

SMART FASHION RECOMMENDER

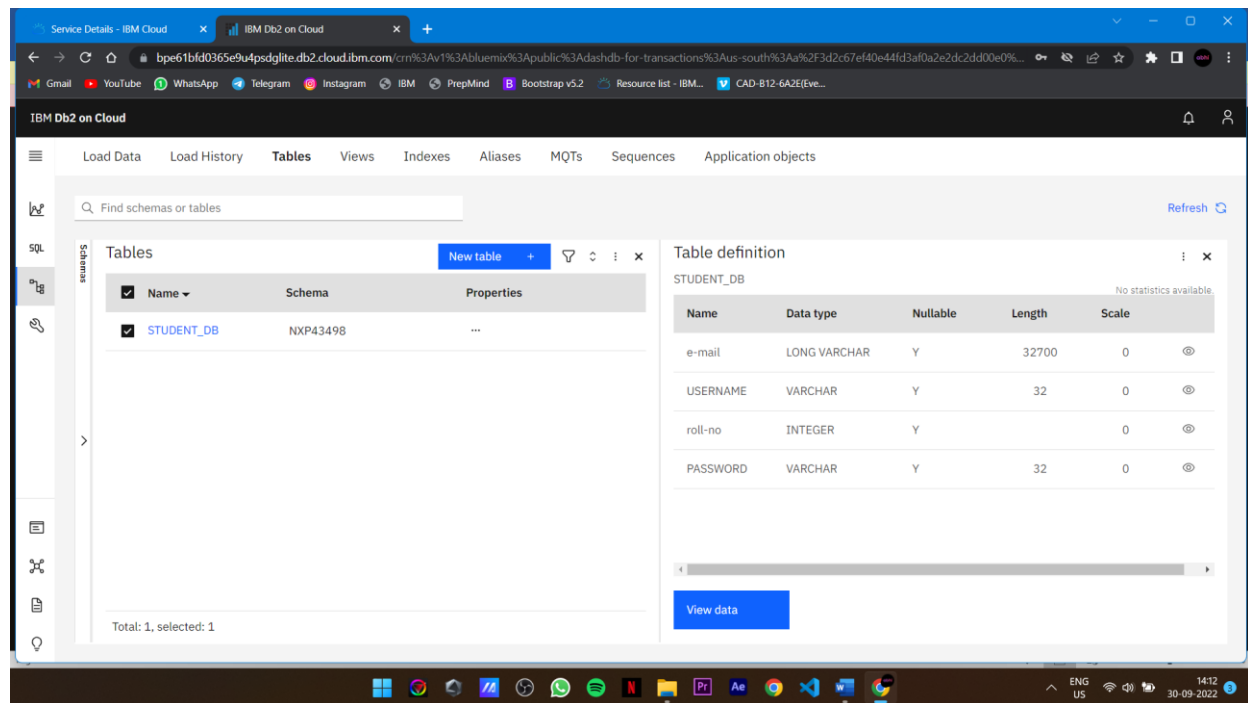
APPLICATION

ASSIGNMENT -2

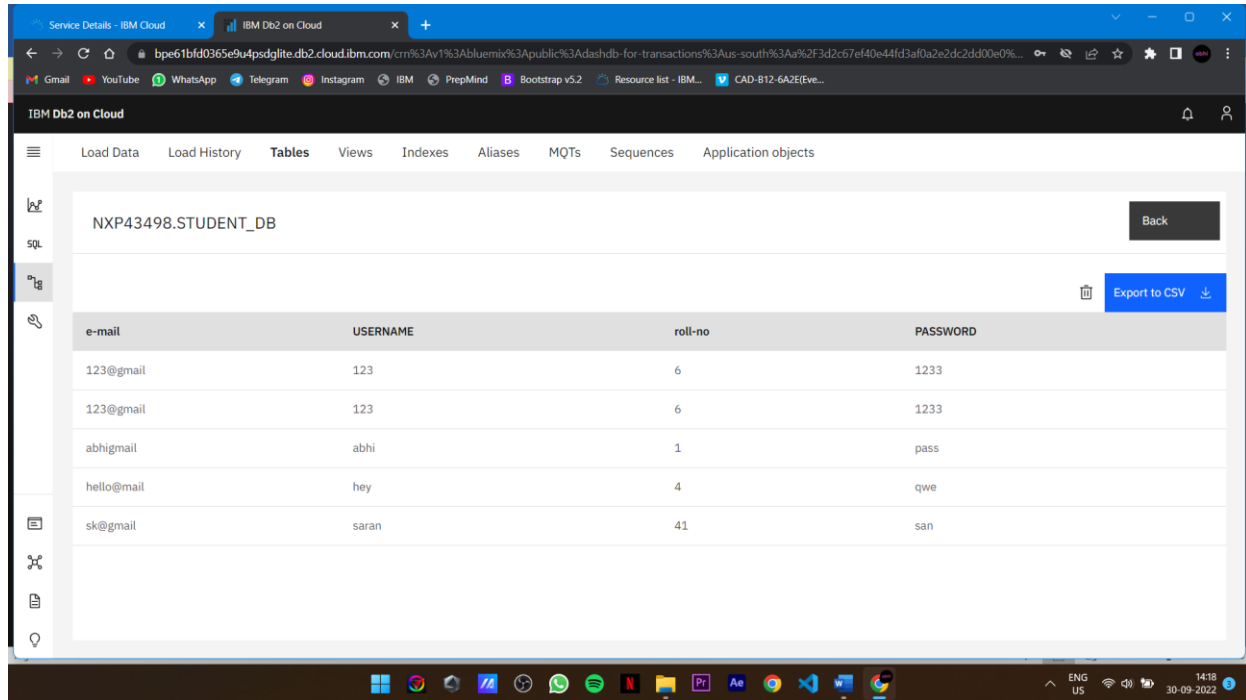
TEAM ID: PNT2022TMID42734

STUDENT NAME: Santhosh.S

1. Create a user 1table with username,Email, Rollnumber and Password.



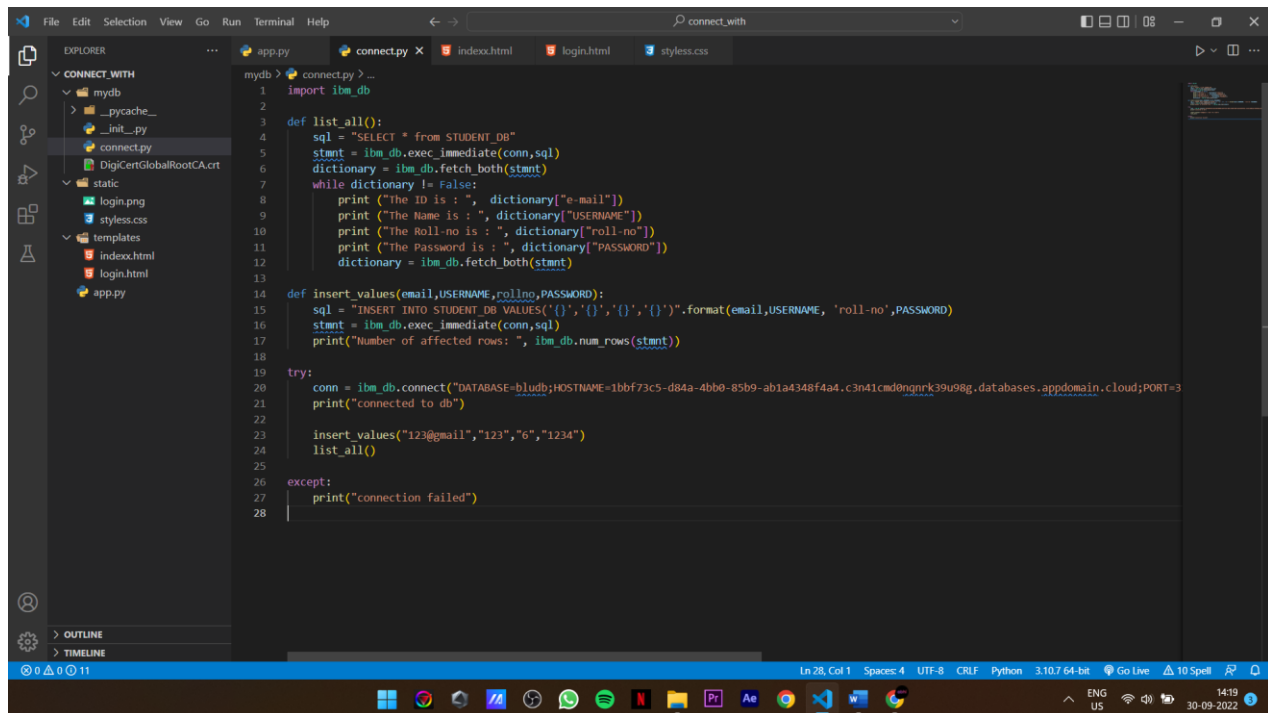
2. Perform update,delete queries with usertable:



The screenshot shows the IBM Db2 on Cloud console interface. The top navigation bar includes 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected, displaying a table named 'NXP43498.STUDENT_DB'. A 'Back' button is in the top right. Below the table name, there is an 'Export to CSV' button. The table structure is as follows:

e-mail	USERNAME	roll-no	PASSWORD
123@gmail	123	6	1233
123@gmail	123	6	1233
abhi@gmail	abhi	1	pass
hello@mail	hey	4	qwe
sk@gmail	saran	41	san

