

# Project Development Phase

## SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES

TEAM ID: PNT2022TMID39310

### TEAM MEMBERS

ROLE	TEAM MEMBERS NAME	ROLL NO
TEAM LEADER	RUBESH .S	(422619104035)
TEAM MEMBER 1	ABIRAMI .S	(422619104002)
TEAM MEMBER 2	HARITHA .S	(422619104016)
TEAM MEMBER 3	TAMIZHSELVAN .D	(422619104044)

## Project Development - Delivery Of Sprint-2

<b>Sprint</b>	<b>Functional Requirement (Epic)</b>	<b>User Story Number</b>	<b>User Story / Task</b>	<b>Story Points</b>	<b>Priority</b>	<b>Team Members</b>
Sprint-2	Create And Cofigure IBM Cloud Services	USN- 3	Create IBM Watson IoT Platform And Device	7	High	S.Rubesh
Sprint-2	Create And Cofigure IBM Cloud Services	USN- 4	Create Node-RED Service	7	High	S.Abirami S.Haritha
Sprint-2	Create And Cofigure IBM Cloud Services	USN-5	Create A Database In Cloudant DB	6	High	D.Tamizhselvan

## Delivery


<b>Sprint</b>	<b>Total Story Points</b>	<b>Duration</b>	<b>Sprint Start Date</b>	<b>Sprint End Date (Planned)</b>	<b>Story Points Completed (as on Planned End Date)</b>	<b>Sprint Release Date (Actual)</b>
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022

