

## Cloudant DB- Create Database

Team id :	PNT2022TMID23090
Project Name :	VirtualEye-Life Guard for swimming Pools to Detect Active Drowning

- In order to manage a connection from a local system you must first initialize the connection by constructing a Cloudant client. We need to import the cloudant library.

```
from cloudant.client import Cloudant
```

- IBM Cloud Identity & Access Management enables you to securely authenticate users and control access to all cloud resources consistently in the IBM Bluemix Cloud Platform.

```
# Authenticate using an IAM API key
client = Cloudant.iam('username', 'apikey', connect=True)
```

In the above `cloudant.iam()` method we have to give `username` & `apikey` to build the connection with cloudant DB.

The screenshot displays the IBM Cloud console interface. At the top, there's a navigation bar with 'IBM Cloud' and a search bar. Below this, the 'Resource list' shows 'Cloudant-66' as an active resource. The left sidebar contains 'Manage', 'Service credentials', 'Plan', and 'Connections'. The main area is titled 'Service credentials' and includes a description: 'You can generate a new set of credentials for cases where you want to manually connect an app or external consumer to an IBM Cloud service. [Learn more](#)'. A table lists the credentials, with one entry 'Service credentials-1' created on '2022-11-09 6:45 PM'. A 'New credential' button is in the top right. The details of the selected credential are shown in a JSON format, including fields like 'apikey', 'host', 'iam\_apikey\_description', 'iam\_apikey\_name', 'iam\_role\_crn', 'iam\_serviceid\_crn', 'url', and 'username'.

- Once a connection is established you can then create a database, open an existing database.

- Create a database as my\_database.

```
# Create a database using an initialized client
my_database = client.create_database('my_database')
```

## Creating Database

API module that maps to a Cloudant or CouchDB database instance.

### **cloudant.database.CloudantDatabase**

Bases: **cloudant.database.CouchDatabase**

Encapsulates a Cloudant database. A CloudantDatabase object is instantiated with a reference to a client/session. It supports accessing the documents, and various database features such as the document indexes, changes feed, design documents, etc.

#### **Parameters:**

- **client** (*Cloudant*) – Client instance used by the database.
- **database\_name** (*str*) – Database name used to reference the database.
- **fetch\_limit** (*int*) – Optional fetch limit used to set the max number of documents to fetch per query during iteration cycles. Defaults to 100.
- **partitioned** (*bool*) – Create as a partitioned database. Defaults to **False**.

**get\_partitioned\_search\_result(partition\_key, ddoc\_id, index\_name, \*\*query\_params)**

Retrieves the raw JSON content from the remote database based on the partitioned search index on the server, using the query\_params provided as query parameters.

See **get\_search\_result()** method for further details.