

# PLASMA DONOR APPLICATION

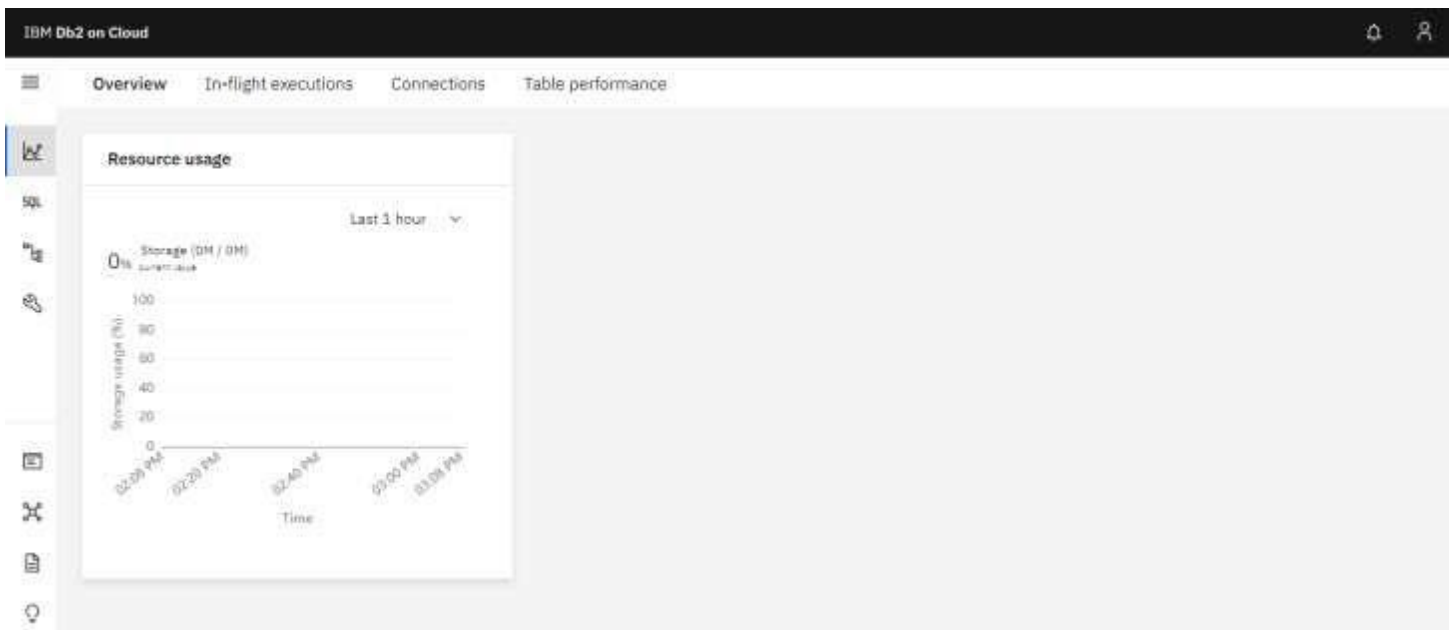
Date	16 Nov 2022
Team ID	PNT2022TMID33940
Project Name	Plasma Donor Application

## Implementing Web Application

### Create IBM DB2 and interact with application

Creation of IBM DB2 database

The screenshot shows the IBM Cloud console interface. At the top, there's a navigation bar with 'IBM Cloud' and a search bar. Below the navigation bar, the 'Resource list' section displays 'Db2-06' with a green 'Active' status and an 'Add tags' link. To the right of the resource name are 'Details' and 'Actions...' buttons. On the left, a 'Manage' sidebar lists 'Getting started', 'Service credentials', and 'Connections'. The main content area is titled 'Getting started' and contains instructions on how to find credentials. It includes a 'Go to UI' button with an external link icon and a 'Getting started docs' button. To the right, a 'Need help?' section prompts the user to submit a support case, with a corresponding 'Support case' button. A blue chat icon is visible in the bottom right corner.



The screenshot shows the IBM Db2 on Cloud console interface. At the top, there's a navigation bar with 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. Below this is a search bar labeled 'Find schemas or tables' and a 'Refresh' button. The main content area is divided into two panels: 'Schemas' and 'Tables'.

**Schemas Panel:**

Name	Type	Tables
DWS78237	User	3

Total: 1, selected: 1

**Tables Panel:**

Name	Schema	Properties
ADMIN	DWS78237	---
DETAILS	DWS78237	---
DONORS	DWS78237	---

Total: 3, selected: 0

## Database connection coding

```
from flask import Flask, render_template, request, session, redirect, url_for
import ibm_db
import os
app=Flask(__name__)
app.secret_key='hidden'
conn = ibm_db.connect(
    f"DATABASE={os.environ.get('DATABASE')};"
    f"HOSTNAME={os.environ.get('HOSTNAME')};"
    f"PORT={os.environ.get('PORT')};"
    f"USERNAME={os.environ.get('DB_USERNAME')};"
    f"PASSWORD={os.environ.get('PASSWORD')};"
    "SECURITY=SSL;"
    f"SSLSERVERCERTIFICATE={os.environ.get('SSLSERVERCERTIFICATE')};",
    '',
    ''
)
print(conn)
@app.route("/")
def front():
    return render_template("front.html")
```