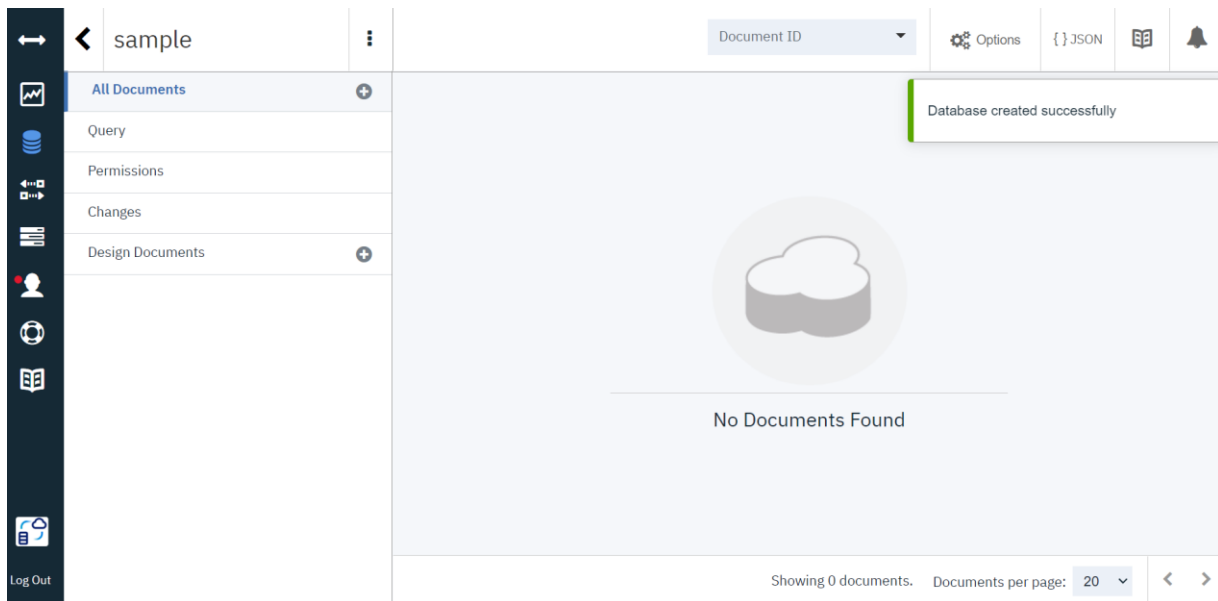
# Create A Database In Cloudant DB

**Task Assigned:** D.Tamizhselvan

**Task Started On:** 02 Nov 2022

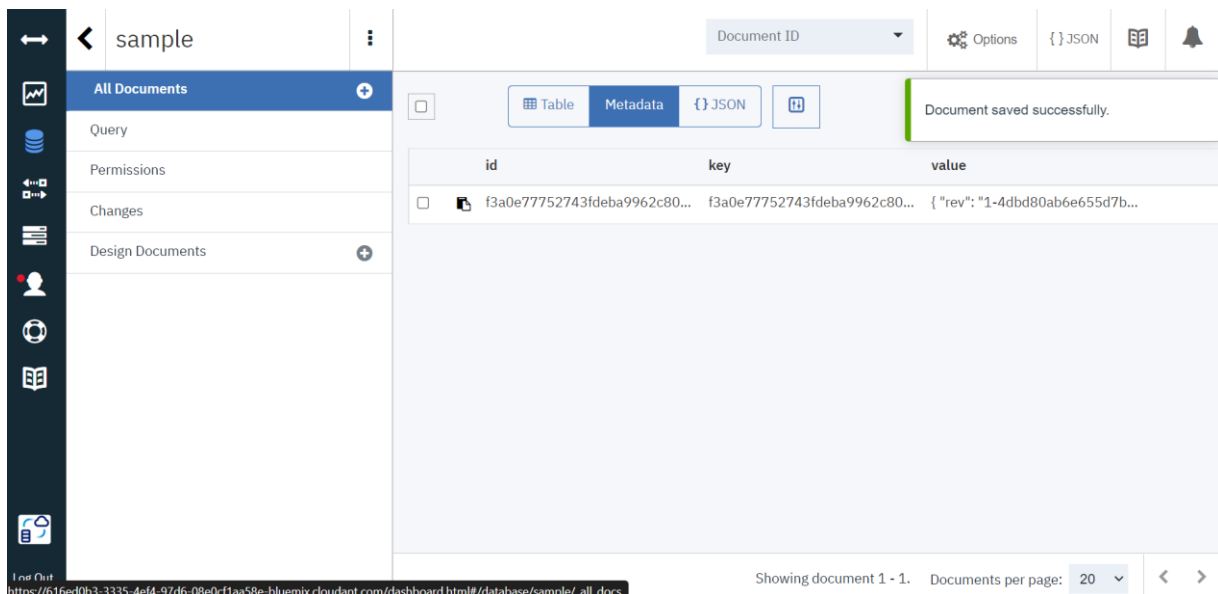
**Task Completion Date:** 04 Nov 2022

 **Launched the cloudant DB and created a database to store the location data**



The screenshot shows the Cloudant dashboard for a database named 'sample'. The left sidebar contains navigation links: All Documents, Query, Permissions, Changes, Design Documents, and Log Out. The main area displays a message 'Database created successfully' and a large cloud icon with the text 'No Documents Found'. The bottom status bar indicates 'Showing 0 documents.' and 'Documents per page: 20'.

 **Document Saved Successfully**



The screenshot shows the Cloudant dashboard for the 'sample' database. The left sidebar is the same as the previous screenshot. The main area displays a message 'Document saved successfully.' and a table view of the document. The table has three columns: id, key, and value. The first row shows the document's ID, key, and value.

id	key	value
f3a0e77752743fdeba9962c80...	f3a0e77752743fdeba9962c80...	{ "rev": "1-4dbd80ab6e655d7b..." }

The bottom status bar indicates 'Showing document 1 - 1.' and 'Documents per page: 20'.

# Integrating the Services

The screenshot shows the IBM Cloudant documentation page for the 'Introduction' section. The left sidebar contains a navigation menu with 'Overview' expanded, showing 'Introduction', 'Endpoint URLs', 'Authentication', 'Auditing', 'Event tracking', 'Error handling', 'Additional headers', 'Rate limits', 'Related APIs', and 'Logging'. The main content area has the title 'Introduction' and a sub-header 'Last updated: 2022-10-24'. The text describes IBM Cloudant as a document-oriented database service (DBaaS) that stores data as documents in JSON format. It mentions scalability, high availability, and durability. It also lists indexing options like MapReduce, IBM Cloudant Query, full-text indexing, and geospatial indexing. A note states that detailed documentation is available as a 'Getting started tutorial', 'API overview documentation, tutorials, and guides'. Below this, it says 'This documentation describes the Python SDK and examples. To see usage information and examples in your preferred SDK, select the language tab in the right pane.' The right pane shows a dark-themed code editor with the title 'Installation' and the command 'pip3 install ibmcloudant'. Below that, it says 'GitHub' and provides a link to the GitHub repository: 'https://github.com/ibm/cloudant-python-sdk'.

