

Assignment 2 – Personal Expense Tracker

Name: Aarthi N

RollNo: 195001002

Creation of table:

The screenshot shows the IBM Db2 on Cloud interface. The 'Tables' tab is selected, displaying a list of tables in the 'WYM47343' schema. The 'Table definition' panel on the right shows the structure of the 'USER_AARTHIN' table.

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	Y	32	0
PASSWORD	VARCHAR	Y	32	0
EMAIL	VARCHAR	Y	32	0

Buttons: 'New table', 'View data', 'Refresh'.

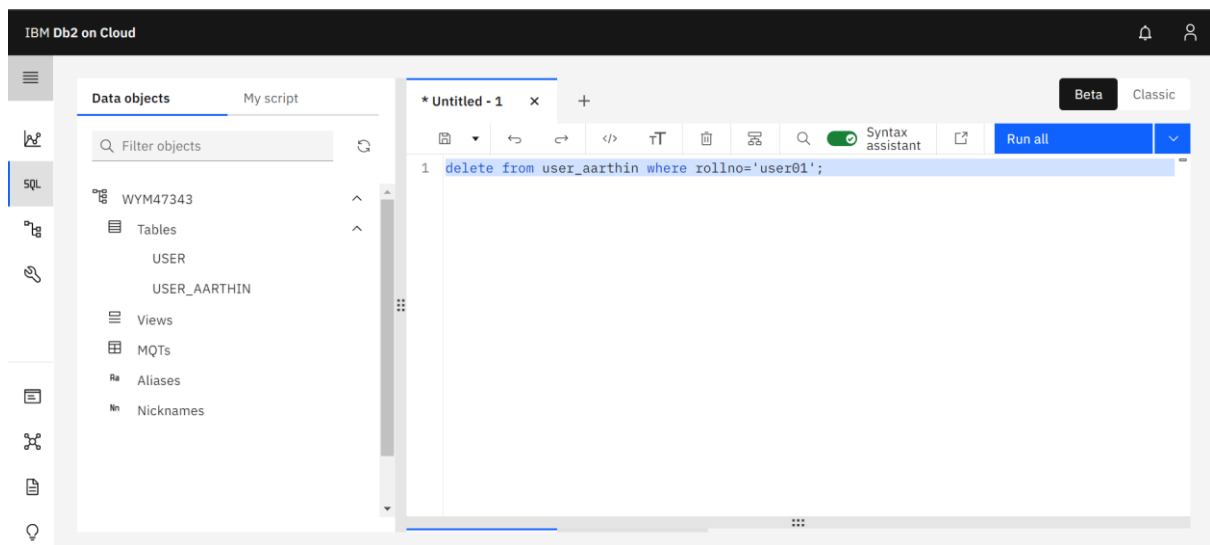
Insertion into USER_AARTHIN table:

The screenshot shows the IBM Db2 on Cloud interface. The 'Data objects' tab is selected, displaying a tree view of the database objects. The 'My script' editor is open, showing a script with four SQL insert statements for the 'USER_AARTHIN' table.

```
1 insert into user_aarthin values('user1','pwd1','user1@gmail.com','user01');
2 insert into user_aarthin values('user2','pwd2','user2@gmail.com','user02');
3 insert into user_aarthin values('user3','pwd3','user3@gmail.com','user03');
4 insert into user_aarthin values('user4','pwd4','user4@gmail.com','user04');
```

Buttons: 'Run all', 'Beta', 'Classic'.

Deleting from USER_AARTHIN table:



The screenshot shows the IBM Db2 on Cloud console displaying the contents of the 'WYM47343.USER_AARTHIN' table. The table has four columns: 'USERNAME', 'PASSOWRD', 'EMAIL', and 'ROLLNO'. The table contains three rows of data. The 'Export to CSV' button is visible in the top right corner of the table view.

