

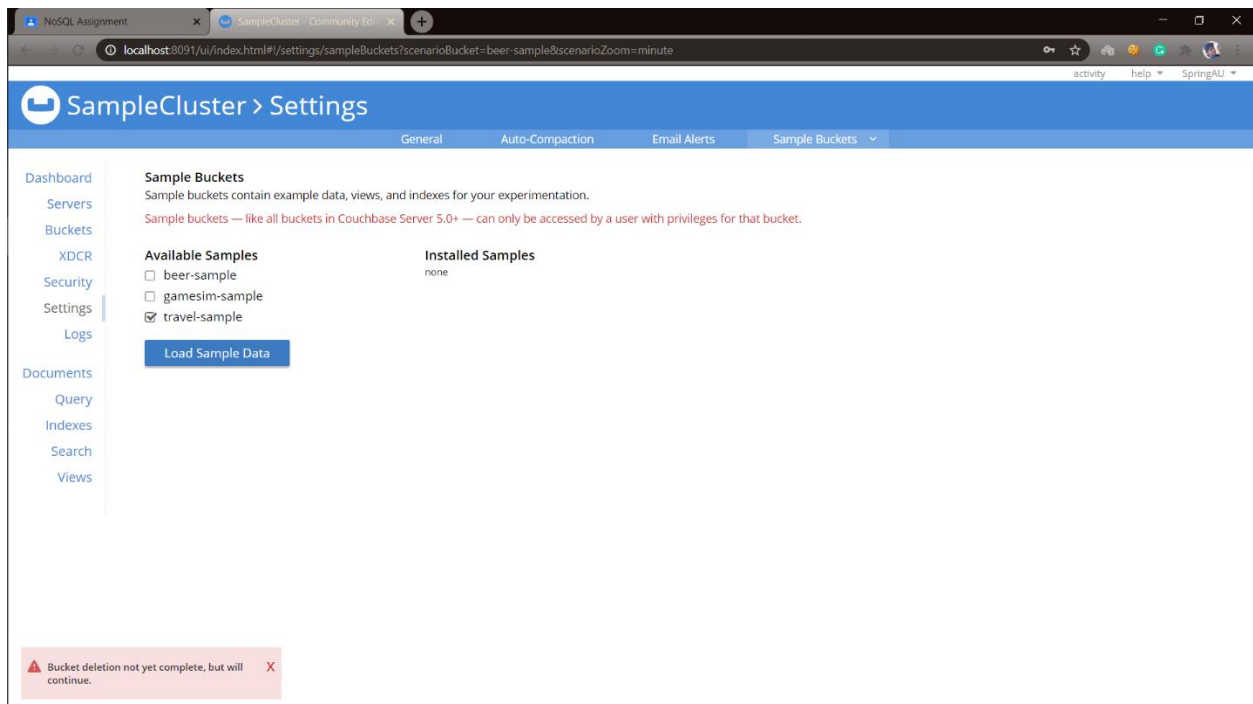
## Spring AU '21 – NoSQL – Morning Session

