ASSIGNMENT 2

Name	Jothika S
Team ID	PNT2022TMID29529
Project Name	Inventory Management System for Retailers

Questions:

1. Create a user table with user with email, username, password, roll no

2. Perform update, and delete queries with user table

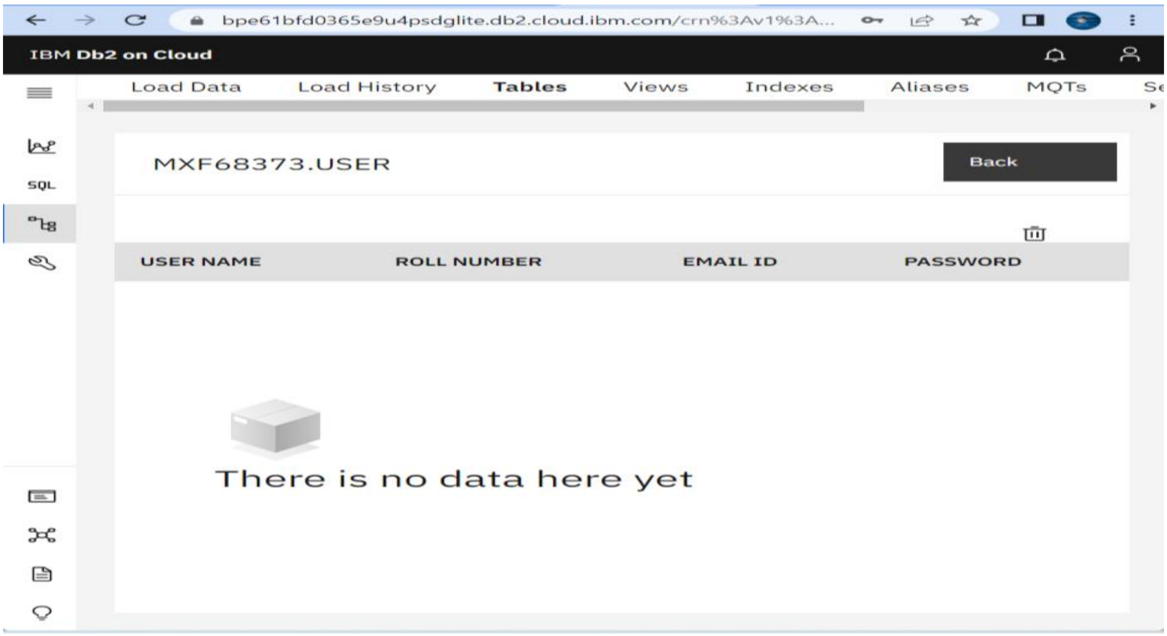
3. Connect python code to database2

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 database and navigate to login page authenticate user username and password if the user is valid show the welcome page.

Answers:

1.

```
CREATE TABLE users(
        username char(32),
        roll_number varchar(32),
        email varchar(32),
        password varchar(32)
        );
```



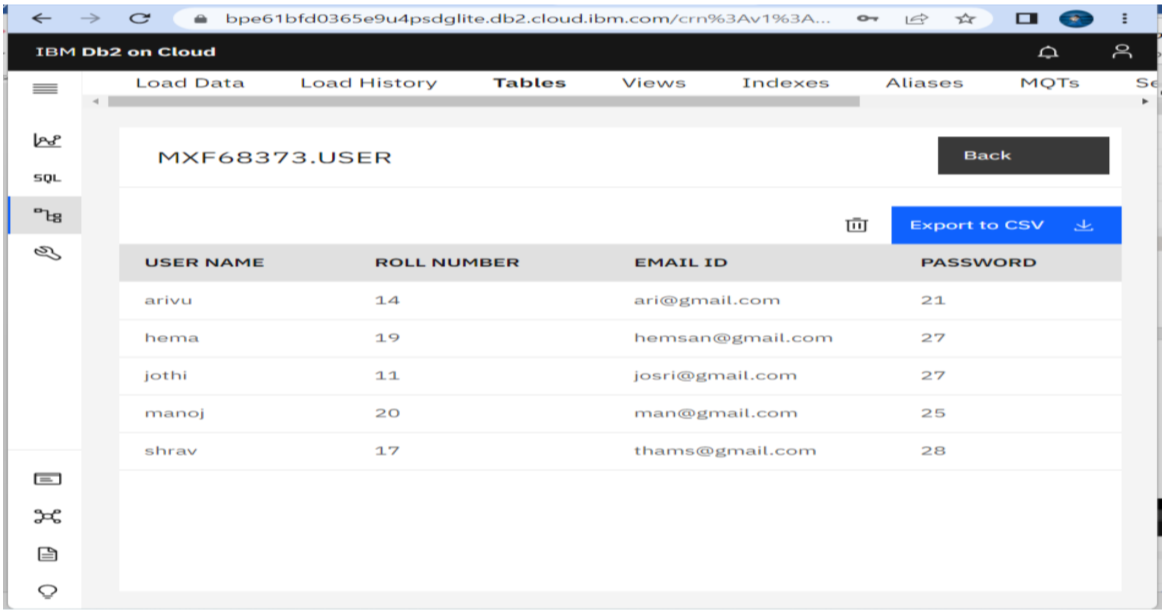
MXF68373.USER

USER NAME	ROLL NUMBER	EMAIL ID	PASSWORD

There is no data here yet

2. Insert Table

```
insert into user values('jothi',11,'josri@gmail.com',27);
insert into user values('arivu',14,'ari@gmail.com',21);
insert into user values('shrav',17,'thams@gmail.com',28);
insert into user values('hema',19,'hemsan@gmail.com',27);
insert into user values('manoj',20,'man@gmail.com',25);
```

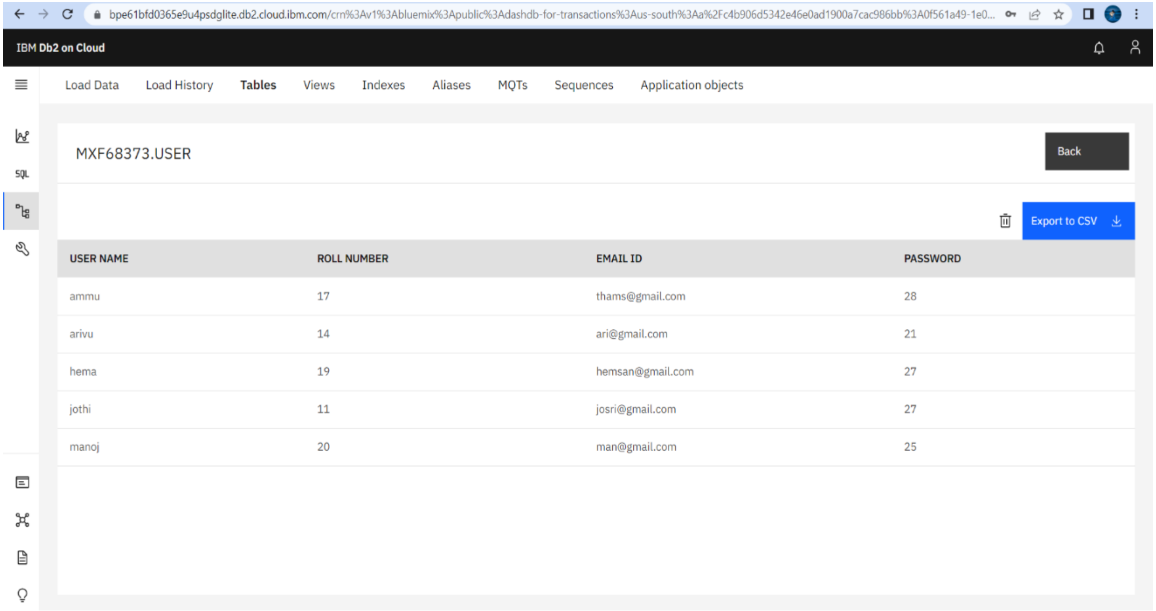


MXF68373.USER

USER NAME	ROLL NUMBER	EMAIL ID	PASSWORD
arivu	14	ari@gmail.com	21
hema	19	hemsan@gmail.com	27
jothi	11	josri@gmail.com	27
manoj	20	man@gmail.com	25
shrav	17	thams@gmail.com	28

UPDATE TABLE:

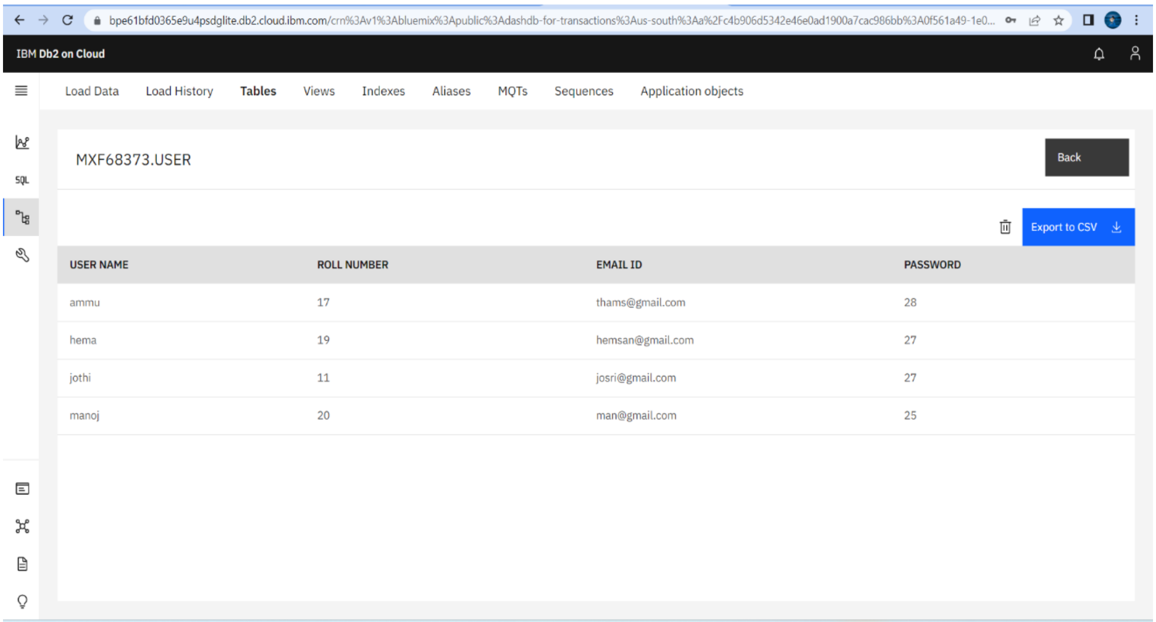
update users (SET username='ammu' WHERE roll_number='17');


bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fc4b906d5342e46e0ad1900a7cac986bb%3A0f561a49-1e0...
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
MXF68373.USER
Back
Export to CSV

USER NAME	ROLL NUMBER	EMAIL ID	PASSWORD
ammu	17	thams@gmail.com	28
arivu	14	ari@gmail.com	21
hema	19	hemsan@gmail.com	27
jothi	11	josri@gmail.com	27
manoj	20	man@gmail.com	25

DELETE TABLE:

delete from users (WHERE roll_number='14');


bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fc4b906d5342e46e0ad1900a7cac986bb%3A0f561a49-1e0...
IBM Db2 on Cloud
Load Data
Load History
Tables
Views
Indexes
Aliases
MQTs
Sequences
Application objects
SQL
MXF68373.USER
Back
Export to CSV

USER NAME	ROLL NUMBER	EMAIL ID	PASSWORD
ammu	17	thams@gmail.com	28
hema	19	hemsan@gmail.com	27
jothi	11	josri@gmail.com	27
manoj	20	man@gmail.com	25

3. Connect python code to database2:

```
conn = ibm db.connect(DATABASE=bludb;
HOSTNAME= 1bbf73c5-d84a-4bb0-85b9-
ab1a4348f4a4.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;
PORT: 32286; SECURITY=SSL;ServercertificateC:\Users\Dell\Desktop\job;
UID=mqc88974";PWD=mbf14fY4F8mTNrXJ;)
```

4.

```
from flask import Flask, render template, request, redirect, url for, session
from flask mysqldb import MySQL import MySQLdb.cursors
import reapp = Flask( name )
app.secret key= 'your secret key'
app.config['MYSQL HOST'] ='localhost'
app.config['MYSQL USER'] = 'root'
app.config['MYSQL PASSWORD'] = 'your password'
app.config['MYSQL DB'] = 'geeklogin'
mysql =MySQL(app)
@app.route('/')
@app.route('/login', methods =['GET', 'POST'])
def login():
msg = "ifrequest.method == 'POST' and 'username' in request.form and
'password' in request.form:
username = request.form['username']
password = request.form['password']
cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
cursor.execute('SELECT * FROM accounts WHERE username = % s
AND password = % s', (username, password,
)) account = cursor.fetchone()
```

```
if account:
session['loggedin'] = True
session['id'] = account['id']
session['username'] = account['username']
msg = 'Logged in successfully !'
return render template('index.html', msg = msg)
else:
msg = 'Incorrect username / password !'
return render template('login.html', msg = msg)
@app.route('/logout') def
logout():
session.pop('loggedin', None)
session.pop('id', None)
session.pop('username', None) return
redirect(url for('login'))
@app.route('/register', methods =['GET', 'POST']) def
register():
msg = ''
if request.method == 'POST' and 'username' in request.form and 'password'
in request.form and 'email' in request.form :
username = request.form['username']
password = request.form['password'] email =
request.form['email']
cursor = mysql.connection.cursor(MySQLdb.cursors.DictCursor)
cursor.execute('SELECT * FROM accounts WHERE username = % s',
(username, ))
account = cursor.fetchone()
```

```
if account:
msg = 'Account already exists !'
elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
msg = 'Invalid email address !' elif not re.match(r'[A-Za-z0-
msg = 'Username must contain only characters
elif not username or not password or not email:
msg = 'Please fill out the form !'
else:
cursor.execute('INSERT INTO accounts VALUES (NULL, %
s, % s, % s)', (username, password, email, ))
mysql.connection.commit()
msg = 'You have successfully registered !'
elif request.method == 'POST':
msg = 'Please fill out the form !' return
render template('register.html', msg = msg)
9]+', username):
and numbers !
```