

Assignment -2

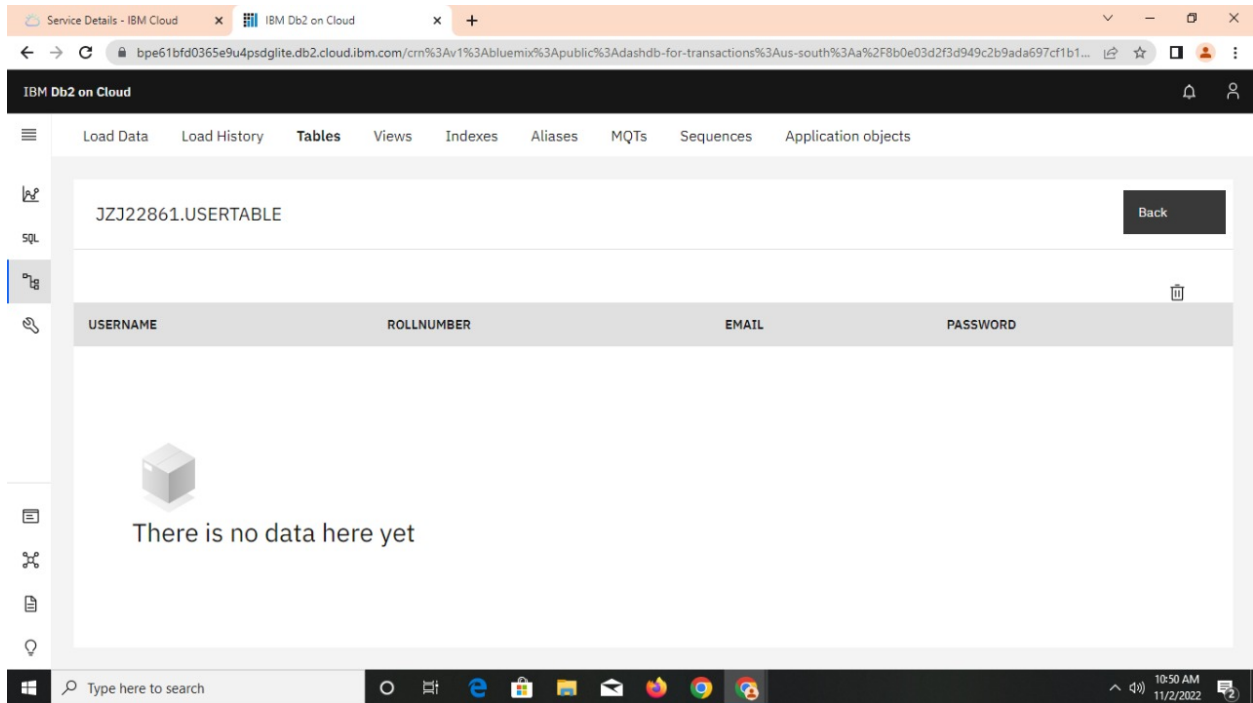
Assignment Date	29 September 2022
Name	YASAR ARAFARTH S
Project Name	Smart Fashion Recommender Application
Maximum marks	2 Marks

Questions: 1.

