

# Hands-on Lab: Working with Databases in Cloudant

Estimated time needed: **30** minutes

## Objectives

After completing this lab you will be able to:

- Create a database through the Cloudant dashboard
- Insert documents into your database to populate it
- Query documents with specific criteria
- Modify documents by updating and deleting them

## Prerequisites

In order to complete this lab, you will need to create an instance of Cloudant on IBM Cloud. If you haven't yet created one, you can create one by referring to the [Create an Instance of IBM Cloudant](#) lab.

Note: While working on this lab, you may be prompted to login when ever your session expires. Use your credentials to authenticate. This may happen when you step out or leave your Cloudant session unattended.

## Exercise 1 - Launch Cloudant Dashboard

Step 1: Click on [cloud.ibm.com/resources](#).

Step 2: Click on the Databases chevron.

Step 3: Click on your instance of Cloudant.

► Click here for Hint

Step 4: Click on Launch Dashboard.

The screenshot shows the IBM Cloud dashboard interface. At the top, there's a navigation bar with 'IBM Cloud', a search bar, and links for Catalog, Docs, Support, and Manage. Below the navigation is a resource list for 'mycloudant'. The 'Overview' tab is selected. On the left, there's a sidebar with 'Manage' (selected), 'Service credentials', 'Plan', and 'Connections' options. The main content area displays deployment details for the database, including CRN, Location (London), External Endpoint (https://4646e655-6aee-42d8-8b93-d2bde6e9a6ca-bluemix.cloudant.com), External Endpoint (preferred) (https://4646e655-6aee-42d8-8b93-d2bde6e9a6ca-bluemix.cloudantnosqldb.a), Authentication methods (IBM Cloud IAM and Cloudant credentials), Activity Tracker event types (Management dropdown), and Disk encryption (Yes, Automatically generated disk encryption key). A 'Capacity details' section is also partially visible at the bottom.

## Exercise 2 - Create Database

Step 1: On the dashboard click on Create Database.

Databases

Your Databases

Name	Size	# of Docs	Partitioned

Database name ▾

Step 2: Type **housing** as database name. Select ‘Non-partitioned’ and click on Create.

Create Database

Database name

Partitioning

Partitioned  Non-partitioned

What is a Partitioned Database?

Cancel Create

In a few moments the database will be created. and you will be taken to a page that looks like the one below.

housing

All Documents +

Query

Permissions

Changes

Design Documents +

Document ID Options { } JSON

No Documents Found

Showing 0 documents. Documents per page: 20

## Exercise 3 - Insert documents

Step 1: Click on **Create Document** to insert a document.

You will be presented the below screen, with a simple sample document.

housing > New Document

Create Document Cancel

```

1 {
2   "_id": "0ec51619d15d138e95f1693c1c085ac6"
3 }

```

{ } JSON

Cloudant uses `_id` key to uniquely identify a document. It is equivalent to the primary key in RDBMS. You can use your own custom values for `_id`.

Copy and paste the below json document and click on **Create Document** button, as show in the image below.

```
{
  "_id": "1",
  "square_feet": 1500,
  "bedrooms": 3,
  "price": 147890
}
```

housing > New Document

Create Document Cancel

```

1  {
2   "_id": "1",
3   "square_feet": 1500,
4   "bedrooms": 3,
5   "price": 147890
6 }
```

Once the document is created, Cloudant will take you to a page with the list of documents.

Click on the Table view button. You should see a screen similar to the one below.

housing

All Documents

	_id	bedrooms	price
<input type="checkbox"/>	1	3	147890

Document ID

Table Metadata { } JSON

Follow the above mentioned process and insert the below 4 documents. Ensure you only insert one document at a time.

```
{
  "_id": "2",
  "square_feet": 1800,
  "bedrooms": 3,
  "price": 182650
}
```

```
{
  "_id": "3",
  "square_feet": 2000,
  "bedrooms": 3,
  "price": 201260
}
```

```
{
  "_id": "4",
  "square_feet": 2200,
  "bedrooms": 4,
  "price": 234980
}
```

```
{  
  "_id":"5",  
  "square_feet":1100,  
  "bedrooms":2,  
  "price":114310  
}
```

After inserting the above documents your database should look like this.

The screenshot shows the Apache CouchDB Futon interface. The left sidebar has a back arrow, the title "housing", and a vertical ellipsis. The main area has a header with "Document ID" and a dropdown, "Options", "JSON", and a book icon. Below is a toolbar with "All Documents" (+), "Query", "Permissions", "Changes", "Design Documents" (+), and a "Create Doc" button. The central part is a table view with columns: \_id, bedrooms, price, and square\_feet. Each row contains a checkbox, a file icon, and the document values.

