**Assignment -2**

Python to db2.

|  |  |
| --- | --- |
| Assignment Date | 19 September 2022 |
| Student Name | DEVANAND V |
| Student Roll Number | 7376191CS151 |
| Maximum Marks | 2 Marks |

**Question:**

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

2. Perform UPDATE, DELETE Queries with user table

3. Connect python code to db2.

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

**Program:**

**db2\_config.py**

The configrations for db2.

import ibm\_db

import sys

conn = ''

def get\_connection():

    db\_name = "bludb"

    db\_host\_name = "6667d8e9-9d4d-4ccb-ba32-21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud"

    db\_port = "30376"

    db\_protocol = "tcpip"

    db\_username = "jzt12971"

    db\_password = "dAmITuIrVMzd1jkp"

    try:

        connection\_str = f"database={db\_name};hostname={db\_host\_name};port={db\_port};protocol={db\_protocol};uid={db\_username};pwd={db\_password};security=ssl"

        # conn\_str = f"DATABASE={db\_name};HOSTNAME={db\_host\_name};PORT={db\_port};SECURITY=SSL;SSLServerCertificate=<FULL\_PATH\_TO\_SERVER\_CERTIFICATE>;UID={db\_username};PWD={db\_password}"

        connection = ibm\_db.connect(connection\_str, '', '')

        # sql = " INSERT INTO  'JZT12971'.'USERS' ('ID','USERNAME','EMAIL','ROLLNO','PASSWORD') VALUES( ?, ?, ?, ?,?)"

        return connection

    except:

        print("Connection failed:", ibm\_db.conn\_errormsg())

        sys.exit(1)

**db2\_operation.py**

To perform db2 operations.

import ibm\_db

import json

def insert\_user\_data(conn, details):

    sql = 'INSERT INTO  "JZT12971"."USERS" ("ID","USERNAME","EMAIL","ROLLNO","PASSWORD") VALUES(seq\_user.nextval, ?, ?, ? ,?);'

    stmt = ibm\_db.prepare(conn, sql)

    # for i in range(1, len(details)):

    #      print(i, details[i - 1])

    for i in range(0, len(details)):

        ibm\_db.bind\_param(stmt, i + 1, details[i])

    ibm\_db.execute(stmt)

def isAuthenticate(conn, username, 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)

    return ibm\_db.fetch\_assoc(stmt)

def isUserExists(conn, username):

    sql = "SELECT \* FROM users where username = ?;"

    stmt = ibm\_db.prepare(conn, sql)

    ibm\_db.bind\_param(stmt, 1, username)

    ibm\_db.execute(stmt)

    acc = ibm\_db.fetch\_assoc(stmt)

    if acc == False:

        return acc

    else:

        print(acc)

        return (username == acc['USERNAME'].strip())

    # print(acc['USERNAME'], ".")

    # print(username == acc['USERNAME'].strip())

    return acc

    # return ibm\_db.fetch\_assoc(stmt)

**auto\_increment\_id.sql**

To increment user id by 1 for every new user entry.

CREATE SEQUENCE seq\_user

MINVALUE 1

START WITH 1

INCREMENT BY 1

CACHE 100;

**app.py**

import traceback

from flask import Flask, request, render\_template, session, redirect, url\_for, flash

from db2\_config import get\_connection

from db2\_operation import insert\_user\_data, isAuthenticate, isUserExists

from flask\_debugtoolbar import DebugToolbarExtension

app = Flask(\_\_name\_\_)

app.config['SECRET\_KEY'] = 'a'

conn = get\_connection()

@app.route('/')

@app.route('/registration', methods=["GET", "POST"])

def registration():

    msg = ""

    details = []

    if request.method == "POST":

        name = request.form.get("name")

        email = request.form.get("email")

        rollno = request.form.get("rollno")

        password = request.form.get("password")

        details = [name, email, rollno, password]

        # conn = get\_connection(details=details)

        try:

            acc = isUserExists(conn=conn, username=name)

            print("register acc:", acc)

            if acc == False:

                insert\_user\_data(conn=conn, details=details)

                return redirect(url\_for('login'))

            else:

                msg = f"User {name} already exists"

                flash(msg)

        except Exception as exp:

            print("insert failed", exp.\_\_traceback\_\_)

            traceback.print\_exc()