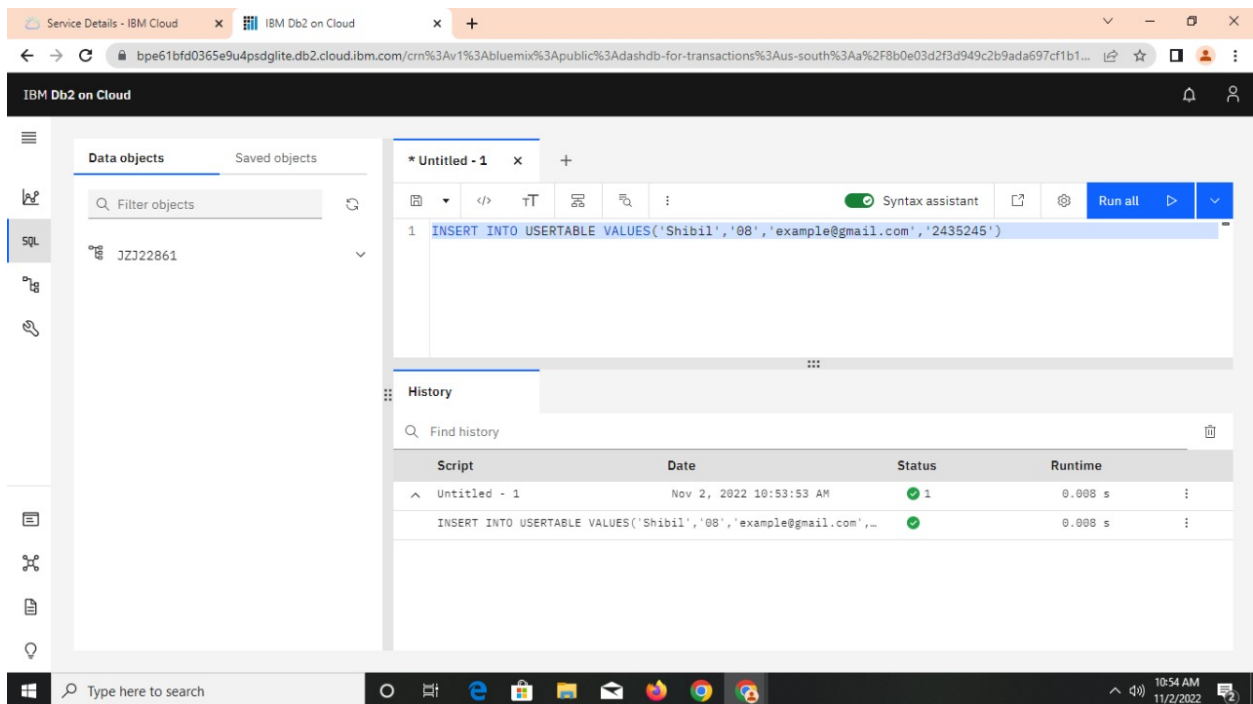
Create User table with user with email, username, roll number, password.

The screenshot displays the IBM Db2 on Cloud web interface. The 'Tables' tab is active, showing a list of tables. The 'USERTABLE' is selected, and its definition is shown on the right. The table has four columns: USERNAME (CHAR, 30, nullable), ROLLNUMBER (INTEGER, nullable), EMAIL (VARCHAR, 32, nullable), and PASSWORD (VARCHAR, 32, nullable). The 'View data' button is visible at the bottom of the table definition panel.

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	Y	30	0
ROLLNUMBER	INTEGER	Y		0
EMAIL	VARCHAR	Y	32	0
PASSWORD	VARCHAR	Y	32	0



2. Perform UPDATE,DELETE Queries with user table



Service Details - IBM Cloud x IBM Db2 on Cloud x +

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F8b0e03d2f3d949c2b9ada697cf1b1...

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

JZJ22861.USERTABLE Back

Export to CSV

USERNAME	ROLLNUMBER	EMAIL	PASSWORD
Shibil	8	example@gmail.com	2435245

Type here to search 10:54 AM 11/2/2022

DELETE TABLE:

Service Details - IBM Cloud x IBM Db2 on Cloud x SQL0000 - SQL0249 - IBM Docu x +

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2F8b0e03d2f3d949c2b9ada697cf1b1...