All Documents	+ Query Permissions Changes Design Documents		Document ID	▼	Options	{ } JSON	Book	
			<input type="checkbox"/>	Table	Metadata	{ } JSON	<input type="button"/>	Create Doc
			<input type="checkbox"/>	<input type="button"/> 1	3	147890	1500	
			<input type="checkbox"/>	<input type="button"/> 2	3	182650	1800	
			<input type="checkbox"/>	<input type="button"/> 3	3	201260	2000	
			<input type="checkbox"/>	<input type="button"/> 4	4	234980	2200	
			<input type="checkbox"/>	<input type="button"/> 5	2	114310	1100	

Cloudant is a NoSQL database. It is a schema less database. All documents in a database need not have the same schema.

Let us insert two documents that have additional keys, compared to the previously inserted documents.

```
{  
    "_id": "6",  
    "square_feet": 1400,  
    "bedrooms": 3,  
    "price": 123140,  
    "type": "apartment",  
    "floor": 5  
}
```

```
{  
    "_id":"7",  
    "square_feet":3400,  
    "bedrooms":4,  
    "price":342720,  
    "type":"villa",  
    "car_parks":3  
}
```

After inserting the above documents your database should now look like this.

The screenshot shows the Apache CouchDB Futon interface. The left sidebar has 'All Documents' selected. The main area displays a table with the following data:

	_id	bedrooms	price	square_feet	type
<input type="checkbox"/>	1	3	147890	1500	
<input type="checkbox"/>	2	3	182650	1800	
<input type="checkbox"/>	3	3	201260	2000	
<input type="checkbox"/>	4	4	234980	2200	
<input type="checkbox"/>	5	2	114310	1100	
<input type="checkbox"/>	6	3	123140	1400	apartment
<input type="checkbox"/>	7	4	342720	3400	villa

## Exercise 4 - Query documents

Click on Query as shown in the image below.

The screenshot shows the Apache CouchDB Futon interface. A blue arrow points to the 'Query' link in the left sidebar. The main area displays a table with the same housing data as the previous screenshot.

	_id	bedrooms	price	square_feet	type
<input type="checkbox"/>	1	3	147890	1500	
<input type="checkbox"/>	2	3	182650	1800	
<input type="checkbox"/>	3	3	201260	2000	
<input type="checkbox"/>	4	4	234980	2200	
<input type="checkbox"/>	5	2	114310	1100	
<input type="checkbox"/>	6	3	123140	1400	apartment
<input type="checkbox"/>	7	4	342720	3400	villa

You will see a screen like this.

housing > Cloudant Query

Query history

Cloudant Query

```
1: {  
2:   "selector": {  
3:     "_id": {  
4:       "$gt": "0"  
5:     }  
6:   },  
7:   "fields": [  
8:     "_id",  
9:     "_rev"  
10:  ],  
11:  "sort": [  
12:    {  
13:      "_id": "asc"  
14:    }  
15:  ]  
16:}
```

Run Query Explain manage indexes



No Documents Found

Replace the default query with the one given below, and click on the Run Query button.

```
{  
  "selector": {}  
}
```

housing > Cloudant Query

Query history

Cloudant Query

```
1:  
2:  
3: {  
4:   "selector": {}  
5: }  
6:  
7:
```

Run Query Explain manage indexes

You should see an output like this.

The screenshot shows the Cloudant Query interface. On the left, there's a sidebar with various icons. The main area has a title "housing > Cloudant Query". Below the title is a dropdown menu set to "Query history". A "Cloudant Query" section contains a code editor with the following JSON:

```
{
  "selector": {}
}
```

Below the code editor are two buttons: "Run Query" and "Explain". To the right of the code editor is a link "manage indexes". A status message "Executed in 3 ms" is displayed. At the top right of the main area are three tabs: "Table" (selected), "{} JSON", and "T". The "Table" tab displays a table with the following data:

_id	bedrooms	price	s
1	3	147890	15
2	3	182650	18
3	3	201260	20
4	4	234980	22
5	2	114310	11
6	3	123140	14
7	4	342720	34

Try out these Cloudant queries.

