

Project Development Phase
Delivery of Sprint-3

Date	24 November 2022
Team ID	PNT2022TMID25192
Project Name	Inventory Management System For Retailer

Creating IBM Db2 database and chat-box using IBM Watson Assistant:

Creating IBM Db2 Database and Connecting it:

Step 1:

Go to IBM cloud resource list and click on database.

Step 2:

Create a database and use the service credentials on your python flask to connect to IBM Db2 database services.

Step 3:

Click on 'Go to UI' and click 'Data' on the left side.

Step 4:

Click tables and select the name of your database.

Step 5:

Create new table according to the database you need.

Step 6:

Verify it is working.

Creating Chat-Box using IBM Watson Assistant:

Step 1:

Go to IBM cloud resource list and click on IBM Watson Assistant.

Step 2:

Click on launch Watson Assistant.

Step 3:

Build your virtual assistant.

Step 4:

Add the script to your python file.

Step 5:

Verify on our application page.

IBM Db2 Database Output:

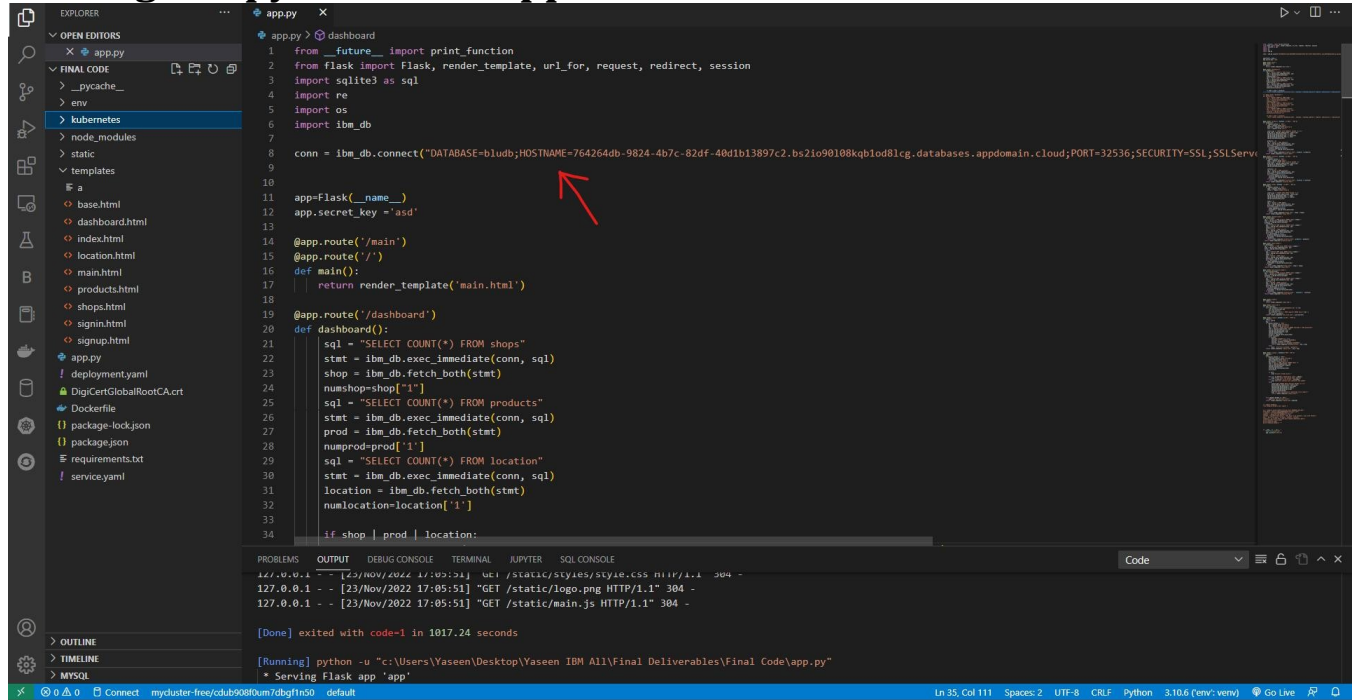
IBM Resource List Page:

Name	Group	Location	Product	Status	Tags
Filter by name or IP address...					
Filter by group or org...					
Filter...					
Filter...					
Filter...					
Containers (2)					
Networking (0)					
Storage (1)					
AI / Machine Learning (1)					
Analytics (0)					
Blockchain (0)					
Databases (1)					
ibm-project-db2oncloud-1669141780805	Default	London	Db2	Active	-
Developer tools (3+)					
Logging and monitoring (2+)					
Migration (0)					
Integration (0+)					
Internet of Things (0)					
Security (0)					
Mobile (0)					
Other (0+)					