IBM Cloudant

Overview

Introduction

Endpoint URLs

Authentication

Auditing

Event tracking

Error handling

Additional headers

Rate limits

Related APIs

Logging

Methods

Waiting for cl...

## Introduction

Last updated: 2022-10-24

IBM® Cloudant® for IBM Cloud® is a document-oriented database as a service (DBaaS). It stores data as documents in JSON format. It is built with scalability, high availability, and durability in mind. It comes with a wide variety of indexing options that include MapReduce, IBM Cloudant Query, full-text indexing, and geospatial indexing. The replication capabilities make it easy to keep data in sync between database clusters, desktop PCs, and mobile devices.

Detailed documentation is also available such as a [Getting started tutorial](#), [API overview documentation, tutorials, and guides](#).

This documentation describes the Python SDK and examples. To see usage information and examples in your preferred SDK, select the language tab in the right pane.

## Endpoint URLs

The IBM Cloudant API uses an instance-specific endpoint URL for all regions. You can find your external endpoint by following these steps:

1. Go to the IBM Cloud dashboard and open an instance.

2. Click the Service credentials tab

The code examples on this tab use the IBM Cloudant SDK for Python.

### Installation

```
pip3 install ibmcloudant
```

GitHub

<https://github.com/ibm/cloudant-python-sdk>

The screenshot shows the IBM Cloudant documentation page for the 'Programmatic authentication' section. The left sidebar contains a navigation menu with 'Overview' expanded, showing 'Introduction', 'Endpoint URLs', 'Authentication', 'Auditing', 'Event tracking', 'Error handling', and 'Additional headers'. The 'Authentication' section is expanded, showing 'Security scheme', 'Authentication with external configuration', and 'Programmatic authentication'. The main content area has the title 'Programmatic authentication' and a sub-header 'In this scenario, authentication is configured by constructing an authenticator instance, supplying the configuration attributes programmatically, and then passing this instance to a client constructor.' Below this, there is a tip: 'Tip: If you are using the IBM Cloud App Service, IBM Cloud® Continuous Delivery or IBM Cloud starter kits then you can programmatically configure your SDK using the IBMCloudEnv tool to obtain the configuration information from bound services. The IBMCloudEnv tool is available for Go, Java&trade (Spring), Node.js, and Python.' The right pane shows a dark-themed code editor with the title 'Python' and the code for 'SDK managing the IAM token.' and 'SDK managing session cookie.' The code includes imports for 'ibmcloudant.cloudant\_v1' and 'ibm\_cloud\_sdk\_core.authenticators', and defines an 'IAMAuthenticator' instance. The code also shows the creation of a 'CloudantV1' instance and setting the service URL. The right pane also has a 'Feedback' button and a 'Go' button.

IBM Cloudant

Overview

Introduction

Endpoint URLs

Authentication

Auditing

Event tracking

Error handling

Additional headers

Programmatic authentication

## Programmatic authentication

In this scenario, authentication is configured by constructing an authenticator instance, supplying the configuration attributes programmatically, and then passing this instance to a client constructor.

☒ **Tip:** If you are using the IBM Cloud App Service, IBM Cloud® Continuous Delivery or IBM Cloud starter kits then you can programmatically configure your SDK using the `IBMCloudEnv` tool to obtain the configuration information from bound services. The `IBMCloudEnv` tool is available for [Go](#), [Java&trade \(Spring\)](#), [Node.js](#), and [Python](#).

### Python

SDK managing the IAM token.

```
from ibmcloudant.cloudant_v1 import CloudantV1
from ibm_cloud_sdk_core.authenticators import IAMAuthenticator

authenticator = IAMAuthenticator('{apikey}')

service = CloudantV1(authenticator=authenticator)

service.set_service_url('{url}')
```

SDK managing session cookie.

```
from ibmcloudant.cloudant_v1 import CloudantV1
```

Feedback

Go