            # return "Your details is " + name + " " + email + " " + number

    # return render\_template("form.html", data = details)

    return render\_template('register.html')

# details = []

@app.route('/login', methods=["GET", "POST"])

def login():

    global user\_id

    msg = ''

    print("lg")

    if request.method == 'POST':

        print("post method")

        username = request.form.get("name")

        password = request.form.get('password')

        print(username, password)

        try:

            account = isAuthenticate(conn=conn, username=username, password=password)

            print("statement executed")

            # account = isAuthenticate(username=username, password=password)

            print("login satus", account)

            if (account):

                print(f"acc user name = {account['USERNAME']}")

                session['loggedin'] = True

                session['id'] = account['USERNAME']

                session['USERNAME'] = account['USERNAME']

                user\_id = account['USERNAME']

                msg = "Login sucessfull"

                return redirect(url\_for('welcome'))

            else:

                msg = "login failed"

            print(msg)

            flash(msg)

        except Exception as exp:

            print("Authentication failed", exp.\_\_traceback\_\_)

        # account = isAuthenticate(username=username, password=password)

      #   return msg

    return render\_template('login.html', msg=msg)

@app.route('/welcome')

def welcome():

    return f"<h1>Welcome {user\_id}!</h1>"

if \_\_name\_\_ == '\_\_main\_\_':

    # toolbar = DebugToolbarExtension(app)

    app.debug = True

    app.run()

**Templates:**

**base.html**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <title>{% block title %}{% endblock %}</title>

    <link href="{{ url\_for('static', filename='css/stylesheet.css') }}" rel="stylesheet">

</head>

<body>

{% block main %}

{% endblock %}

</body>

</html>

**Login.html**

{% extends 'base.html' %}

{% block title %}Login{% endblock %}

{% block main %}

<!-- <h1>{{msg}}</h1> -->

{% include 'flash.html' %}

<form action="{{ url\_for('login')}}" method="post">

    <fieldset>

        <legend>Login Form:</legend>

        <label for="name">User Name:</label><br>

        <input type="text" name="name" id="name"><br>

        <label for="password">Password:</label><br>

        <input type="password" name="password" id="password"><br>

        <button type="submit">submit</button>

    </fieldset>

</form>

{% endblock %}

**register.html**

{% extends 'base.html' %}

{% block title %}Register{% endblock %}

{% block main %}

{% include 'flash.html' %}

<form action="{{ url\_for('registration')}}" method="post">

    <fieldset>

        <legend>Registration Form:</legend>

        <label for="name">User Name:</label><br>

        <input type="text" name="name" id="name"><br>

        <label for="email">Email</label><br>

        <input type="email" name="email" id="email"><br>

        <label for="rollno">Roll No:</label> <br>

        <input type="text" name="rollno" id="rollno"><br>

        <label for="password">password</label><br>

        <input type="password" name="password" id="password"><br>

        <button type="submit">submit</button>

    </fieldset>

</form>

<div>

    <!-- <a href="{{url\_for('login')}}"><button>Login</button></a> -->

</div>

{% endblock %}

**flash.html**

<div>

    {% for message in get\_flashed\_messages() %}

    <h2>{{ message }}</h2>

    {% endfor %}

</div>

**Login Page:**

{% extends 'base.html' %}

{% block title %}Login{% endblock %}

{% block main %}

<!-- <h1>{{msg}}</h1> -->

{% include 'flash.html' %}

<form action="{{ url\_for('login')}}" method="post">

    <fieldset>

        <legend>Login Form:</legend>

        <label for="name">User Name:</label><br>

        <input type="text" name="name" id="name"><br>

        <label for="password">Password:</label><br>

        <input type="password" name="password" id="password"><br>

        <button type="submit">submit</button>

    </fieldset>

</form>

{% endblock %}

**Static:**

**Stylesheet.css**

\*, \*:before, \*:after {

    -moz-box-sizing: border-box;

    -webkit-box-sizing: border-box;

    box-sizing: border-box;

  }

  body {

    font-family: 'Nunito', sans-serif;

    color: #384047;

  }

  form {

    max-width: 300px;

    margin: 10px auto;

    padding: 10px 20px;

    background: #f4f7f8;

    border-radius: 8px;

  }

  h1,h2 {

    margin: 0 0 30px 0;

    text-align: center;

