Nalaiya Thiran IBM Assignment – 02

Rohit A S

195001308

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

2. Performing update & delete queries

INSERT INTO USERS VALUES ('NAVEEN','naveenmr13122001@gmail.com','12344','1234');

INSERT INTO USERS VALUES ('ROHIT','rohit@gmail.com','195001305','4321');

INSERT INTO USERS VALUES ('SAI','sai@gmail.com','195001309','5678'); INSERT

INTO USERS VALUES

('SREENIDTHI','sreenidthi@gmail.com','195001310','6789')

USERNAME	EMAIL	ROLLNUMBER	PASSWORD
NAVEEN	naveenmr13122001@gmail.com	12344	1234
ROHIT	rohit@gmail.com	195001305	4321
SAI	sai@gmail.com	195001309	5678
SREENIDTHI	sreenidthi@gmail.com	195001310	6789

DELETE FROM USERS WHERE USERNAME = 'SREENIDTHI';

USERNAME	↑↓	EMAIL	ROLLNUMBER	PASSWORD
NAVEEN	USERNAME	nmr13122001@gmail.com	12344	1234
ROHIT		rohit@gmail.com	195001305	4321
SAI		sai@gmail.com	195001309	5678

UPDATE USERS SET EMAIL = 'rohita@gmail.com' WHERE USERNAME = 'ROHIT';

USERNAME	EMAIL	ROLLNUMBER	PASSWORD
NAVEEN	naveenmr13122001@gmail.com	12344	1234
ROHIT	rohita@gmail.com	195001305	4321
SAI	sai@gmail.com	195001309	5678

3. Connect python code to db2

```
conn = ibm_db.connect("DATABASE=bludb;"
 "HOSTNAME=b0aebb68-94fa-46ec-a1fc-
1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;"
 "PORT=31249;"
"SECURITY=SSL;"
"SSLServerCertificate=DigitCertGlobalRootCA.crt;"
 "UID=nsc03399;"
"PWD="Naveen@01;", "", "")
```

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

```
App.py

from flask import Flask, render_template, request, redirect, url_for, session

import ibm_db

import re

app = Flask(__name__)

app.secret_key = 'Zenik'

conn = ibm_db.connect("DATABASE=bludb;"
 "HOSTNAME=b0aebb68-94fa-46ec-a1fc-
1c999edb6187.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;"
 "PORT=31249;"
"SECURITY=SSL;"
"SSLServerCertificate=DigitCertGlobalRootCA.crt;"
 "UID=nsc03399;"
"PWD="Naveen@01;", "", "")

@app.route('/')

@app.route('/home')

def home():
 return render_template('home.html', title='Home', msg=" ")

@app.route('/dashboard')

def dashboard():
 sql = "SELECT * FROM USERS WHERE username =?"

 stmt = ibm_db.prepare(conn, sql)

 ibm_db.bind_param(stmt, 1, session['username'])

 ibm_db.execute(stmt)

 account = ibm_db.fetch_assoc(stmt)

 return render_template('dashboard.html', title='Dashboard', account=account)

@app.route('/logout')

def logout():
 session.pop('Loggedin', None)

 session.pop('id', None)

 session.pop('username', None)

 return redirect('/')
```

```python
@app.route('/login', methods=['GET', 'POST'])
def login():
 global userid
 msg = " "
 if request.method == "POST":
 username = request.form['username']
 password = request.form['password']
 sql = "SELECT * FROM USERS WHERE username =? AND password =?" stmt
 = ibm_db.prepare(conn, sql)
 ibm_db.bind_param(stmt, 1, username)
 ibm_db.bind_param(stmt, 2, password)
 ibm_db.execute(stmt)
 account = ibm_db.fetch_assoc(stmt)
 print(account)
 if account:
 session['Loggedin'] = True
 session['id'] = account['USERNAME']
 userid = account['USERNAME']
 session['username'] = account['USERNAME']
 return redirect('/dashboard')
 else:
 msg = "Incorrect login credentials"
 return render_template('login.html', title='Login', msg=msg)
@app.route('/register', methods=['GET', 'POST'])
def register():
 msg = " "
 if request.method == "POST":
 username = request.form['username']
 email = request.form['email']
 password = request.form['password']
 password1 = request.form['password1']
 rollno = request.form['rollno']
```

```python
        sql = "SELECT * FROM USERS WHERE username =? or email=? "
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.bind_param(stmt, 2, email)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print(account)
        if account:
        msg = "Account already exists"
        elif password1 != password:
        msg = "re-entered password doesnt match"
        elif not re.match(r'[A-Za-z0-9]+', username):
        msg = "Username should be only alphabets and numbers"
        else:
        sql = "INSERT INTO USERS VALUES (?,?,?,?)"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.bind_param(stmt, 2, email)
        ibm_db.bind_param(stmt, 3, rollno)
        ibm_db.bind_param(stmt, 4, password)
        ibm_db.execute(stmt)
        return redirect('/login')
        return render_template('register.html', msg=msg, title="Register")
if __name__ == '__main__':
    app.run()
```

Fin App

Welcome to the fin app

Login
Register

Log In

bhavika

Username

••••••••

Password

Login

Don't have an account? Register

Register

bhavika

Your Username

Your Email

Your Roll Number

••••••

Password

Repeat your password

Register