Db2 Service Credentials:

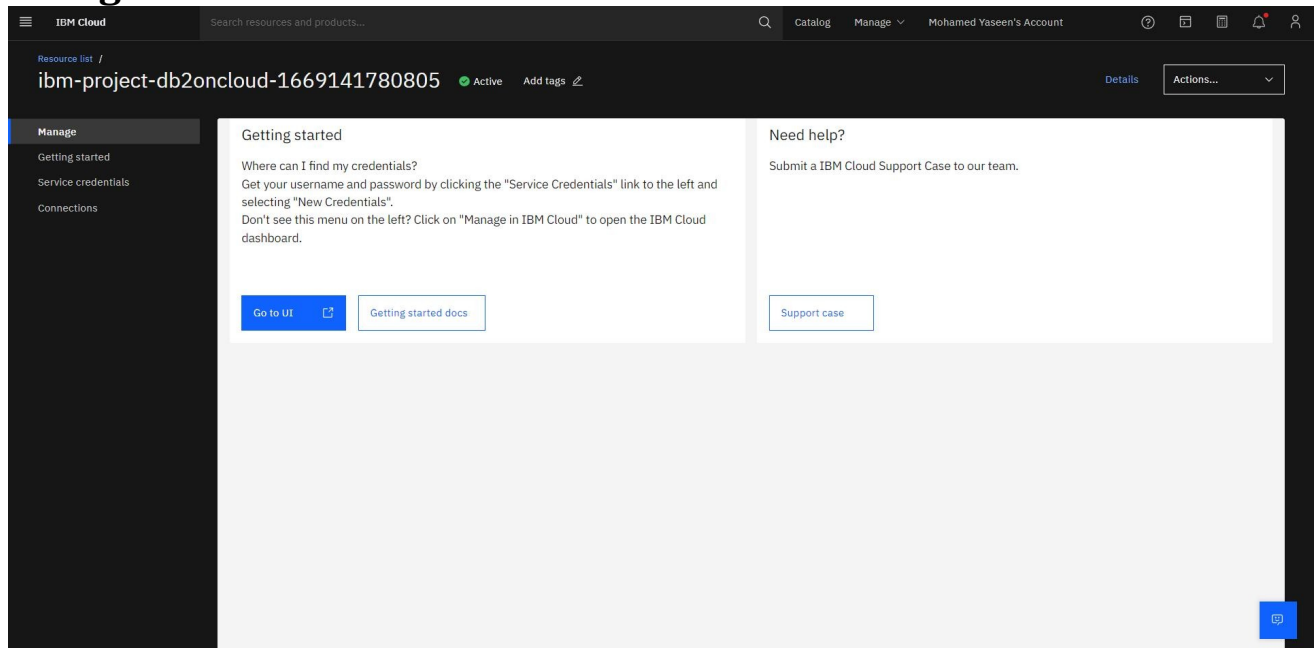
Key name	Date created
dc2e89f4-baeb-487a-bbe7-1d855e286bac	2022-11-23 12:00 AM

Linking our python flask application with IBM Db2:



```
1 from __future__ import print_function
2 from flask import Flask, render_template, url_for, request, redirect, session
3 import sqlite3 as sql
4 import re
5 import os
6 import ibm_db
7
8 conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=764264db-9824-4b7c-82df-48d1b13897c2.bs2io90108kqblod81cg.databases.appdomain.cloud;PORT=32536;SECURITY=SSL;SSLServerCertificate=ca.crt", "", "")
9
10
11 app=Flask(__name__)
12 app.secret_key = "asd"
13
14 @app.route('/main')
15 @app.route('/')
16 def main():
17     return render_template('main.html')
18
19 @app.route('/dashboard')
20 def dashboard():
21     sql = "SELECT COUNT(*) FROM shops"
22     stmt = ibm_db.exec_immediate(conn, sql)
23     shop = ibm_db.fetch_both(stmt)
24     numshop=shop["1"]
25     sql = "SELECT COUNT(*) FROM products"
26     stmt = ibm_db.exec_immediate(conn, sql)
27     prod = ibm_db.fetch_both(stmt)
28     numprod=prod["1"]
29     sql = "SELECT COUNT(*) FROM location"
30     stmt = ibm_db.exec_immediate(conn, sql)
31     location = ibm_db.fetch_both(stmt)
32     numlocation=location["1"]
33
34     if shop | prod | location:
```

Going to IBM Db2 database UI:



IBM Cloud

Search resources and products...

Resource list /

ibm-project-db2oncloud-1669141780805 Active Add tags

Details Actions...