Select all fields in all documents

```
{
  "selector": {}
}
```

Select all fields in all documents with \_id greater than 4

```
{
  "selector": {
    "_id": {
      "$gt": "4"
    }
  }
}
```

Select all fields in all documents with \_id less than 4

```
{
  "selector": {
    "_id": {
      "$lt": "4"
    }
  }
}
```

Select the fields \_id, square\_feet and price in all documents

```
{
  "selector": {},
```

```
"fields": [
    "_id",
    "price",
    "square_feet"
]
```

Select the fields \_id, square\_feet and price in documents with \_id less than 4

```
{
  "selector": {
    "_id": {
      "$lt": "4"
    }
  },
  "fields": [
    "_id",
    "price",
    "square_feet"
]
}
```

Select the fields \_id, bedrooms and price in documents with \_id greater than 2 and sort by \_id ascending

```
{
  "selector": {
    "_id": {
      "$gt": "2"
    }
  },
  "fields": [
    "_id",
    "price",
    "bedrooms"
  ],
  "sort": [
    {
      "_id": "asc"
    }
  ]
}
```

Select the fields \_id, bedrooms and price in documents with \_id greater than 2 and sort by \_id descending

```
{
  "selector": {
    "_id": {
      "$gt": "2"
    }
  },
  "fields": [
    "_id",
    "price",
    "bedrooms"
  ],
  "sort": [
    {
      "_id": "desc"
    }
  ]
}
```

## Exercise 5 - Update documents

Click on the database name housing as shown in the image below.

The screenshot shows the Cloudant Query interface. On the left, there's a sidebar with various icons. In the center, the database name "housing" is highlighted in blue with a large blue arrow pointing to it. Below it is the title "Cloudant Query". Under "Cloudant Query", there's a code editor containing a simple selector: "1 + [{}], 2 | {\"selector\": {}}, 3 }". Below the code editor are two buttons: "Run Query" and "Explain". A note says "Executed in 3 ms". To the right of the code editor is a table view with columns: \_id, bedrooms, price, and square\_. The table contains 7 rows of data. At the top of the table view, there are buttons for "Table", "{} JSON", and a grid icon. A link "manage indexes" is also visible.

_id	bedrooms	price	square_
1	3	147890	1500
2	3	182650	1800
3	3	201260	2000
4	4	234980	2200
5	2	114310	1100
6	3	123140	1400
7	4	342720	3400

You will see a screen as in the image below.

The screenshot shows the Cloudant Database interface. On the left, there's a sidebar with various icons. In the center, the database name "housing" is highlighted in blue with a large blue arrow pointing to it. Below it is the title "All Documents". Under "All Documents", there are four options: "Query", "Permissions", "Changes", and "Design Documents". To the right of the sidebar is a table view with columns: \_id, bedrooms, price, and square\_. The table contains 7 rows of data. At the top of the table view, there are buttons for "Table", "Metadata", "{} JSON", and a grid icon. A link "Document ID" is also visible.

_id	bedrooms	price	square_
1	3	147890	1500
2	3	182650	1800
3	3	201260	2000
4	4	234980	2200
5	2	114310	1100
6	3	123140	1400
7	4	342720	3400

Click on the document with \_id 7.

The screenshot shows the Apache CouchDB Futon interface. On the left is a sidebar with various icons for database management. The main area has a title bar "housing" and a "Document ID" field. Below the title is a navigation bar with "All Documents" and a "+" button. Underneath are sections for "Query", "Permissions", "Changes", and "Design Documents". The main content area displays a table with columns: "\_id", "bedrooms", "price", and "square\_feet". The table contains 7 rows of data. Row 7 is highlighted with a blue border.

	_id	bedrooms	price	square_feet
<input type="checkbox"/>	1	3	147890	1500
<input type="checkbox"/>	2	3	182650	1800
<input type="checkbox"/>	3	3	201260	2000
<input type="checkbox"/>	4	4	234980	2200
<input type="checkbox"/>	5	2	114310	1100
<input type="checkbox"/>	6	3	123140	1400
<input type="checkbox"/>	7	4	342720	3400

The document will open up like this.

The screenshot shows the Apache CouchDB Futon interface with the document "7" selected. The top bar says "housing > 7". Below it is a toolbar with "Save Changes" (checked) and "Cancel". The main area is a code editor showing the document's JSON structure. The "car\_parks" value is currently set to 3. A blue arrow points to the "car\_parks" field.

```

1 [
2   "_id": "7",
3   "_rev": "1-6c4aba2b36f6cf01bee45f5b2220a578",
4   "square_feet": 3400,
5   "bedrooms": 4,
6   "price": 342720,
7   "type": "villa",
8   "car_parks": 3
9 ]

```

Change the number of car\_parks to 4 and add facing key with value East, as shown in the image below. Click Save Changes to save the document.

The screenshot shows the Apache CouchDB Futon interface with the document "7" selected. The top bar says "housing > 7". Below it is a toolbar with "Save Changes" (checked) and "Cancel". The main area is a code editor showing the updated document's JSON structure. The "car\_parks" value is now 4, and a new "facing" field with the value "East" has been added. A blue arrow points to the "facing" field.

```

1 [
2   "_id": "7",
3   "_rev": "1-6c4aba2b36f6cf01bee45f5b2220a578",
4   "square_feet": 3400,
5   "bedrooms": 4,
6   "price": 342720,
7   "type": "villa",
8   "car_parks": 4,
9   "facing": "East"
10 ]

```

## Exercise 6 - Delete documents

Select the document you wish to delete and click on the delete icon as shown in the image below.

The screenshot shows the Apache CouchDB Futon interface. On the left, there's a sidebar with various icons. In the main area, the database name 'housing' is at the top. Below it is a table with columns: '\_id', 'bedrooms', 'price', and 'square\_'. There are seven rows of data. Row 5 has a checked checkbox and a delete icon. A blue arrow points from the 'Design Documents' section in the sidebar to the delete icon in the header toolbar. Another blue arrow points from the 'Design Documents' section to the row for '\_id': 5.

_id	bedrooms	price	square_
1	3	147890	1500
2	3	182650	1800
3	3	201260	2000
4	4	234980	2200
5	2	114310	1100
6	3	123140	1400
7	4	342720	3400

You will get a pop up asking “Are you sure you want to delete this doc?”

Click OK.

## Practice exercises

1. Create a database named **iamonds**.

- Click here for Hint
- Click here for Solution

2. Insert the below documents into the **iamonds** database.

```
{
  "_id": "1",
  "carat": 0.31,
  "cut": "Ideal",
  "color": "J",
  "clarity": "SI2",
  "depth": 62.2,
  "table": 54,
  "price": 339
}

{
  "_id": "2",
  "carat": 0.2,
  "cut": "Premium",
  "color": "E",
  "clarity": "SI2",
  "depth": 60.2,
  "table": 62,
  "price": 351
}

{
  "_id": "3",
  "carat": 0.32,
  "cut": "Premium",
  "color": "E",
  "clarity": "I1",
  "depth": 60.9,
  "table": 58,
  "price": 342
}

{
  "_id": "4",
  "carat": 0.3,
  "cut": "Good",
  "color": "J",
  "clarity": "SI1",
  "depth": 63.4,
  "table": 54,
  "price": 349
}

{
  "_id": "5",
  "carat": 0.3,
  "cut": "Good",
  "color": "J",
  "clarity": "SI1",
  "depth": 63.8,
  "table": 56,
  "price": 347
}
```

```

}
{
  "_id": "6",
  "carat": 0.3,
  "cut": "Very Good",
  "color": "J",
  "clarity": "SI1",
  "depth": 62.7,
  "table": 59,
  "price": 349
}
{
  "_id": "7",
  "carat": 0.3,
  "cut": "Good",
  "color": "I",
  "clarity": "SI2",
  "depth": 63.3,
  "table": 56,
  "price": 343
}
{
  "_id": "8",
  "carat": 0.23,
  "cut": "Very Good",
  "color": "E",
  "clarity": "VS2",
  "depth": 63.8,
  "table": 55,
  "price": 339
}
{
  "_id": "9",
  "carat": 0.23,
  "cut": "Very Good",
  "color": "H",
  "clarity": "VS1",
  "depth": 61,
  "table": 57,
  "price": 323
}
{
  "_id": "10",
  "carat": 0.31,
  "cut": "Very Good",
  "color": "J",
  "clarity": "SI1",
  "depth": 59.4,
  "table": 62,
  "price": 346
}

```

- [Click here for Hint](#)
- [Click here for Solution](#)

*3. Write a query to fetch all documents*

- [Click here for Hint](#)
- [Click here for Solution](#)

*4. Write a query to fetch all documents with \_id greater than 2*

- [Click here for Hint](#)
- [Click here for Solution](#)

*5. Write a query to fetch all documents with \_id less than 4*

- [Click here for Hint](#)
- [Click here for Solution](#)

*6. Set the price of the diamond with \_id 7 to 352*

- [Click here for Hint](#)
- [Click here for Solution](#)

*7. Delete the document with \_id 3*

- [Click here for Hint](#)
- [Click here for Solution](#)

## Authors

Ramesh Sannareddy

## **Other Contributors**

Rav Ahuja



**Skills Network**