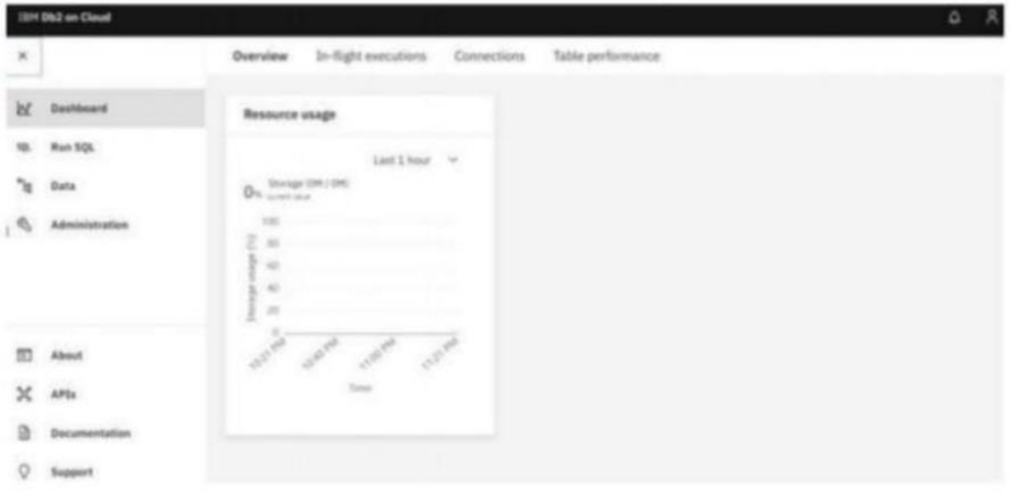
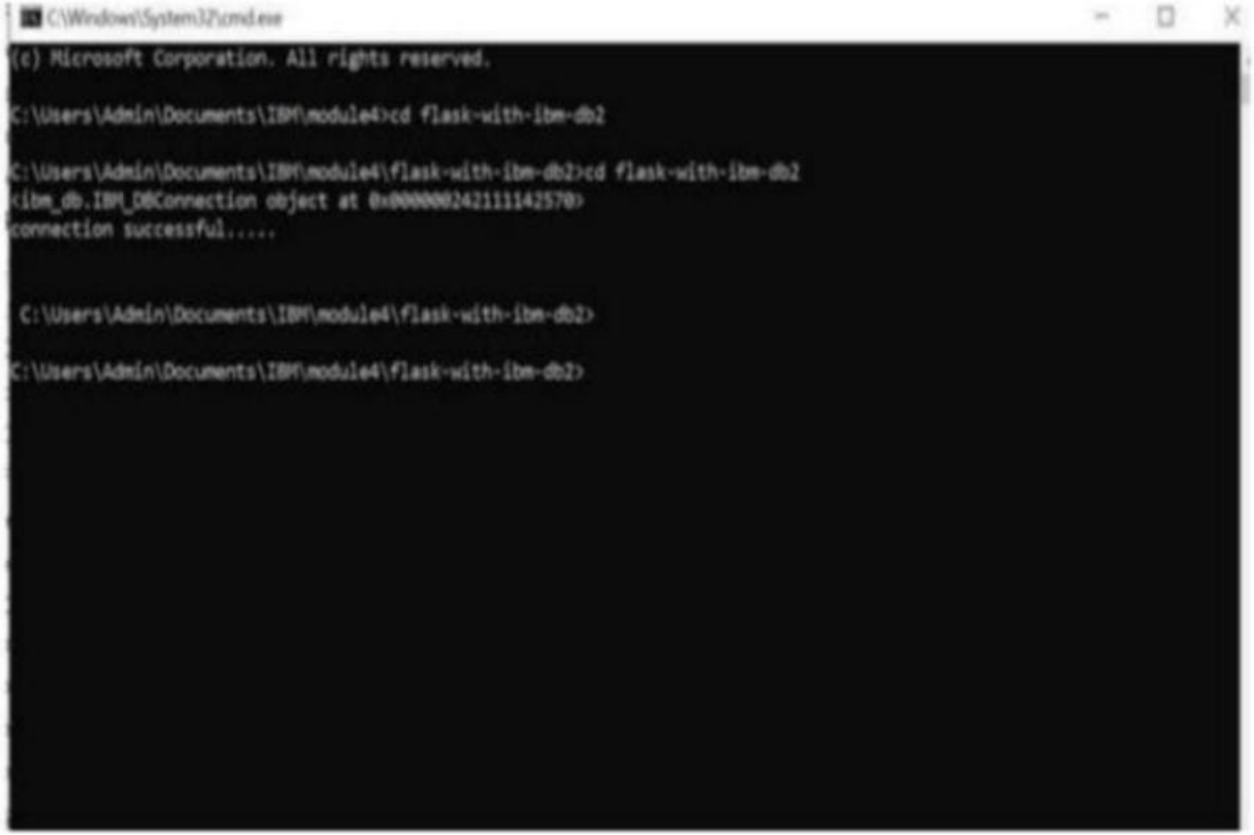
IMPLEMENTATION WEB APPLICATIONS