Manage

Getting started

Service credentials

Connections

Getting started

Where can I find my credentials?

Get your username and password by clicking the "Service Credentials" link to the left and selecting "New Credentials".

Don't see this menu on the left? Click on "Manage in IBM Cloud" to open the IBM Cloud dashboard.

[Go to UI](#) [Getting started docs](#)

Need help?

Submit a IBM Cloud Support Case to our team.

[Support case](#)


IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

SQL

My Computer
A single delimited text file (CSV) without header row.
Amazon S3
Cloud Object Storage

File selection



Drag a file here or browse files

Next

IBM Db2 on Cloud

Load DataLoad History**Tables**ViewsIndexesAliasesMQTsSequencesApplication objects

Find schemas or tables

Refresh

Schemas

<input checked="" type="checkbox"/> Name	Type	Tables
<input checked="" type="checkbox"/> MXS38013	User	4

Total: 1, selected: 1

Tables

New table +

<input type="checkbox"/> Name	Schema	Properties
<input type="checkbox"/> LOCATION	MXS38013	...
<input type="checkbox"/> PRODUCTS	MXS38013	...
<input type="checkbox"/> REGISTER	MXS38013	...
<input type="checkbox"/> SHOPS	MXS38013	...

Total: 4, selected: 0

IBM Db2 on Cloud

Load DataLoad History**Tables**ViewsIndexesAliasesMQTsSequencesApplication objects

SQL

MXS38013.PRODUCTS

Back

Export to CSV

NAME	QUANTITY	COST
Shirt	1	500

IBM Db2 on Cloud

Load DataLoad History**Tables**ViewsIndexesAliasesMQTsSequencesApplication objects

Find schemas or tables

Refresh

Schemas

Tables

New table

Name	Schema	Properties
<input type="checkbox"/> LOCATION	MXS38013	...
<input checked="" type="checkbox"/> PRODUCTS	MXS38013	...
<input type="checkbox"/> REGISTER	MXS38013	...
<input type="checkbox"/> SHOPS	MXS38013	...