USERNAME	PASSOWRD	EMAIL	ROLLNO
user2	pwd2	user2@gmail.com	user02
user3	pwd3	user3@gmail.com	user03
user4	pwd4	user4@gmail.com	user04

Connecting to IBM DB2 from python:

```
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=19af6446-6171-4641-8aba-9dcff8e1b6ff.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30699;Security=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=wym47343;PWD=KRCuyIBtvZY1adNZ;", "
```

Source Code:

App.py

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

app=Flask(__name__, template_folder='resources')
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=wym47343;PWD=KRCuyIBtvZY1adNZ;", "
", "")

@app.route('/', methods=['GET', 'POST'])
def register():
    msg=" "
    if request.method=='POST':
        username=request.form['username']
        password=request.form['password']
        email=request.form['email']
        rollno=request.form['rollno']
        sql="INSERT INTO user_aarthin VALUES(?,?,?,?)"
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.bind_param(stmt,2,username)
        ibm_db.bind_param(stmt,3,rollno)
        ibm_db.bind_param(stmt,4,password)
        ibm_db.execute(stmt);
        msg='Succesfully registered!'
```

```
        return redirect(url_for('login'))
    elif request.method=='GET':
        msg='Please fill the form'
    return render_template('register.html',msg=msg)
```

```
@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 user 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"]
            msg='Logged in successfully'
            return render_template("welcome.html",msg=msg,username=userid)
        else:
            msg='Incorrect username/password'
```

```
        return render_template('login.html', msg=msg)
    elif request.method=='GET':
        return render_template('login.html')
```

```
if __name__=='__main__':
    app.run(host='0.0.0.0')
```

Register.html

```
<html>

<body>

<h2 style="color:blue">{{msg}}</h2>

<form action="/" method="POST">

<fieldset>

    <legend>User Information</legend>

    <h2>Username:</h2> <input type="text" name="username" placeholder="Enter username"
    "><br/><br/>

    <h2>Password:</h2> <input type="password" name="password"><br/><br/>

    <h2>Email:</h2> <input type="email" name="email" placeholder="Enter email"><br/><br/>

    <h2>Roll number:</h2> <input type="text" name="rollno" placeholder="Enter roll
    number"><br/><br/>

    <input type="submit" value="Register">

</fieldset>

</form>

</body>

</html>
```

Login.html

```
<html>

<body>

<h2 style="color:blue">{{msg}}</h2>

<form action="/login" method="POST">

  <fieldset>

    <legend>User Login:</legend>

    <h2>Username:</h2> <input type="text" name="username" placeholder="Enter
username"><br/><br/>

    <h2>Password:</h2> <input type="password" name="password"
placeholder="*****"><br/><br/>