  }

  input[type="text"],

  input[type="password"],

  input[type="date"],

  input[type="datetime"],

  input[type="email"],

  input[type="number"],

  input[type="search"],

  input[type="tel"],

  input[type="time"],

  input[type="url"],

  textarea,

  select {

    background: rgba(255,255,255,0.1);

    border: none;

    font-size: 16px;

    height: auto;

    margin: 0;

    outline: 0;

    padding: 15px;

    width: 100%;

    background-color: #e8eeef;

    color: #8a97a0;

    box-shadow: 0 1px 0 rgba(0,0,0,0.03) inset;

    margin-bottom: 30px;

  }

  input[type="radio"],

  input[type="checkbox"] {

    margin: 0 4px 8px 0;

  }

  select {

    padding: 6px;

    height: 32px;

    border-radius: 2px;

  }

  button {

    padding: 19px 39px 18px 39px;

    color: #FFF;

    background-color: #4bc970;

    font-size: 18px;

    text-align: center;

    font-style: normal;

    border-radius: 5px;

    width: 100%;

    border: 1px solid #3ac162;

    border-width: 1px 1px 3px;

    box-shadow: 0 -1px 0 rgba(255,255,255,0.1) inset;

    margin-bottom: 10px;

  }

  fieldset {

    margin-bottom: 30px;

    border: none;

  }

  legend {

    font-size: 1.4em;

    margin-bottom: 10px;

  }

  label {

    display: block;

    margin-bottom: 8px;

  }

  label.light {

    font-weight: 300;

    display: inline;

  }

  .number {

    background-color: #5fcf80;

    color: #fff;

    height: 30px;

    width: 30px;

    display: inline-block;

    font-size: 0.8em;

    margin-right: 4px;

    line-height: 30px;

    text-align: center;

    text-shadow: 0 1px 0 rgba(255,255,255,0.2);

    border-radius: 100%;

  }

  @media screen and (min-width: 480px) {

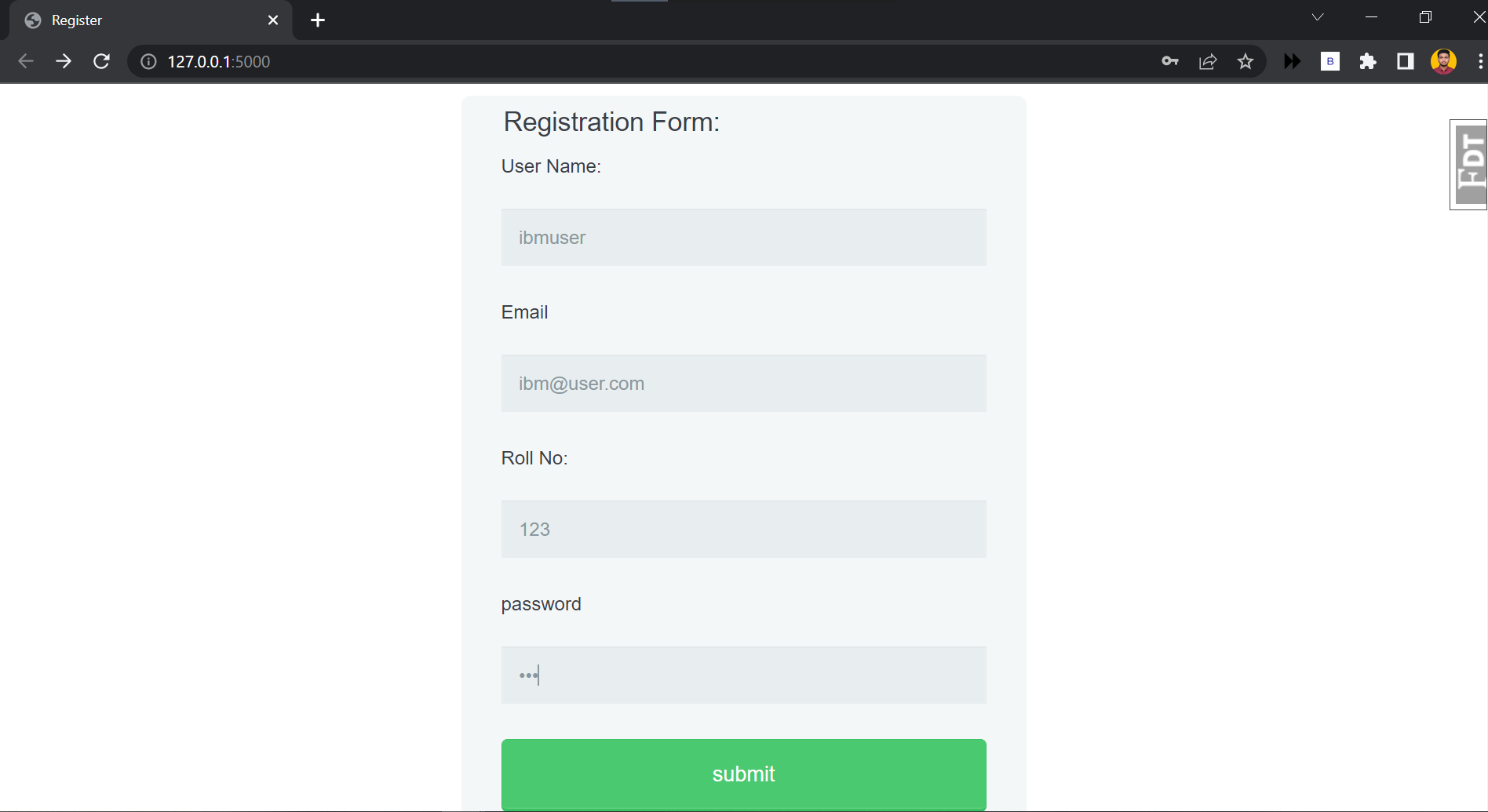