Total: 4, selected: 1

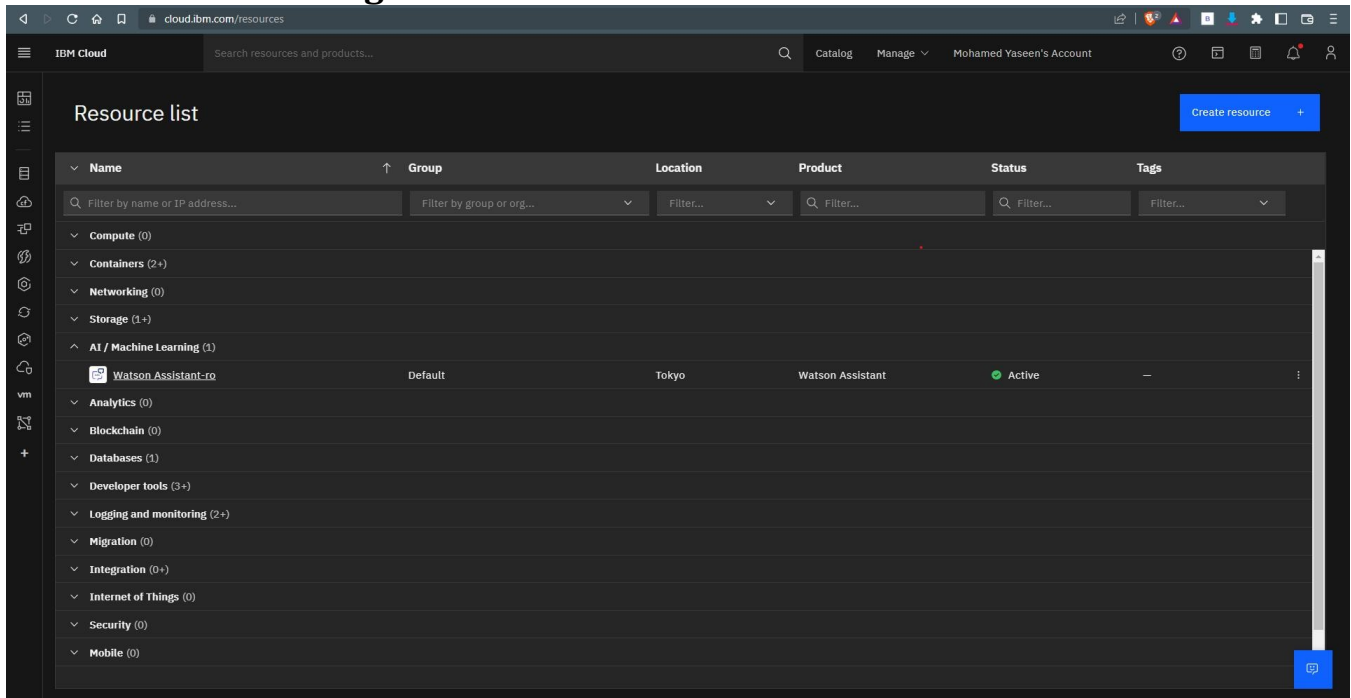
Table definition

Name	Data type	Nullable	Length	Scale	
NAME	VARCHAR	Y	32	0	ⓘ
QUANTITY	INTEGER	Y		0	ⓘ
COST	INTEGER	Y		0	ⓘ

View data

Chat-Box Output:

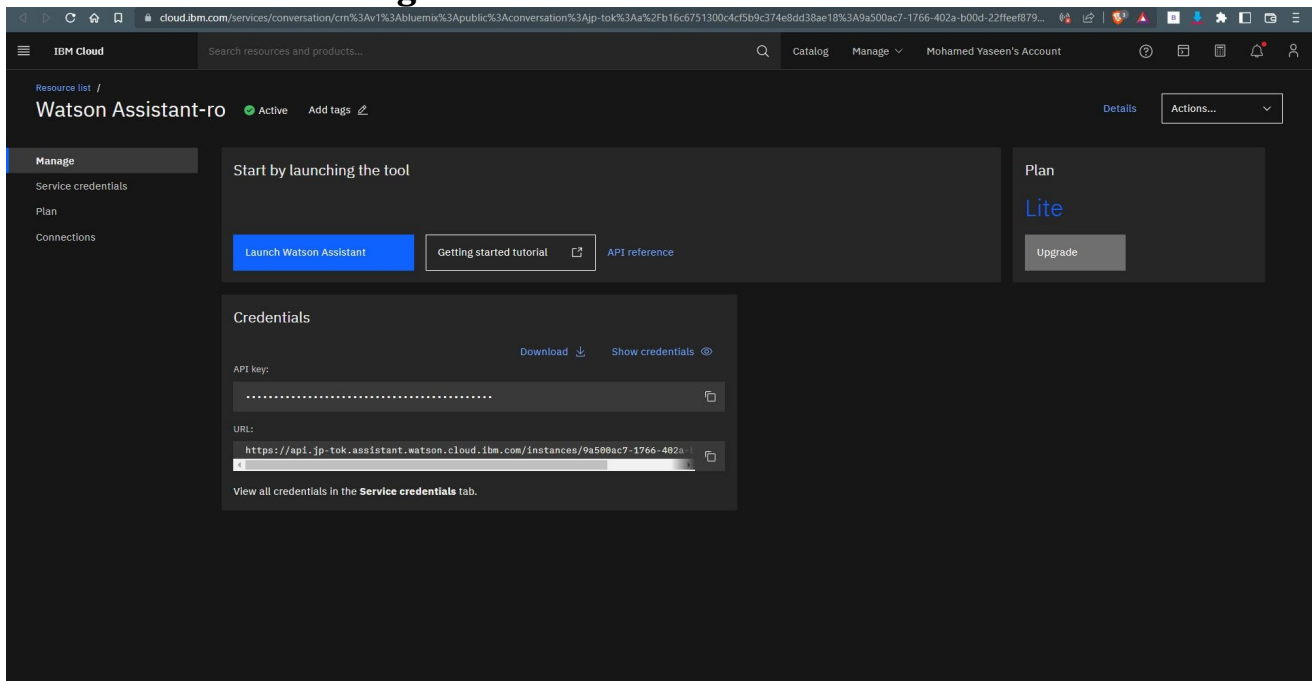
IBM Resource List Page:



The screenshot shows the IBM Cloud 'Resource list' page. The interface includes a top navigation bar with the IBM Cloud logo, a search bar, and links to 'Catalog', 'Manage', and the user's account. A left sidebar contains icons for various cloud services. The main content area displays a table of resources with columns for Name, Group, Location, Product, Status, and Tags. A 'Create resource' button is in the top right. The table lists several resource groups like Compute, Containers, Networking, Storage, and AI / Machine Learning. The 'AI / Machine Learning' group is expanded, showing a single resource named 'Watson Assistant-ro' with a status of 'Active'.

