

1.Import a Bucket.

Ans. Done.

‘2. Write a query to get the sum of all distances where type=”route” for each airline id.

Ans. `select id , sum(distance) from `travel-sample` where type='route' group by id;`

BETA - PREVIEW MODE - UNSUPPORTED - NOT FOR USE IN PRODUCTION

activity help arun0701

Emp > Query

IMPORT EXPORT

Workbench

Dashboard

Servers

Buckets

Backup

XDCR

Security

Settings

Logs

Documents

Query

Indexes

Search

Analytics

Eventing

Views

Query Editor

history (36/36)

bucket.scope

1 `select id , sum(distance) from `travel-sample` where type='route' group by id;`

Execute Explain

External Query Advisor success just now | elapsed: 1.5s | execution: 1.5s | docs: 24024 | size: 1445796 bytes

format

Query Results

Table JSON Tree Plan Plan Text

```
60 {"id": 58933
61 },
62 {
63   "$1": 1624.9245723165843,
64   "id": 21243
65 },
66 {
67   "$1": 1350.675344615582,
68   "id": 64532
69 },
70 {
71   "$1": 11.711682700583648,
72   "id": 3045
73 },
74 {
75   "$1": 651.2512417553768,
76   "id": 58952
77 },
78 {
79   "$1": 2781.7893117770172,
80   "id": 27742
81 },
82 {
83   "$1": 308.3265938631413,
84   "id": 57692
```

Data Insights

travel-sample 6 collections

default scope 31591 docs

- stops = 0, type = "route" 75.6%
- type = "landmark" 14.2%
- type = "hotel" 2.6%
- type = "airline" 1.0%
- type = "airport" 6.6%

inventory scope

Refresh

3..Write queries to join(LEFT,RIGHT,INNER) type=”route”&“airline” and fetch the data whose sourceairport=”SFO”

Ans. `select * from (select * from `travel-sample` a left join `travel-sample` b on a.type = 'route' and b.type = 'airline') as tt where sourceairport='SFO' limit 10 ;`

`select * from (select * from `travel-sample` a right join `travel-sample` b on a.type = 'route' and b.type = 'airline') as tt where sourceairport='SFO' limit 10 ;`

`select * from (select * from `travel-sample` a inner join `travel-sample` b on a.type = 'route' and b.type = 'airline') as tt where sourceairport='SFO' limit 10 ;`

4.Write a mapreduce to get the number of all documents based on entities(type).

Ans. `function (doc,meta){
emit(meta.id,null);
}`

The screenshot shows the Couchbase Workbench interface. On the left is a sidebar with navigation links: Dashboard, Servers, Buckets, Backup, XDCR, Security, Settings, Logs, Documents, Query, Indexes, Search, Analytics, Eventing, and Views. The main area is titled 'Query Editor' and contains a query with a syntax error:

```
1 function (doc,meta){
2   emit(meta.id,null);
3 }
```

Buttons for 'Execute', 'Explain', and 'External Query Advisor' are visible. The 'Query Results' section shows the JSON output of the 'explain' command, indicating a syntax error at line 2, column 10.

```
1 [
2   {
3     "_sequence_num": 1,
4     "_sequence_query": "function (doc,meta){\nemit(meta.id,null);\n",
5     "_sequence_query_status": "executing",
6     "_sequence_result": {
7       "errors": [
8         {
9           "code": 3000,
10          "msg": "syntax error - at function",
11          "query_from_user": "explain function (doc,meta){\nemit(meta.id,null);\n"
12        }
13      ]
14    }
15  }
16 ]
```

On the right, the 'Data Insights' section shows a collection named 'travel-sample' with 6 collections. A 'Refresh' button is at the bottom right.

5.Run the below command to export the data from travel-sample bucket:

Ans . cbexport json -c couchbase://127.0.0.1 -u arun0701 -p qwerty \\
-b default -o /data/lines.json -f lines -t 4

D) Create a new Bucket “NewBucket” & import the data to it:

Ans. cbimport json -c couchbase://127.0.0.1 -u arun0701 -p qwerty -b NewBucket -f lines -d
file://travelSample.json -t 4

The screenshot shows the Couchbase Workbench interface with the 'NewBucket' bucket selected. The 'Document Editor' tab is active, showing a table of 10 results. The table has columns for 'id' and 'Document ID'. The results are as follows:

id	Document ID
10	optional...
10000	optional...
10001	optional...
10002	optional...
10003	optional...
10004	optional...
10005	optional...
10006	optional...
10007	optional...

The interface also shows a 'Retrieve Docs' button and a 'show range' dropdown set to 'NTQL WHERE'. The 'enable field editing' toggle is turned on.