    <input type="submit" value="Login">

  </fieldset>

</form>

</body>

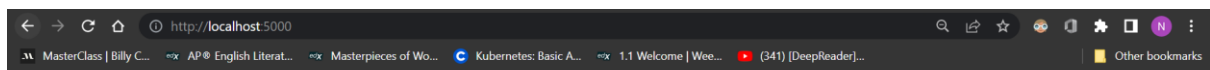
</html>
```

Welcome.html

```
<h1 style="color:blue">Welcome {{username}}!</h1>

<h2>{{msg}}</h2>
```

Register.html



Please fill the form

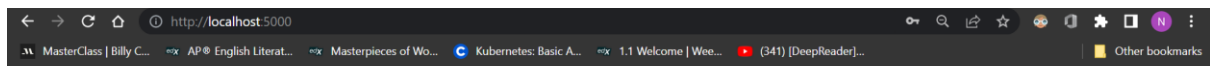
User Information

Username:

Password:

Email:

Roll number:



Please fill the form

User Information

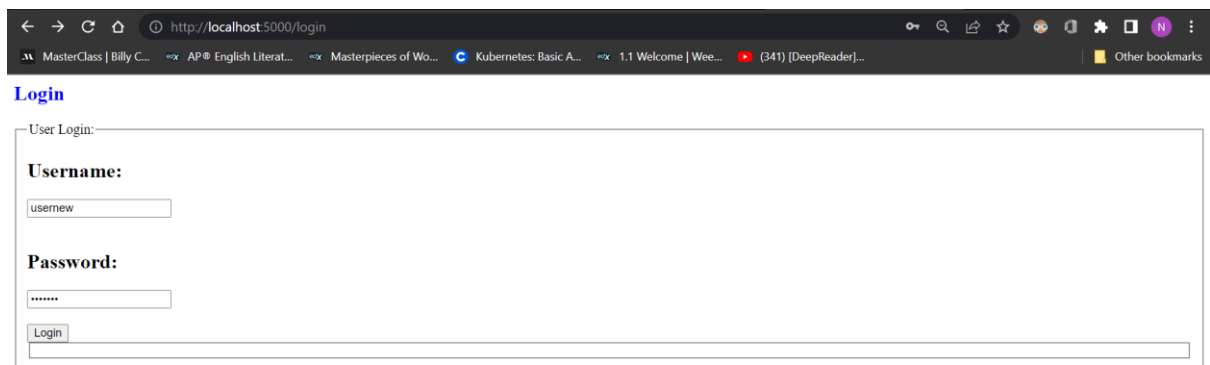
Username:

Password:

Email:

Roll number:

Login.html (correct login)

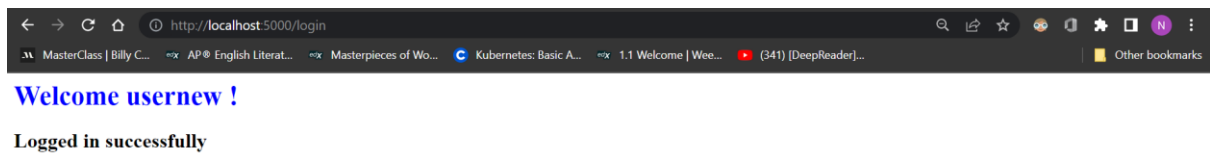


The screenshot shows a web browser window with the address bar displaying `http://localhost:5000/login`. The browser's tab bar shows several open tabs, including "MasterClass | Billy C...", "AP® English Literat...", "Masterpieces of Wo...", "Kubernetes: Basic A...", "1.1 Welcome | Wee...", and "(341) [DeepReader]...". The browser's bookmark bar shows "Other bookmarks".

The main content of the page is a login form titled "User Login:". The form contains the following elements:

- A label "Username:" followed by a text input field containing the text "usernew".
- A label "Password:" followed by a password input field containing eight asterisks "*****".
- A "Login" button.
- A horizontal line below the "Login" button.

Welcome.html



The screenshot shows a web browser window with the address bar displaying `http://localhost:5000/login`. The browser's tab bar shows several open tabs, including "MasterClass | Billy C...", "AP® English Literat...", "Masterpieces of Wo...", "Kubernetes: Basic A...", "1.1 Welcome | Wee...", and "(341) [DeepReader]...". The browser's bookmark bar shows "Other bookmarks".

The main content of the page is a welcome message:

Welcome usernew !

Logged in successfully

IBM Db2 on Cloud

Load Data

Load History

Tables

Views

Indexes

Aliases

MQTs

Sequences

Application objects

WYM47343.USER_AARTHIN

Back

Export to CSV

USERNAME	PASSOWRD	EMAIL	ROLLNO
user2	pwd2	user2@gmail.com	user02
user3	pwd3	user3@gmail.com	user03
user4	pwd4	user4@gmail.com	user04
usernew@gmail.com	usernew	usernew11	usernew

Login.html (incorrect login)

←

→

↺

⌂

http://localhost:5000/login

🔍

🔖

☆

🌐

⚙️

📱

👤

MasterClass | Billy C...

AP® English Literat...

Masterpieces of Wo...

Kubernetes: Basic A...

1.1 Welcome | Wee...

(341) | DeepReader...

Other bookmarks

Login

User Login:


Username:

usernew1

Password:

Login

Login.html (incorrect login)



← → ↻ 🏠 http://localhost:5000/login 🔍 📄 ☆ 🧑 🖨️ ⚙️ 📁 N ⋮

MasterClass | Billy C... AP® English Literat... Masterpieces of Wo... Kubernetes: Basic A... 1.1 Welcome | Wee... (341) [DeepReader]... Other bookmarks

Incorrect username/password

User Login:

Username:

Password:

Login