Name: Sheik Abudhahir K

Date: 08/01/2021

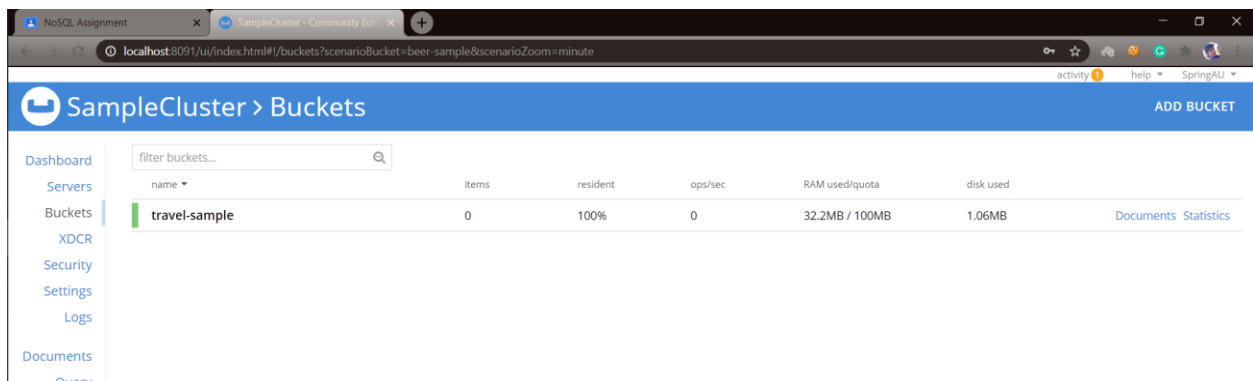
Title: NO SQL Assignment

Q1: Import bucket



The screenshot shows the 'SampleCluster > Settings' page in a web browser. The left sidebar contains links for Dashboard, Servers, Buckets, XDCR, Security, Settings (selected), Logs, Documents, Query, Indexes, Search, and Views. The main content area is titled 'Sample Buckets' and includes a description: 'Sample buckets contain example data, views, and indexes for your experimentation. Sample buckets — like all buckets in Couchbase Server 5.0+ — can only be accessed by a user with privileges for that bucket.' Below this, there are two columns: 'Available Samples' with checkboxes for 'beer-sample', 'gamesim-sample', and 'travel-sample' (which is checked), and 'Installed Samples' which currently shows 'none'. A 'Load Sample Data' button is located below the 'Available Samples' column. At the bottom of the page, there is a red error message: 'Bucket deletion not yet complete, but will continue.'

Bucket imported



The screenshot shows the 'SampleCluster > Buckets' page in a web browser. The left sidebar is the same as in the previous screenshot, with 'Settings' selected. The main content area is titled 'Buckets' and includes a search bar labeled 'filter buckets...'. Below the search bar is a table with the following columns: 'name', 'Items', 'resident', 'ops/sec', 'RAM used/quota', 'disk used', and links for 'Documents' and 'Statistics'. The table contains one entry: 'travel-sample' with 0 items, 100% resident, 0 ops/sec, 32.2MB / 100MB RAM used, and 1.06MB disk used. An 'ADD BUCKET' button is located in the top right corner of the main content area.

name	Items	resident	ops/sec	RAM used/quota	disk used	Documents	Statistics
travel-sample	0	100%	0	32.2MB / 100MB	1.06MB		

Q2: Write a query to get the sum of all distances where type="route" for each airline id.

The screenshot shows the SampleCluster Query Editor interface. The query editor contains the following SQL query:

```
1 select airlineid, sum(distance) as total_distance from `travel-sample` where type="route" group by airlineid
```

The query has been executed successfully. The query results are displayed in a JSON format, showing the sum of distances for each airline id where the type is "route".

**Query Results (JSON):**

```
1+ [
2+ {
3+   "airlineid": "airline_4305",
4+   "total_distance": 58875.41271258684
5+ },
6+ {
7+   "airlineid": "airline_3661",
8+   "total_distance": 6400.345938790037
9+ },
10+ {
11+   "airlineid": "airline_1909",
12+   "total_distance": 24868.667281884267
13+ },
14+ {
15+   "airlineid": "airline_596",
16+   "total_distance": 829222.0711321105
17+ },
18+ {
19+   "airlineid": "airline_16726",
20+   "total_distance": 42317.501052111955
}
```

**Data Insights:**

Queryable Buckets

travel-sample sampled 1000 of 31591

- type = "route" 96.5%
- type = "landmark" 3.0%
- free\_breakfast = true, free\_internet = true, type = "airport" 0.3%
- Indexes

Q3: Write queries to join (LEFT, RIGHT, INNER) type="route" & "airline" and fetch the data whose sourceairport="SFO"

Left Join – 249 records

The screenshot shows the SampleCluster Query Editor interface. The query editor contains the following SQL query:

```
1 SELECT * FROM `travel-sample` AS route LEFT JOIN `travel-sample` AS airline ON route.airlineid = META(airline).id
2 WHERE route.sourceairport="SFO"
```

The query has been executed successfully. The query results are displayed in a JSON format, showing the data for the 'route' table joined with the 'airline' table where the source airport is "SFO".

