

CREATE DB2 AND CONNECT PYTHON

DATE	17 NOVEMBER 2022
TEAM ID	PNT2022TMID36064
PROJECT NAME	SMART FASHION RECOMMENDER APPLICATION

DATABASE CONNECTIVITY IN IBM:

The screenshot displays the IBM Data Studio interface. The top navigation bar includes options like Load Data, Load History, Tables, Views, Indexes, Aliases, MQTs, Sequences, and Application objects. The 'Tables' tab is active, showing a list of tables under the 'JOB' schema. The 'Table definition' panel on the right shows the structure of the 'JOB' table, including columns like USERNAME, EMAIL, QUALIFICATION, SKILLS, and JOBS, all of type VARCHAR with a length of 32 and a nullable status of Y.

Name	Schema	Properties
JOB	DJH26469	...
USERS	DJH26469	...

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	Y	32	0
EMAIL	VARCHAR	Y	32	0
QUALIFICATION	VARCHAR	Y	32	0
SKILLS	VARCHAR	Y	32	0
JOBS	VARCHAR	Y	32	0

CONNECT WITH PYTHON CODE:

The screenshot shows a Python application code in a code editor. The code is a Flask web application that connects to an IBM DB2 database and handles a login request. The code includes imports for Flask, IBM DB2, and request handling. It defines a Flask app, sets a secret key, and defines a login route that checks user credentials against the database.

```
1 from flask import Flask, render_template, request, redirect, url_for, session
2 import ibm_db
3 import re
4
5 app = Flask(__name__)
6
7 app.secret_key = 'a'
8
9 conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-4859-88f0-a6c3e6b4b907.c3n41cmd0nqnk39u98g.database-
10
11 @app.route('/')
12
13 def homer():
14     return render_template('login.html')
15
16
17 @app.route('/Login',methods =['GET', 'POST'])
18 def login():
19     global userid
20     msg = ''
21
22
23     if request.method == 'POST' :
24         username = request.form['username']
25         password = request.form['password']
26         sql = "SELECT * FROM users WHERE username =? AND password=?"
27         stmt = ibm_db.prepare(conn, sql)
28         ibm_db.bind_param(stmt,1,username)
29         ibm_db.bind_param(stmt,2,password)
30         ibm_db.execute(stmt)
31         account = ibm_db.fetch_assoc(stmt)
32         print (account)
```