IBM Db2 on Cloud

Data objects Saved objects

Filter objects

JZJ22861

*Untitled - 1 x +

SQL

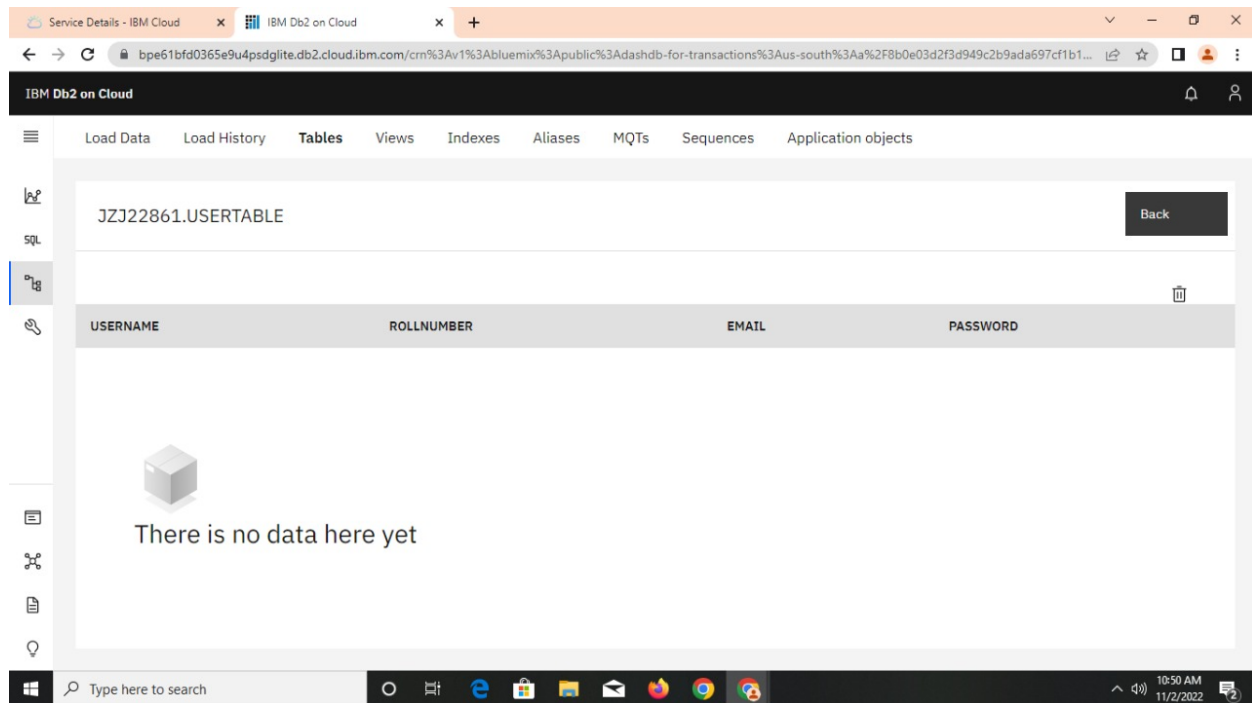
```
1 DELETE FROM USERTABLE WHERE ROLLNUMBER = 67;
```

History Results

Find history

Script	Date	Status	Runtime
Untitled - 1	Nov 2, 2022 10:56:39 AM	1	0.008 s
INSERT INTO USERTABLE VALUES('Shibil','08','example@gmail.com...			0.008 s
Untitled - 1	Nov 2, 2022 10:55:51 AM	1	0.015 s
INSERT INTO USERTABLE VALUES('Shibil','08','example@gmail.com...			0.015 s
Untitled - 1	Nov 2, 2022 10:53:53 AM	1	0.008 s
INSERT INTO USERTABLE VALUES('Shibil','08','example@gmail.com...			0.008 s

Type here to search 10:58 AM 11/2/2022



3.Connect python code to db2

```
1 from flask import Flask, render_template, request, redirect, url_for, session
2 import ibm_db
3 import re
4 app = Flask(__name__)
5 app.secret_key = 'a'
6 conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=b70af05b-76e4-4bca-a1f5-
7 23dbb4c6a74e.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=
8 32716;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=jzc43091;PWD=PI8VtGRvZlSV
9 T65A",'','')
10 @app.route('/')
11 def homer():
12     return render_template('home.html')
13 @app.route('/login',methods =['GET', 'POST'])
14 def login():
15     global userid
16     msg = ''
```

```

16 if request.method == 'POST' :
17     username = request.form['username']
18     password = request.form['password']
19     sql = "SELECT * FROM users WHERE username =? AND password=?"
20     stmt = ibm_db.prepare(conn, sql)
21     ibm_db.bind_param(stmt,1,username)
22     ibm_db.bind_param(stmt,2,password)
23     ibm_db.execute(stmt)
24     account = ibm_db.fetch_assoc(stmt)
25     print (account)
26     if account:
27         session['loggedin'] = True
28         session['id'] = account['USERNAME']
29         userid= account['USERNAME']
30         session['username'] = account['USERNAME']
31         msg = 'Logged in successfully !'
32         msg = 'Logged in successfully !'
33         return render_template('dashboard.html', msg = msg)
34     else:
35         msg = 'Incorrect username / password !'
36         return render_template('login.html', msg = msg)
37 @app.route('/register', methods=['GET', 'POST'])
38 def registet():
39     msg = ''
40     if request.method == 'POST' :
41         username = request.form['username']
42         email = request.form['email']
43         password = request.form['password']
44         sql = "SELECT * FROM users WHERE username =?"
45         stmt = ibm_db.prepare(conn, sql)
46         ibm_db.bind_param(stmt,1,username)
47         ibm_db.execute(stmt)
48         account = ibm_db.fetch_assoc(stmt)
49         print(account)
50         if account:
51             msg = 'Account already exists !'
52         elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
53             msg = 'Invalid email address !'
54         elif not re.match(r'[A-Za-z0-9]+', username):
55             msg = 'name must contain only characters and numbers !'

```

```

56 else:
57     insert_sql = "INSERT INTO users VALUES (?, ?, ?)"
58     prep_stmt = ibm_db.prepare(conn, insert_sql)
59     ibm_db.bind_param(prepare_stmt, 1, username)
60     ibm_db.bind_param(prepare_stmt, 2, email)
61     ibm_db.bind_param(prepare_stmt, 3, password)
62     ibm_db.execute(prepare_stmt)
63     msg = 'You have successfully registered !'
64     elif request.method == 'POST':
65     msg = 'Please fill out the form !'
66     return render_template('register.html', msg = msg)
67 @app.route('/dashboard')
68 def dash():
69     return render_template('dashboard.html')
70 @app.route('/apply', methods = ['GET', 'POST'])
71 def apply():
72     msg = ''
73     if request.method == 'POST' :
74     username = request.form['username']
75     email = request.form['email']
76     qualification= request.form['qualification']
77     skills = request.form['skills']
78     jobs = request.form['s']
79     sql = "SELECT * FROM users WHERE username =?"
80     stmt = ibm_db.prepare(conn, sql)
81     ibm_db.bind_param(stmt,1,username)
82     ibm_db.execute(stmt)
83     account = ibm_db.fetch_assoc(stmt)
84     print(account)
85     if account:
86     msg = 'there is only 1 job position! for you'
87     return render_template('apply.html', msg = msg)
88     insert_sql = "INSERT INTO job VALUES (?, ?, ?, ?, ?)"
89     prep_stmt = ibm_db.prepare(conn, insert_sql)
90     ibm_db.bind_param(prepare_stmt, 1, username)
91     ibm_db.bind_param(prepare_stmt, 2, email)
92     ibm_db.bind_param(prepare_stmt, 3, qualification)
93     ibm_db.bind_param(prepare_stmt, 4, skills)
94     ibm_db.bind_param(prepare_stmt, 5, jobs)
95     ibm_db.execute(prepare_stmt)

```

```

96 msg = 'You have successfully applied for job !'
97 session['loggedin'] = True
98 TEXT = "Hello,a new application for job position" +jobs+"is
    requested"
99 elif request.method == 'POST':
100 msg = 'Please fill out the form !'
101 return render_template('apply.html', msg = msg)
102 @app.route('/display')
103 def display():
104 print(session["username"],session['id'])
105 cursor = mysql.connection.cursor()
106 cursor.execute('SELECT * FROM job WHERE userid = % s',
    (session['id'],))
107 account = cursor.fetchone()
108 print("accountdisplay",account)
109 return render_template('display.html',account = account)
110 @app.route('/logout')
111 def logout():
112 session.pop('loggedin', None)
113 session.pop('id', None)
114 session.pop('username', None)
115 return render_template('home.html')
116 if __name__ == '__main__':
117 app.run(host='0.0.0.0')

```

Apply.html

```

1 <html>
2 <body>
3 <form action="http://localhost:5000/login" method="POST">
4 <p>Enter Username:</p>
5 <p><input type="text" name="username" /></p>
6 <p>Enter Email:</p>
7 <p><input type="email" name="email" /></p>
8 <p>Enter Qualification:</p>
9 <p><input type="text" name="qualification" /></p>
10 <p>Enter Skill:</p>
11 <p><input type="text" name="skill" /></p>

```

```
12 <p>Enter Jobs:</p>
13 <p><input type="text" name="jobs" /></p>
14 <p><input type="submit" value="submit"></p>
15 </form>
16 </body>
17 </html>
```

Login.html

```
1 <html>
2 <body>
3 <form action="http://localhost:5000/login" method="POST">
4 <p>Enter Username:</p>
5 <p><input type="text" name="username" /></p>
6 <p>Enter Password:</p>
7 <p><input type="password" name="password" /></p>
8 <p><input type="submit" value="submit"></p>
9 </form>
10 </body>
11 </html>
```

Register.html

```
1 <html>
2 <body>
3 <form action="http://localhost:5000/login" method="POST">
4 <p>Enter Username:</p>
5 <p><input type="text" name="username" /></p>
6 <p>Enter Email:</p>
7 <p><input type="email" name="email" /></p>
8 <p>Enter Password:</p>
9 <p><input type="password" name="password" /></p>
10 <p><input type="submit" value="submit"></p>
11 </form>
12 </body>
13 </html>
```