**Query Results (JSON):**

```
1+ [
2+ {
3+   "route": {
4+     "airline": "AI",
5+     "airlineid": "airline_218",
6+     "destinationairport": "HKG",
7+     "distance": 11128.182035009515,
8+     "equipment": "77W",
9+     "id": 10624,
10+     "schedule": [
11+       {
12+         "day": 0,
13+         "flight": "AI472",
14+         "utc": "09:43:00"
15+       },
16+       {
17+         "day": 1,
18+         "flight": "AI468",
19+         "utc": "10:19:00"
20+       }

```

**Data Insights:**

Queryable Buckets

travel-sample sampled 1000 of 31591

- stops = 0, type = "route" 80.0%
- type = "landmark" 12.7%
- type = "hotel" 2.2%
- type = "airport" 5.1%
- Indexes

## Inner Join – 183 records

The screenshot shows the SampleCluster Query Workbench interface. The Query Editor contains the following SQL query:

```
1 SELECT * FROM `travel-sample` AS route INNER JOIN `travel-sample` AS airline ON route.airlineid = META(airline).id
2 WHERE route.sourceairport='SFO'
```

The Query Results tab shows the first 20 records of the query results in JSON format. The results show a right join between the `travel-sample` and `airline` collections. The first record is:

```
{
  "airline": {
    "callsign": "JETBLUE",
    "country": "United States",
    "iata": "B6",
    "icao": "JBU",
    "id": "3029",
    "name": "JetBlue Airways",
    "type": "airline"
  },
  "route": {
    "airline": "B6",
    "airlineid": "airline_3029",
    "destinationairport": "AUS",
    "distance": "2416.0035377223094",
    "equipment": "320",
    "id": "14239",
    "schedule": [

```

The Data Insights panel on the right shows the distribution of the `type` field in the `airline` collection:

type	percentage
"route"	80.0%
"landmark"	12.7%
"hotel"	2.2%
"airport"	5.1%

## Right Join – 183 records

The screenshot shows the SampleCluster Query Workbench interface. The Query Editor contains the following SQL query:

```
1 SELECT * FROM `travel-sample` AS route RIGHT
2 JOIN `travel-sample` AS airline ON route.airlineid = META(airline).id
3 WHERE route.sourceairport='SFO'
```

The Query Results tab shows the first 20 records of the query results in JSON format. The results show a right join between the `travel-sample` and `airline` collections. The first record is:

```
{
  "airline": {
    "callsign": "CITRUS",
    "country": "United States",
    "iata": "FL",
    "icao": "TRS",
    "id": "1316",
    "name": "AirTran Airways",
    "type": "airline"
  },
  "route": {
    "airline": "FL",
    "airlineid": "airline_1316",
    "destinationairport": "ATL",
    "distance": "3434.7108309317646",
    "equipment": "736",
    "id": "25489",
    "schedule": [

```

The Data Insights panel on the right shows the distribution of the `type` field in the `airline` collection:

type	percentage
"route"	80.3%
"landmark"	11.3%
"hotel"	2.7%
"airline"	0.8%
"airport"	4.9%

Q4: Write a mapreduce to get the number of all documents based on entities(type).

Type = route

Full Cluster Data Set

The screenshot shows the Couchbase N1QL Index Advisor interface. On the left, a sidebar contains links for Logs, Documents, Query, Indexes, Search, and Views. The main area is titled 'View Index Code' and contains a MapReduce job configuration. The Map function is defined as:

```
1 function (doc, meta) {
2   if(doc.type='route')
3     emit(meta.id, null);
4 }
```

The Reduce function is built-in and counts the results. The Results section shows a filter: `?limit=6&state=false&connection_timeout=60000&inclusive_end=true&skip=0&full_set=true`. Below the filter, there are two tabs: 'Development Time Subset' and 'Full Cluster Data Set'. The 'Full Cluster Data Set' tab is selected, and the results are displayed in a table with columns 'Key' and 'Value'.

Key	Value
null	31591
undefined	

Development Time Subset

This screenshot is identical to the one above, but the 'Development Time Subset' tab is selected. The Results section shows the same filter: `?limit=6&state=false&connection_timeout=60000&inclusive_end=true&skip=0&full_set=true`. The results table shows a count of 38 for the 'null' key.

Key	Value
null	38
undefined	

Type = airline

Full Cluster Data Set

The screenshot shows the Couchbase N1QL Index Advisor interface. The left sidebar contains links for Logs, Documents, Query, Indexes, Search, and Views. The main area is titled 'View Index Code' and contains a Map function:

```
1 function (doc, meta) {  
2   if(doc.type='airline')  
3     emit(meta.id, null);  
4 }
```

Below the Map function is a 'Results' section with a filter: `?limit=6&stale=false&connection_timeout=60000&inclusive_end=true&skip=0&full_set=true`. The 'Development Time Subset' is selected, and the 'Full Cluster Data Set' is highlighted. The results table shows:

Key	Value
null undefined	31591

Development Time Subset

The screenshot shows the Couchbase N1QL Index Advisor interface. The left sidebar contains links for Logs, Documents, Query, Indexes, Search, and Views. The main area is titled 'View Index Code' and contains a Map function:

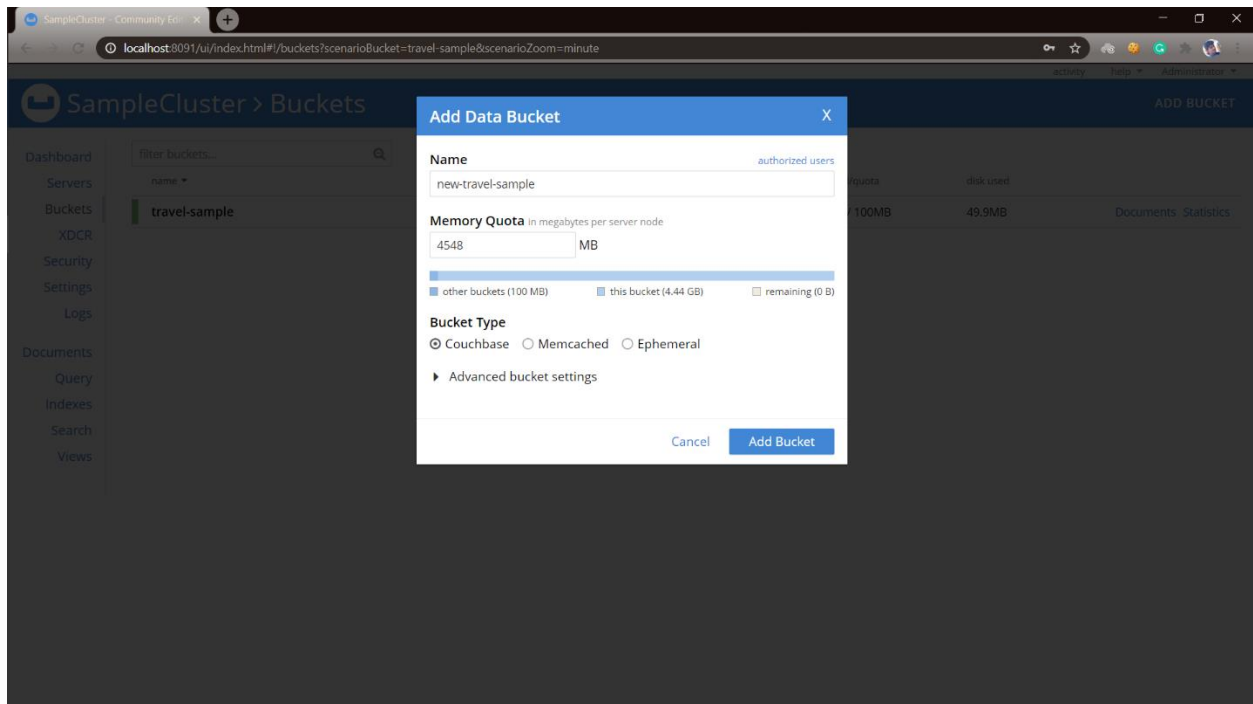
```
1 function (doc, meta) {  
2   if(doc.type='airline')  
3     emit(meta.id, null);  
4 }
```

Below the Map function is a 'Results' section with a filter: `?limit=6&stale=false&connection_timeout=60000&inclusive_end=true&skip=0&full_set=`. The 'Development Time Subset' is selected, and the 'Full Cluster Data Set' is highlighted. The results table shows:

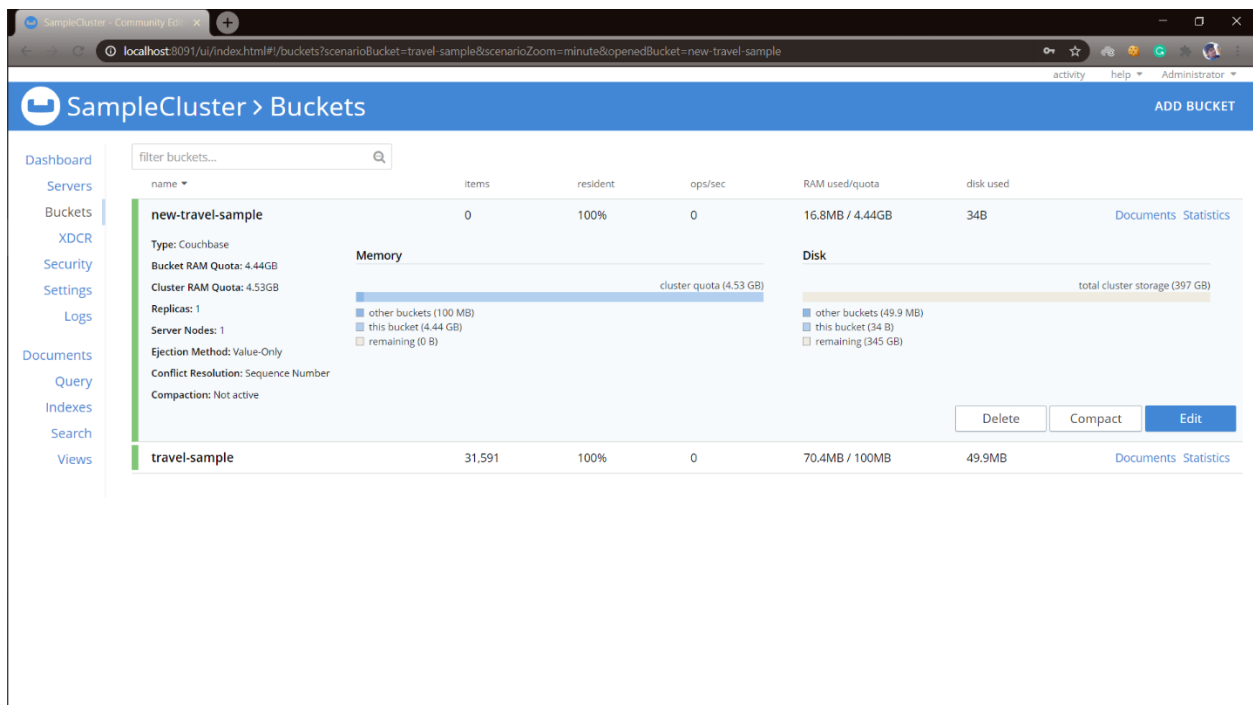
Key	Value
null undefined	38

## Q5: Export the travel-sample data from CLI

### New bucket creation



### Empty bucket



## Command to export the old bucket and import into the newly created bucket

```
C:\Windows\System32\cmd.exe

C:\Program Files\Couchbase\Server\bin>cbexport json -c couchbase://127.0.0.1 -u Administrator -p Sharp@007@ -b travel-sample -o c:/travelSample.json -f lines -t 4
Json export failed: open c:/travelSample.json: Access is denied.

C:\Program Files\Couchbase\Server\bin>cbexport json -c couchbase://127.0.0.1 -u Administrator -p Sharp@007@ -b travel-sample -o f:/travelSample.json -f lines -t 4
Json exported to 'f:/travelSample.json' successfully
Documents exported: 31591 Documents skipped: 0

C:\Program Files\Couchbase\Server\bin>cbimport json -c couchbase://127.0.0.1 -u Administrator -p Sharp@007@ -b new-travel-sample -f lines -d file://f:/travelSample.json -t 4 -g %id%
JSON 'file://f:/travelSample.json' imported to 'http://127.0.0.1:8091' successfully
Documents imported: 31591 Documents failed: 0

C:\Program Files\Couchbase\Server\bin>
```

## New bucket after import the travel-sample – 20.9 mb disk used – items imported

The screenshot shows the 'SampleCluster > Buckets' page in a web browser. The page displays a table of buckets with columns: name, items, resident, ops/sec, RAM used/quota, and disk used. The 'new-travel-sample' bucket is highlighted, showing 28,579 items, 100% resident, 0 ops/sec, 78.2MB / 4.44GB RAM used/quota, and 20.9MB disk used. Below the table, there are sections for 'Memory' and 'Disk' usage, each with a bar chart showing the bucket's usage relative to the cluster quota. The 'Memory' section shows the bucket's usage (4.44 GB) as a small portion of the 4.53 GB cluster quota. The 'Disk' section shows the bucket's usage (20.9 MB) as a small portion of the 397 GB total cluster storage. The 'travel-sample' bucket is also visible in the table below, showing 31,591 items, 100% resident, 0 ops/sec, 70.4MB / 100MB RAM used/quota, and 49.9MB disk used.

name	items	resident	ops/sec	RAM used/quota	disk used
<b>new-travel-sample</b>	28,579	100%	0	78.2MB / 4.44GB	20.9MB
<b>travel-sample</b>	31,591	100%	0	70.4MB / 100MB	49.9MB