Name	Group	Location	Product	Status	Tags
▼ Compute (0)					
▼ Containers (2+)					
▼ Networking (0)					
▼ Storage (1+)					
▲ AI / Machine Learning (1)					
Watson Assistant-ro	Default	Tokyo	Watson Assistant	Active	—
▼ Analytics (0)					
▼ Blockchain (0)					
▼ Databases (1)					
▼ Developer tools (3+)					
▼ Logging and monitoring (2+)					
▼ Migration (0)					
▼ Integration (0+)					
▼ Internet of Things (0)					
▼ Security (0)					
▼ Mobile (0)					

IBM Watson Service Page on IBM Cloud:



The screenshot shows the IBM Watson Assistant service page. The top navigation bar is consistent with the previous page. The left sidebar has a 'Manage' tab selected. The main content area is titled 'Watson Assistant-ro' and shows the service is 'Active'. It includes a 'Launch Watson Assistant' button, a 'Getting started tutorial' link, and an 'API reference' link. A 'Plan' section shows the 'Lite' plan with an 'Upgrade' button. A 'Credentials' section displays the API key and URL, with a 'Show credentials' link. The URL is 'https://api.jp-tok.assistant.watson.cloud.ibm.com/instances/9a509ac7-1766-402a-b00d-22f1ee879...'.

Resource list / Watson Assistant-ro Active Add tags

Manage

Service credentials

Plan

Connections

Start by launching the tool

Launch Watson Assistant

Getting started tutorial

API reference

Plan

Lite

Upgrade

Credentials

Download

Show credentials

API key:

URL:

View all credentials in the Service credentials tab.

Launch Watson Assistant and create a virtual chat-box:

The screenshot displays the IBM Watson Assistant interface for a chatbot named 'Inventory Management'. The top navigation bar includes 'IBM Watson Assistant', 'Lite', 'Upgrade', and 'Hospital Bot'. The main content area is divided into three sections:

- Customer starts with:** A section for defining initial phrases. It includes a text input field and a 'Total: 5' indicator.
- Conversation steps:** A flowchart showing the chatbot's logic. It starts with a welcome message, followed by a decision point '1 is Sign up/in'. If 'Sign up/in' is selected, it goes to step 2, which is a welcome message. If 'Shops' is selected, it goes to step 3, which is a message about the store's status. If 'Products' is selected, it goes to step 4, which is a message about the products tab. Each step has a 'Continue to next step' button.
- Preview:** A window showing the chatbot's interaction. It displays the welcome message, the user's input 'inventory management', and the chatbot's response 'Welcome to our inventory management system, we are glad you are here. Please follow the instructions to register: 1. Click on register or sign up 2. Enter your name, email address and password 3. Click on Create Account'. The preview also shows the chatbot's internal state and a 'Type something...' input field.

Embed it on your python file:

The screenshot displays the IBM Watson Assistant interface for the 'Inventory Management' chatbot, specifically the 'Embed' tab. The top navigation bar includes 'IBM Watson Assistant', 'Lite', 'Upgrade', and 'Hospital Bot'. The main content area is divided into two sections:

- Web chat:** A section for managing the chatbot's web chat interface. It includes a 'Draft' button, a 'Close' button, and a 'Save and exit' button.
- Embed:** A section for embedding the chatbot on a website. It includes a code editor with the following JavaScript code:

```
<script>
window.watsonAssistantChatOptions = {
  integrationID: "01ebdd1a-1a11-43a6-a065-ed2662bdcf2b", // The ID of this integration.
  region: "jp-tok", // The region your integration is hosted in.
  serviceInstanceID: "9a800ac7-1766-402a-b00d-22f8e187956", // The ID of your service instance.
  onLoad: function(instance) { instance.render(); }
};
setTimeout(function() {
  const t=document.createElement('script');
  t.src="https://web-chat.global.assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions
  document.head.appendChild(t);
});
</script>
```

The code editor also includes a 'Show more' button.

```
22 <!-- Chat bot script -->
23 <script>
24   window.watsonAssistantChatOptions = {
25     integrationID: "01ebdd1a-1a11-43a6-a065-ed2662bdcf2b", // The ID of this integration.
26     region: "jp-tok", // The region your integration is hosted in.
27     serviceInstanceID: "9a500ac7-1766-402a-b08d-22ffef87956", // The ID of your service instance.
28     onLoad: function(instance) { instance.render(); }
29   };
30   setTimeout(function(){
31     const t=document.createElement('script');
32     t.src="https://web-chat.global/assistant.watson.appdomain.cloud/versions/" + (window.watsonAssistantChatOptions.clientVersion || 'latest') + "/WatsonAssistant.js";
33     document.head.appendChild(t);
34   });
35 </script>
36
37 { % endblock % }
```

Verifying on our page:

