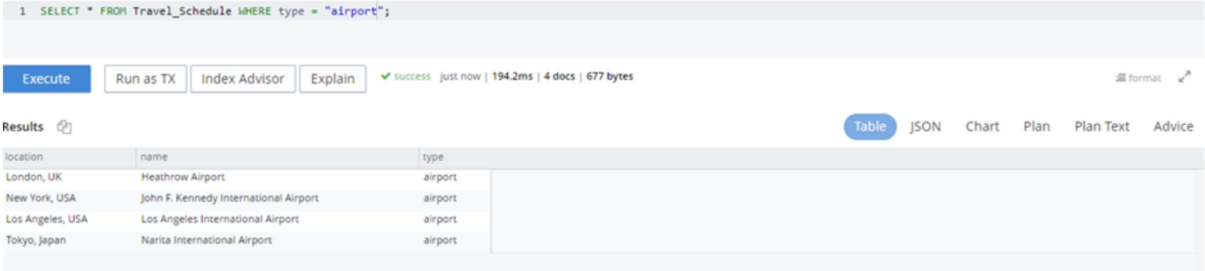


```
SELECT * FROM Travel_Schedule WHERE type = "route";
```

✔ success just now | 181.1ms | 3 docs | 754 bytes

airline	arrival_airport	departure_airport	distance	flight_time	type
Etihad	NRT	LAX	8800	11:00	route
Qatar	DXB	CDG	5400	06:45	route
Emirates	LHR	JFK	5555	07:30	route

SELECT * FROM Travel_Schedule WHERE type = "airport";

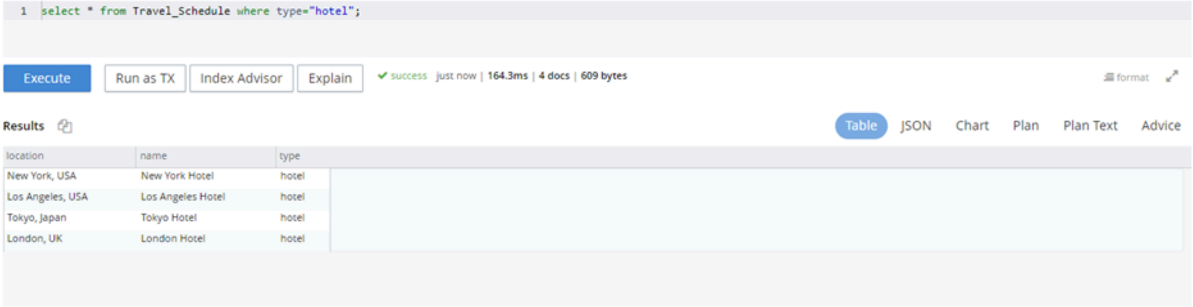


```
SELECT * FROM Travel_Schedule WHERE type = "airport";
```

✔ success just now | 194.2ms | 4 docs | 677 bytes

location	name	type
London, UK	Heathrow Airport	airport
New York, USA	John F. Kennedy International Airport	airport
Los Angeles, USA	Los Angeles International Airport	airport
Tokyo, Japan	Narita International Airport	airport

SELECT * FROM Travel_Schedule WHERE type = "hotel";



```
select * from Travel_Schedule where type="hotel";
```

✔ success just now | 164.3ms | 4 docs | 609 bytes

location	name	type
New York, USA	New York Hotel	hotel
Los Angeles, USA	Los Angeles Hotel	hotel
Tokyo, Japan	Tokyo Hotel	hotel
London, UK	London Hotel	hotel

3. UPDATE:

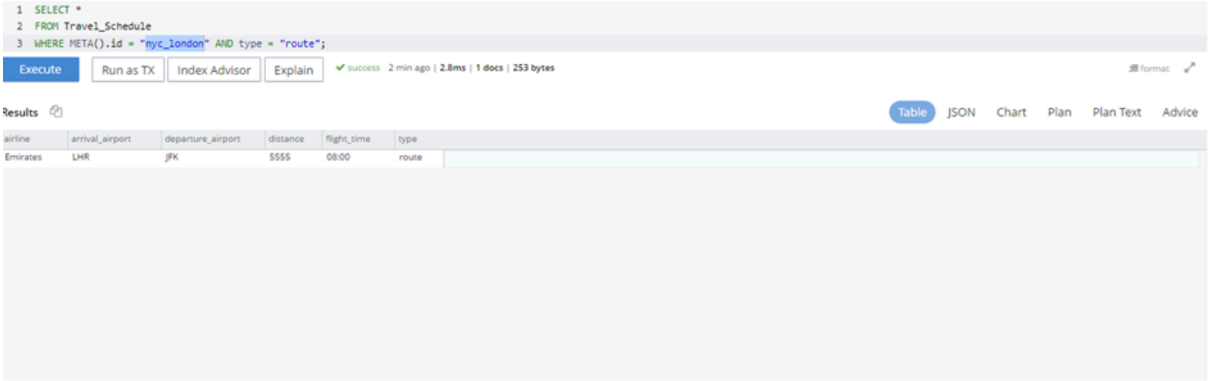
UPDATE Travel_Schedule

SET flight_time = "08:00"

WHERE META().id = "nyc_london" AND type = "route";

- **UPDATE** statement targets a specific document in the **Travel_Schedule** bucket, identified by its key **"nyc_london"**.
- It modifies the **flight_time** attribute of the route document with the key **"nyc_london"**, changing its value to **"08:00"**.
- By using the **META().id** function, Couchbase can efficiently locate and update the document without needing to perform a full document scan.

- This operation ensures that the flight time for the route from New York to London is updated to **"08:00"**, reflecting any changes to the schedule accurately.



```
SELECT *
FROM Travel_Schedule
WHERE META().id = "nyc_london" AND type = "route";
```

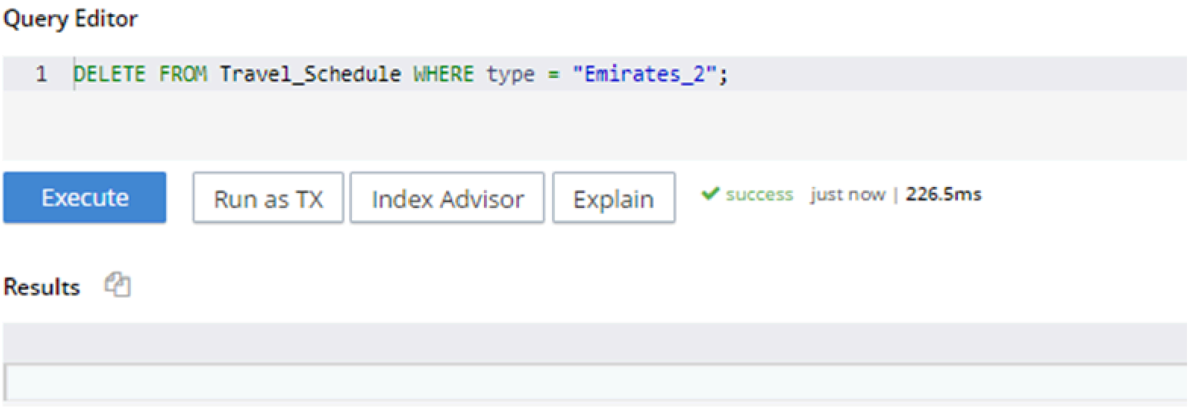
airline	arrival_airport	departure_airport	distance	flight_time	type
Emirates	LHR	JFK	5555	08:00	route

4.DELETE:

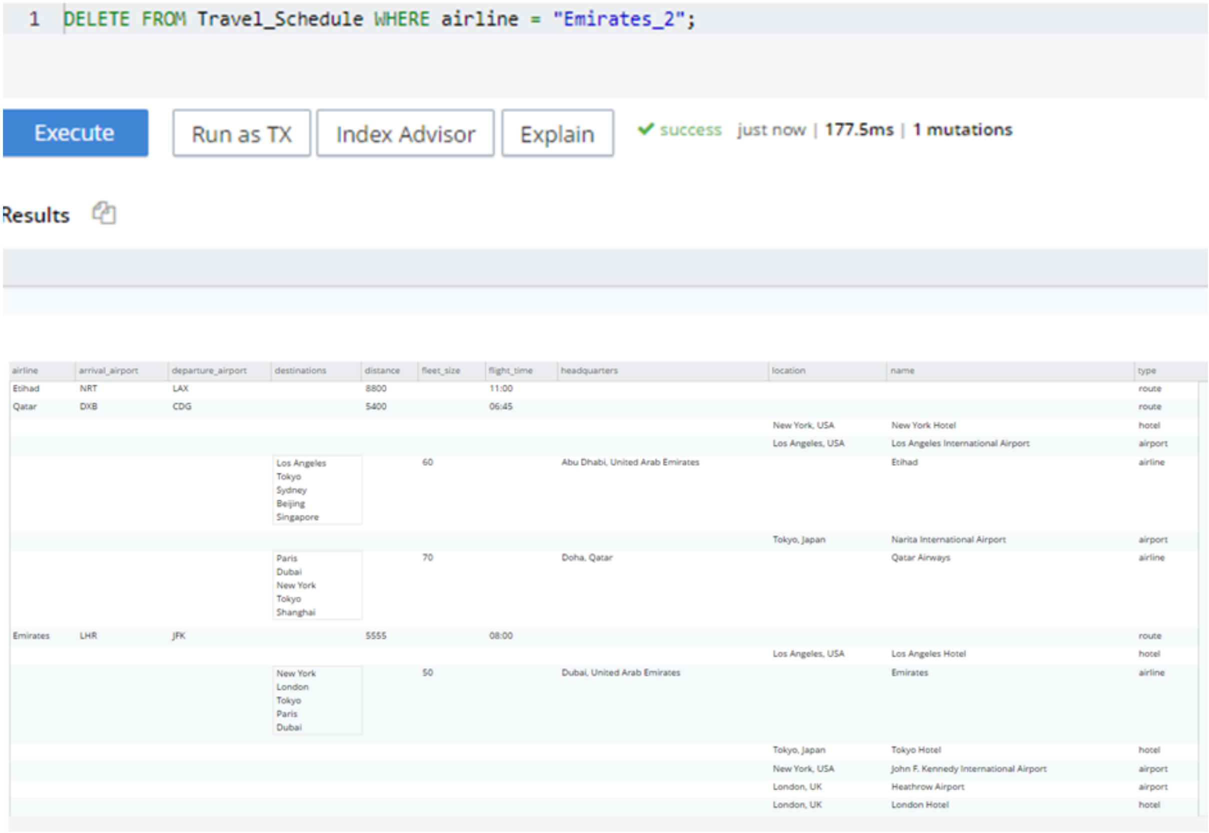
Delete from Travel_Schedule where type=”Emirates_2”

Delete from Travel_Schedule where route=”Emirates_2”

This command deletes documents from the **Travel_Schedule** bucket where the **type** attribute is "route" and the **airline** attribute is "Emirates". This accurately targets documents representing routes operated by the airline "Emirates".



```
DELETE FROM Travel_Schedule WHERE type = "Emirates_2";
```



```
DELETE FROM Travel_Schedule WHERE airline = "Emirates_2";
```

airline	arrival_airport	departure_airport	destinations	distance	fleet_size	flight_time	headquarters	location	name	type
Etihad	NRT	LAX		8800		11:00				route
Qatar	DXB	CDG		5400		06:45				route
								New York, USA	New York Hotel	hotel
								Los Angeles, USA	Los Angeles International Airport	airport
			Los Angeles Tokyo Sydney Beijing Singapore		60		Abu Dhabi, United Arab Emirates		Etihad	airline
								Tokyo, Japan	Narita International Airport	airport
			Paris Dubai New York Tokyo Shanghai		70		Doha, Qatar		Qatar Airways	airline
Emirates	LHR	JFK		5555		08:00				route
								Los Angeles, USA	Los Angeles Hotel	hotel
			New York London Tokyo Paris Dubai		50		Dubai, United Arab Emirates		Emirates	airline
								Tokyo, Japan	Tokyo Hotel	hotel
								New York, USA	John F. Kennedy International Airport	airport
								London, UK	Heathrow Airport	airport
								London, UK	London Hotel	hotel

REFERENCES AND CITATIONS:

https://docs.couchbase.com/home/server.html

https://www.youtube.com/watch?v=9ir8-81vLY0

https://cloud.couchbase.com/playground/sqlPlusPlus/LEARN_QUERY_BASICS?oid=580a995c-7bab-47c1-8f88-dbd13c9f5559

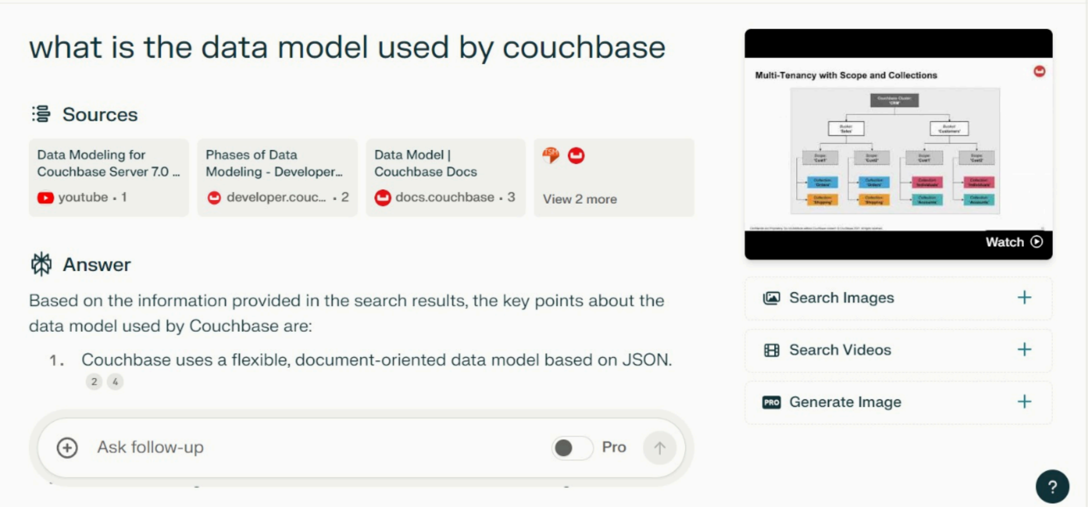
https://docs.couchbase.com/cloud/get-started/run-first-queries.html

https://github.com/couchbase-examples/python-quickstart/

https://docs.couchbase.com/cloud/get-started/create-account.html

https://www.perplexity.ai/search/couch-base-applicattions-uoDxLhlASvG5f3hrMtZIBA

PERPLEXITY AI PROMPT AND OUTPUT:



what is the data model used by couchbase

Sources

Data Modeling for Couchbase Server 7.0 ... youtube · 1

Phases of Data Modeling - Developer... developer.couc... · 2

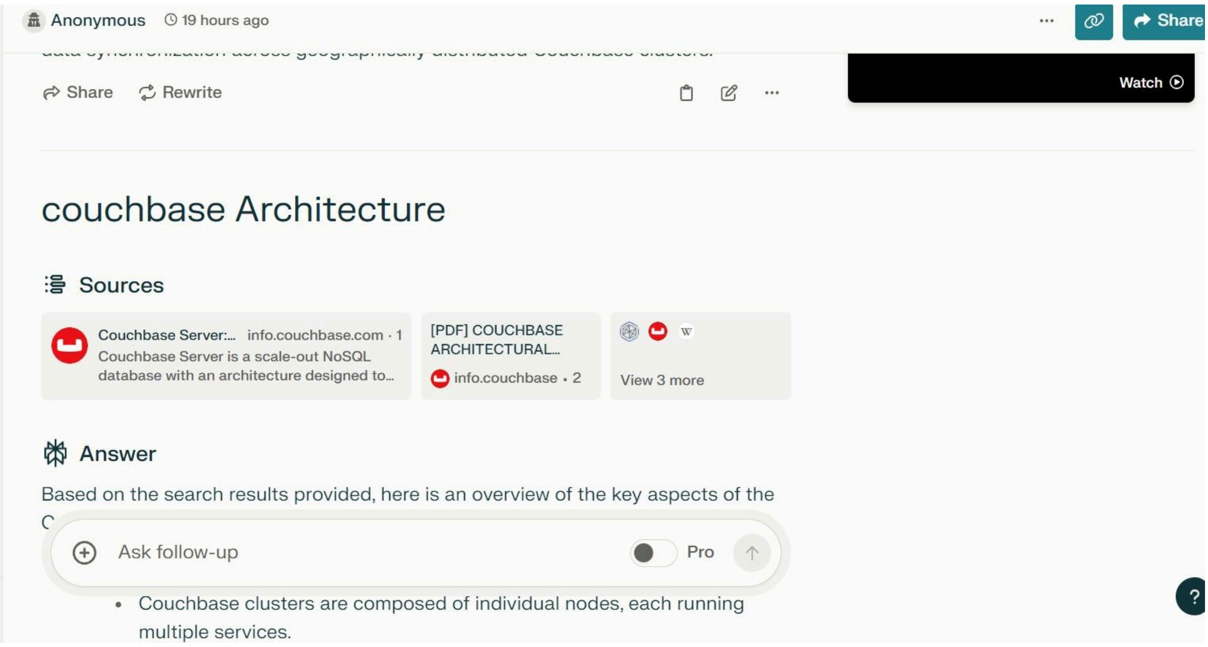
Data Model | Couchbase Docs docs.couchbase · 3

View 2 more

Answer

Based on the information provided in the search results, the key points about the data model used by Couchbase are:

1. Couchbase uses a flexible, document-oriented data model based on JSON. [2] [4]



data synchronization across geographically distributed Couchbase clusters.

couchbase Architecture

Sources

Couchbase Server:... info.couchbase.com · 1 Couchbase Server is a scale-out NoSQL database with an architecture designed to...

[PDF] COUCHBASE ARCHITECTURAL... info.couchbase · 2

View 3 more

Answer

Based on the search results provided, here is an overview of the key aspects of the C

- Couchbase clusters are composed of individual nodes, each running multiple services.