    form {

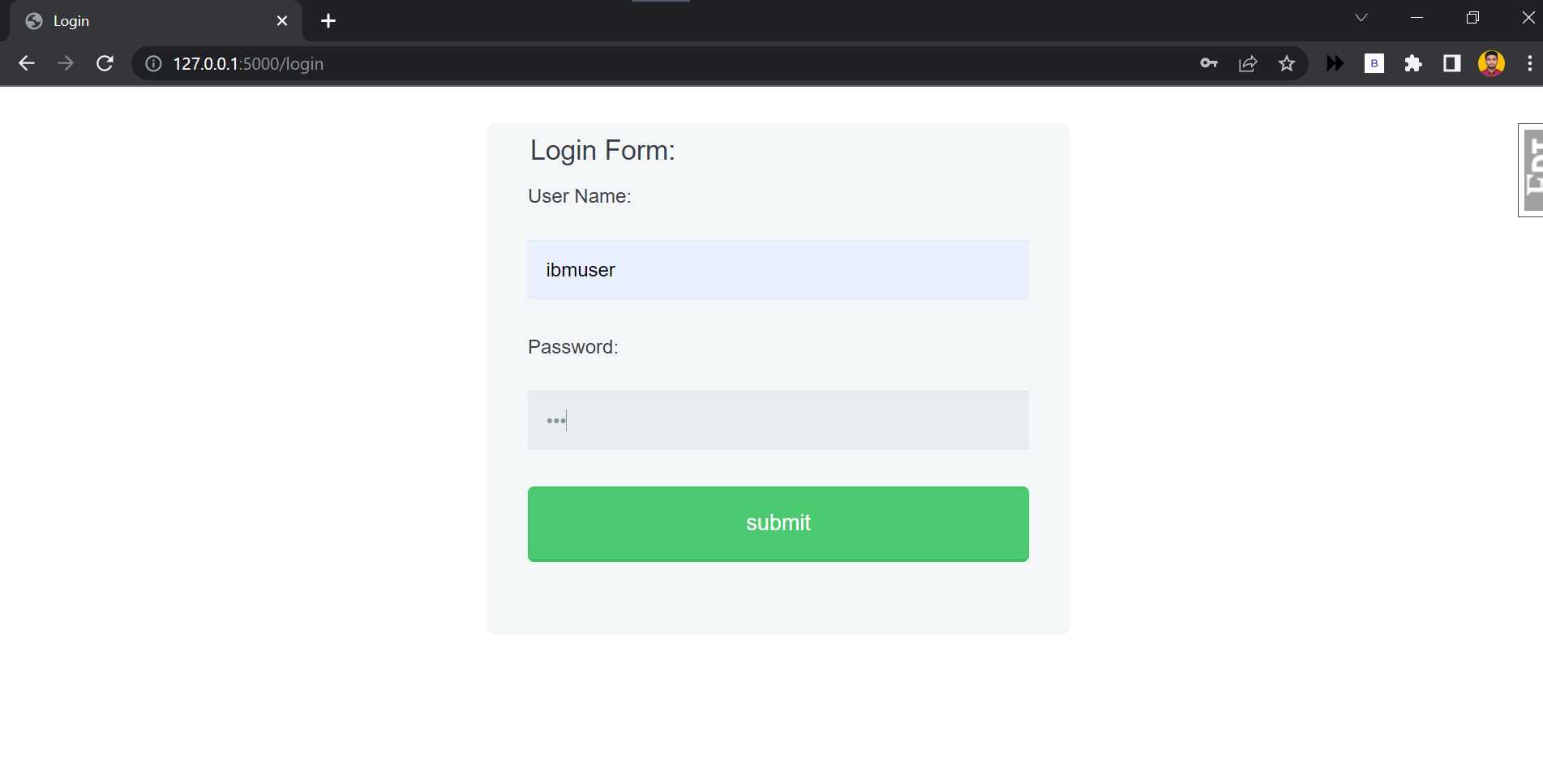
      max-width: 480px;

    }

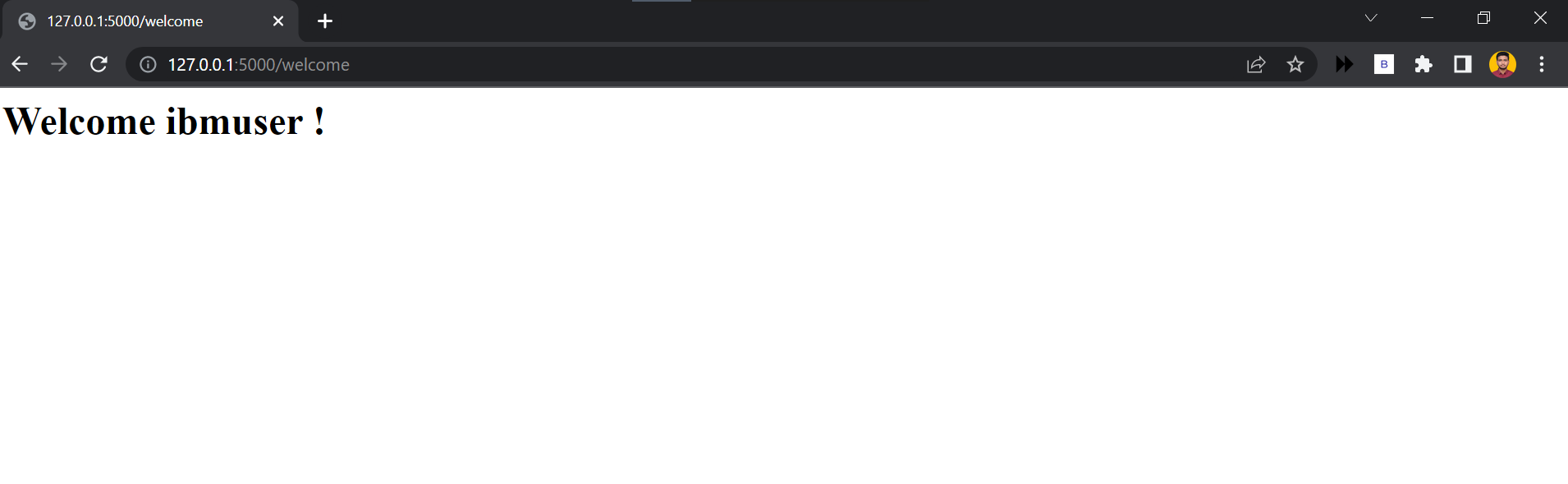
  }

**Registration page**

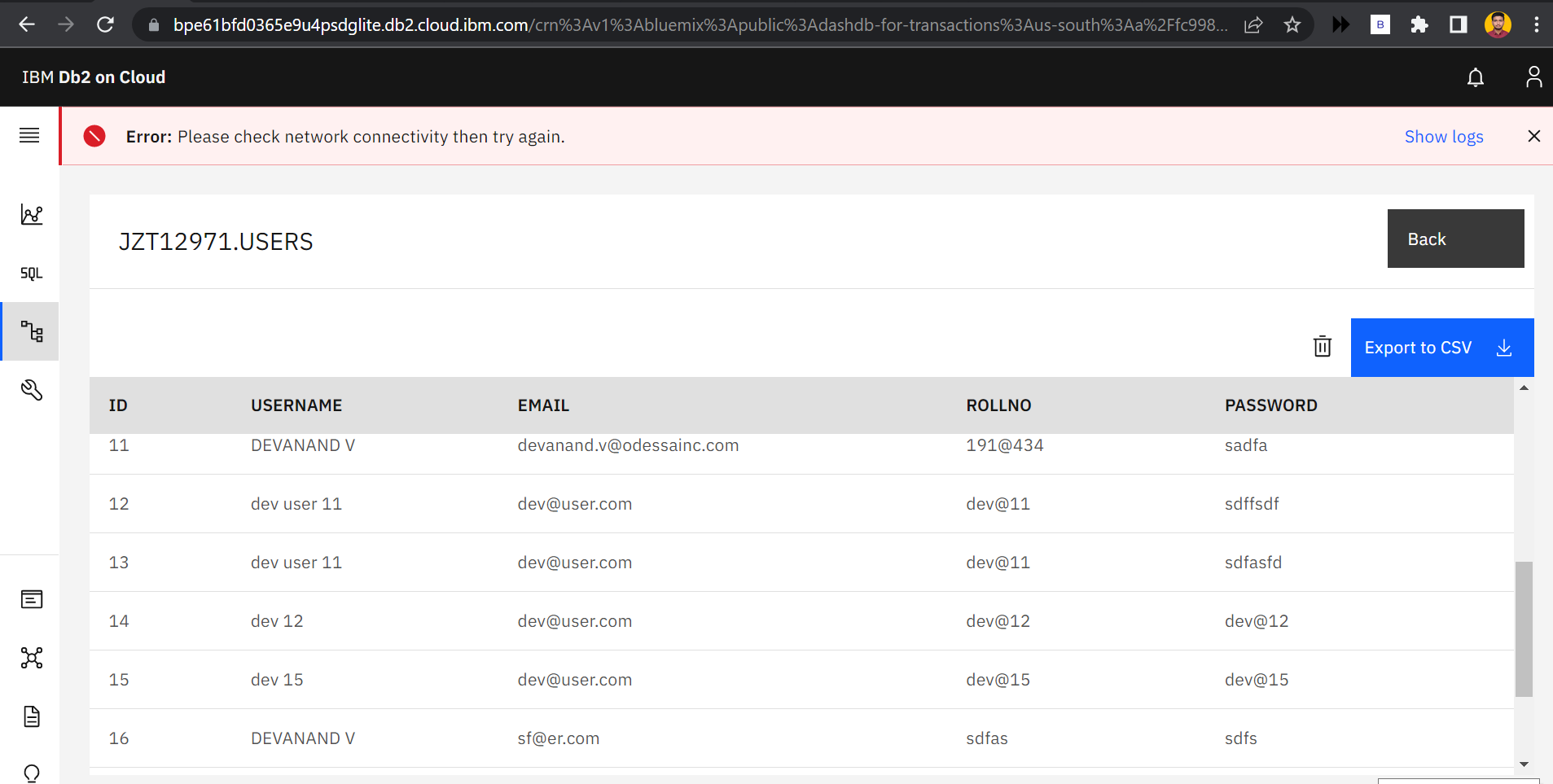




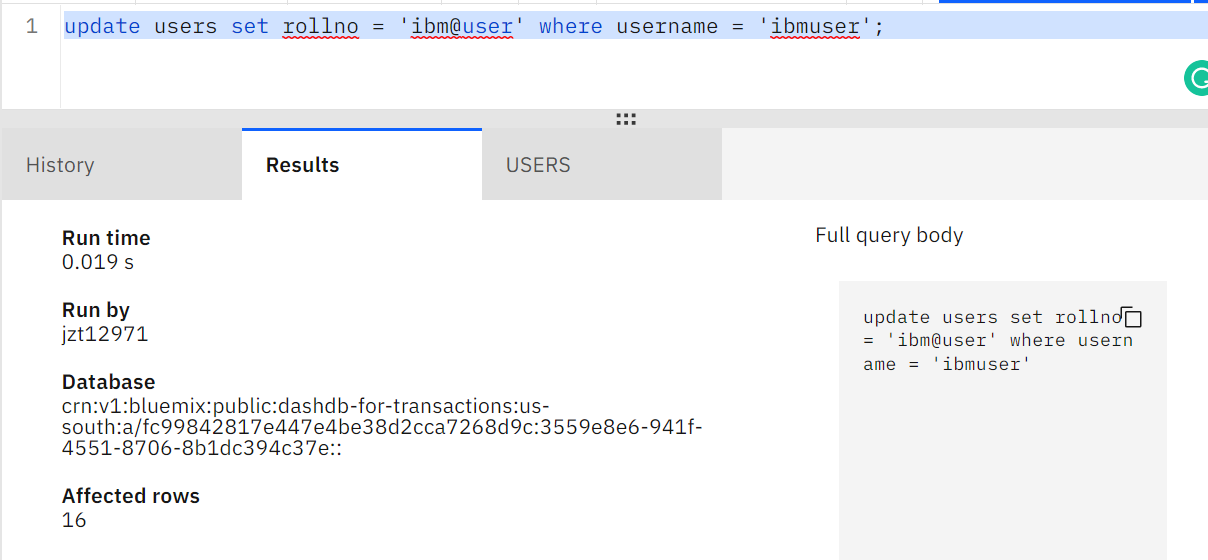
**Welcome page:**



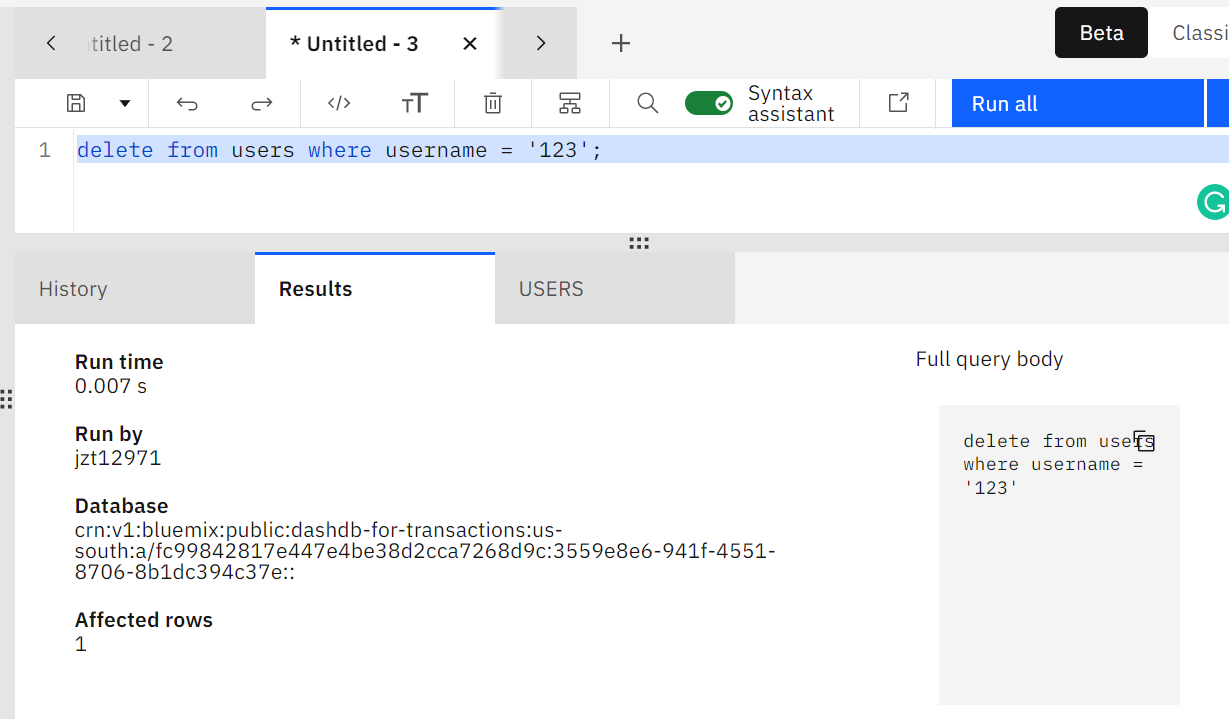
**IBM Cloud user table.**



**Update statement:**

****

**Delete statement:**

****