CREATE IBM AND CONNECT WITH PYTHON

IBM DB2

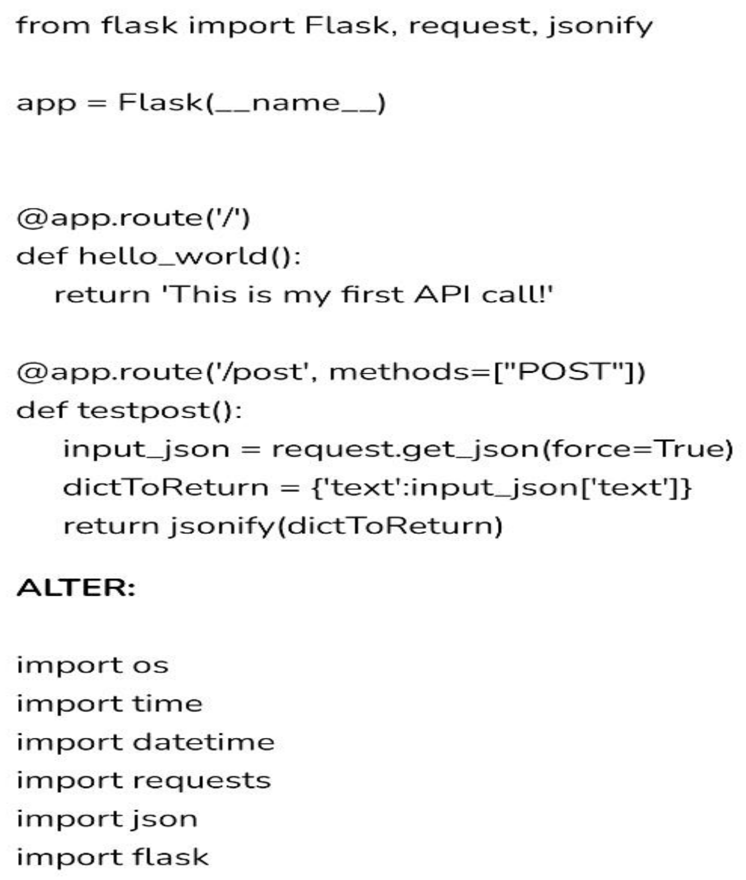


DB2 WITH PYTHON



```
(c) Microsoft Corporation. All rights reserved.

C:\Users\Admin\Documents\IBM\module4>cd flask-with-ibm-db2

C:\Users\Admin\Documents\IBM\module4\flask-with-ibm-db2>cd flask-with-ibm-db2
<ibm_db.IBM_DBConnection object at 0x000000242111142570>
connection successful.....


 C:\Users\Admin\Documents\IBM\module4\flask-with-ibm-db2>

C:\Users\Admin\Documents\IBM\module4\flask-with-ibm-db2>
```

CREATE APIS IN FLASK



```
from flask import Flask, request, jsonify

app = Flask(__name__)


@app.route('/')
def hello_world():
    return 'This is my first API call!'

@app.route('/post', methods=["POST"])
def testpost():
     input_json = request.get_json(force=True)
     dictToReturn = {'text':input_json['text']}
     return jsonify(dictToReturn)
```

ALTER:

```
import os
import time
import datetime
import requests
import json
import flask
import boto3
import psycopg2
from flask import request

app = flask.Flask(__name__)
app.config["DEBUG"] = True
db_cred = os.environ['DB_CRED']

def connectToDb():
    conn = psycopg2.connect(db_cred)
    cur = conn.cursor()
    return conn, cur

def closeDbConnection(conn, cur):
    cur.close()
    conn.close()

#This function will be used to execute a query and return the result
def db_query(query):
    conn, cur = connectToDb()
    cur.execute(query)
    res = cur.fetchall()
    closeDbConnection(conn, cur)
    return res

#This function will be used to insert a row or rows into the database
def db_insert(query, values):
    conn, cur = connectToDb()
    cur.execute(query, values)
    conn.commit()
    close
```

Why use an API?

* **Decoupling**: By using an API other components are only dependent on the API, not the internals of the component, which means you can make changes to the internals without breaking the dependents.

* **Language Independence**: An API can be called from any programming language.

* **Isolation**: By using an API, the components using the API are isolated from each other. One component can be replaced without affecting any other component that uses the API.

What is Flask?

Flask is a microframework for Python based on Werkzeug, Jinja 2 and good intentions.

![flask](/images/flask.png)

Why use Flask?

* **Light**: Flask is one of the lightest frameworks available.

* **Easy to use**: Flask is easy to use and it’s well documented.

* **Flexible**