3. Connect pythoncode withdb2(Database):

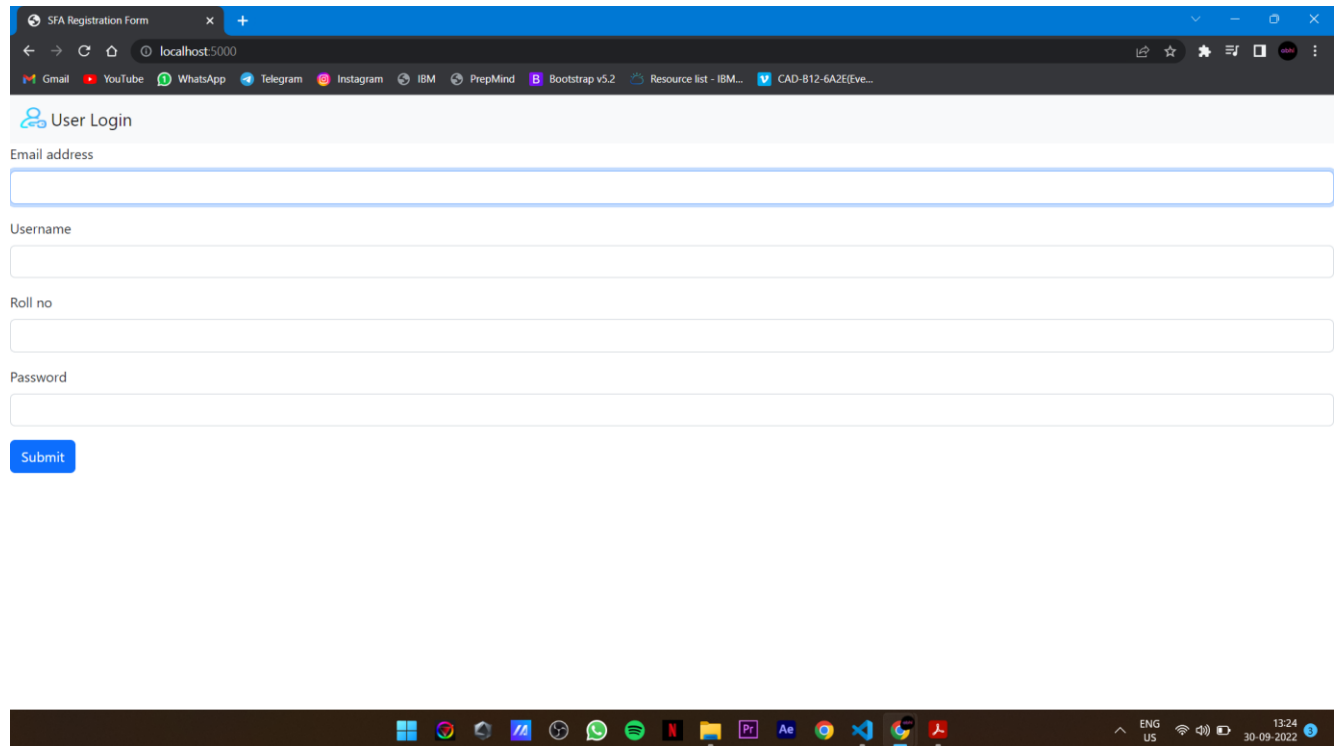


The screenshot shows a Python script named 'connect.py' in a code editor. The script connects to an IBM Db2 database and performs several operations:

```
1 import ibm_db
2
3 def list_all():
4     sql = "SELECT * from STUDENT_DB"
5     stmt = ibm_db.exec_immediate(conn,sql)
6     dictionary = ibm_db.fetch_both(stmt)
7     while dictionary != False:
8         print("The ID is : ", dictionary["e-mail"])
9         print("The Name is : ", dictionary["USERNAME"])
10        print("The Roll-no is : ", dictionary["roll-no"])
11        print("The Password is : ", dictionary["PASSWORD"])
12        dictionary = ibm_db.fetch_both(stmt)
13
14 def insert_values(email,USERNAME,rollno,PASSWORD):
15     sql = "INSERT INTO STUDENT_DB VALUES('{}','{}','{}','{}').format(email,USERNAME, 'roll-no',PASSWORD)
16     stmt = ibm_db.exec_immediate(conn,sql)
17     print("Number of affected rows: ", ibm_db.num_rows(stmt))
18
19 try:
20     conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=1bbf73c5-d84a-4bb0-85b9-ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=3
21     print("connected to db")
22
23     insert_values("123@gmail","123","6","1234")
24     list_all()
25
26 except:
27     print("connection failed")
28
```

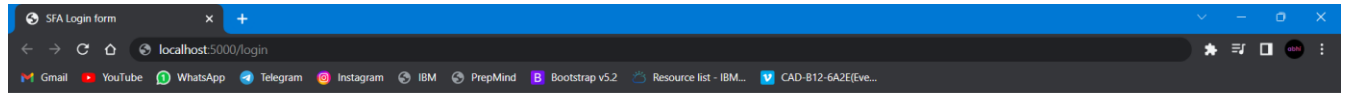
4. 4) Create a flask app with registration page, login page and welcome page. By default load the registration page once the user enters all the fields, store the data in the database and navigate to the login page to authenticate user username and password. If the user is valid, show the welcome page.

a) Registration page:



The screenshot shows a web browser window with the title 'SFA Registration Form' and the address bar displaying 'localhost:5000'. The page content includes a header with a user icon and the text 'User Login'. Below the header, there are four input fields labeled 'Email address', 'Username', 'Roll no', and 'Password'. A blue 'Submit' button is located at the bottom left of the form area. The browser's taskbar at the bottom shows various application icons and the system clock indicating 13:24 on 30-09-2022.

b) Login page:



Email address

Password

Submit



c) Homepage:

