

Assignment-3

Team id	PNT2022TMID49572
PROJECT NAME	PERSONAL EXPENSE TRACKER APPLICATION
NAME	SAFEER ALI S
ROLL NO	950019104039

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

```
CREATE TABLE user(email varchar(50) not null,  
                    username varchar(20) not null,  
                    roll_number int not null, password  
                    varchar(20) not null)
```

```
INSERT INTO user(email, username, roll_number, password)  
VALUES('msabeer208@gmail.com','safeer12','4039','safeer')
```

2.Perform UPDATE,DELETE Queries with user table

```
UPDATE user SET username='safeer' WHERE email='msabeer208@gmail.com'  
DELETE FROM user WHERE username = 'safeer'
```

3.Connect python code to db2.

```
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-  
8aba-9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;  
PORT=30699;  
Security=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=nkh40273;  
PWD=hHZYtlj4Daek5Od3;","")
```

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.

login.html

```
<html>

  <head>

    <meta charset="UTF-8">

    <title> Login </title>

    <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

  </head>

  <body></br></br></br></br></br>

    <div align="center">

      <div align="center" class="border">

        <div class="header">

          <h1 class="word">Login</h1>

        </div></br></br></br>

        <h2 class="word">

          <form action="{{ url_for('login') }}" method="post">

            <div class="msg">{{ msg }}</div>

            <input id="username" name="username" type="text"
placeholder="Enter Your Username" class="textbox"/></br></br>

            <input id="password" name="password"
type="password" placeholder="Enter Your Password" class="textbox"/></br></br></br>

            <input type="submit" class="btn" value="Sign
In"></br></br>

          </form>

        </h2>

        <p class="bottom">Don't have an account? <a class="bottom"
href="{{ url_for('register') }}"> Sign Up here</a></p>

      </div>

    </div>
```

```
        </div>

    </body>

</html>
```

register.html:

```
<html>

    <head>

        <meta charset="UTF-8">

        <title> Register </title>

        <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

    </head>

    <body></br></br></br></br></br>

        <div align="center">

            <div align="center" class="border">

                <div class="header">

                    <h1 class="word">Register</h1>

                    </div></br></br></br>

                    <h2 class="word">

                        <form action="{{ url_for('register') }}" method="post">

                            <div class="msg">{{ msg }}</div>

                                <input id="username" name="username" type="text"
placeholder="Enter Your Username" class="textbox"/></br></br>

                                <input id="password" name="password"
type="password" placeholder="Enter Your Password" class="textbox"/></br></br>

                                <input id="email" name="email" type="text"
placeholder="Enter Your Email ID" class="textbox"/></br></br>

                                <input type="submit" class="btn" value="Sign
Up"></br>

                            </form>

                        </h2>

                        <p class="bottom">Already have an account? <a class="bottom"
href="{{ url_for('login') }}"> Sign In here</a></p>
```

```
        </div>
    </div>
</body>
</html>
```

index.html:

```
<html>
    <head>
        <meta charset="UTF-8">
        <title> Index </title>
        <link rel="stylesheet" href="{{ url_for('static', filename='style.css') }}">

    </head>
    <body><br><br><br><br><br>
        <div align="center">
            <div align="center" class="border">
                <div class="header">
                    <h1 class="word">Index</h1>
                </div><br><br><br>
                <h1 class="bottom">
                    Hi {{ session.username }}!!<br><br> Welcome to Our
page...

                </h1><br><br><br>
                <a href="{{ url_for('logout') }}" class="btn">Logout</a>
            </div>
        </div>
    </body>
</html>
```

app.py:

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

import ibm_db import re

app = Flask(__name__) app.secret_key

= 'a'

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-
9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;Security=SSL
;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=nkh40273;PWD=hHZYtlj4Daek5O
d3;","")

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

    if request.method == 'POST' and 'username' in request.form and 'password' in
request.form:

        username = request.form['username']

password = request.form['password']

        stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE username = ? AND
password = ?')

        ibm_db.bind_param(stmt,1,username)

ibm_db.bind_param(stmt,2,password)        ibm_db.execute(stmt)

        account = ibm_db.fetch_assoc(stmt)

if account:

        session['loggedin'] = True

        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']
        stmt = ibm_db.prepare(conn,'SELECT * FROM users WHERE username = ?')
ibm_db.bind_param(stmt,1,username)    ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
if account:
        msg = 'Account already exists !'    elif not
re.match(r'^[^\s@]+@[^\s@]+\.[^\s@]+', email):
        msg = 'Invalid email address !'    elif
not re.match(r'[A-Za-z0-9]+', username):
        msg = 'Username must contain only characters and numbers !'
elif not username or not password or not email:
        msg = 'Please fill out the form !'
else:
        prep_stmt = ibm_db.prepare(conn,"INSERT INTO users VALUES(?, ?, ?)")
ibm_db.bind_param(prepare_stmt, 1, username)
ibm_db.bind_param(prepare_stmt, 2, email)    ibm_db.bind_param(prepare_stmt, 3,
password)    ibm_db.execute(prepare_stmt)

```

```
msg = 'You have successfully registered !'

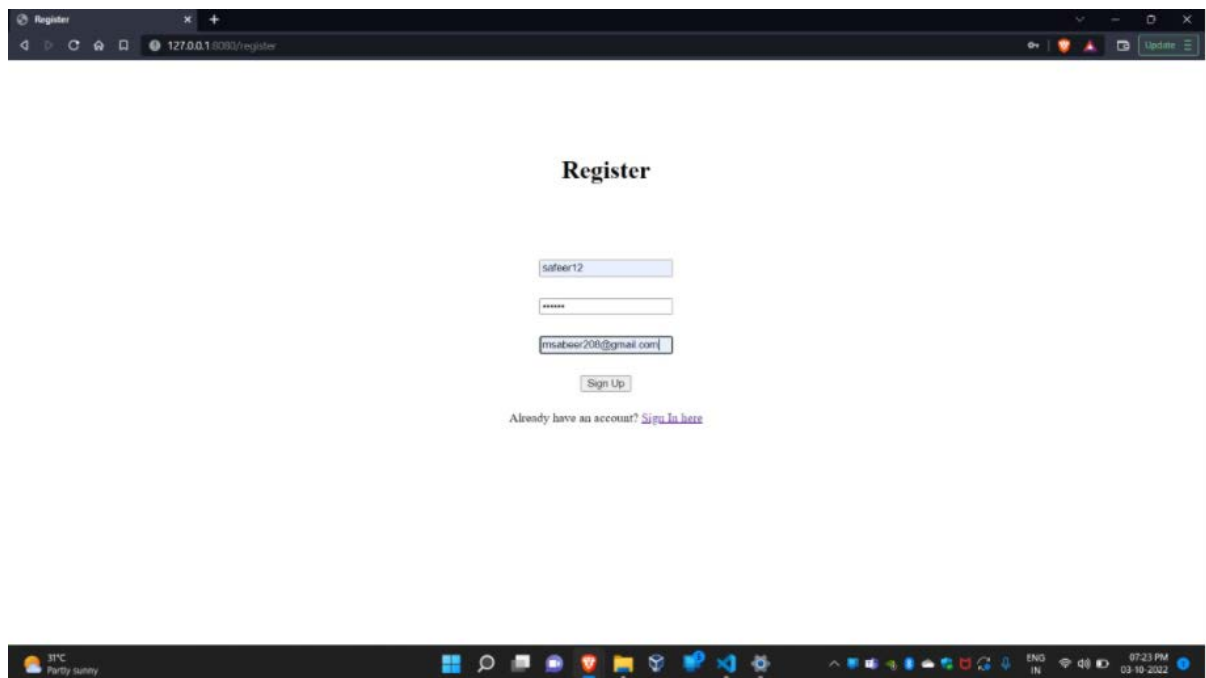
elif request.method == 'POST':    msg =

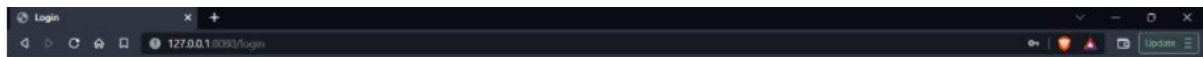
'Please fill out the form !'

return render_template('register.html', msg = msg)

if __name__ == '__main__':
    app.debug = True
    app.run(host='0.0.0.0',port=8080)
```

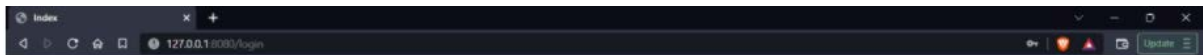
Output:





Login

Don't have an account? [Sign Up here](#)



Index

Hi safeer12!!

Welcome to Our page...

[Logout](#)



IBM

Service Details - IBM Cloud

IBM Db2 on Cloud

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fdd933ae5700d411c8c51a47cf7d01...

Gmail YouTube Maps

IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

Find schemas or tables

Refresh

Tables

New table

Name

Schema

Properties

USERS

NKH40273

...

Total: 1, selected: 0

Table definition

USERS

Approximate 2 rows (32.0 KB)
Updated on 2022-10-03 13:47:19

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	N	20	0
EMAIL	VARCHAR	N	50	0
PASSWORD	VARCHAR	N	20	0

View data

Type here to search

29°C

19:52

03-10-2022

IBM

Service Details - IBM Cloud

IBM Db2 on Cloud

bpe61bfd0365e9u4psdglite.db2.cloud.ibm.com/crn%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aus-south%3Aa%2Fdd933ae5700d411c8c51a47cf7d01...

Gmail YouTube Maps

IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

NKH40273.USERS

Back

Export to CSV

USERNAME	EMAIL	PASSWORD
safer	msabee208@gmail.com	sefer
safer12	msabee208@gmail.com	safer

Type here to search

29°C

19:51

03-10-2022