Assignment - 3

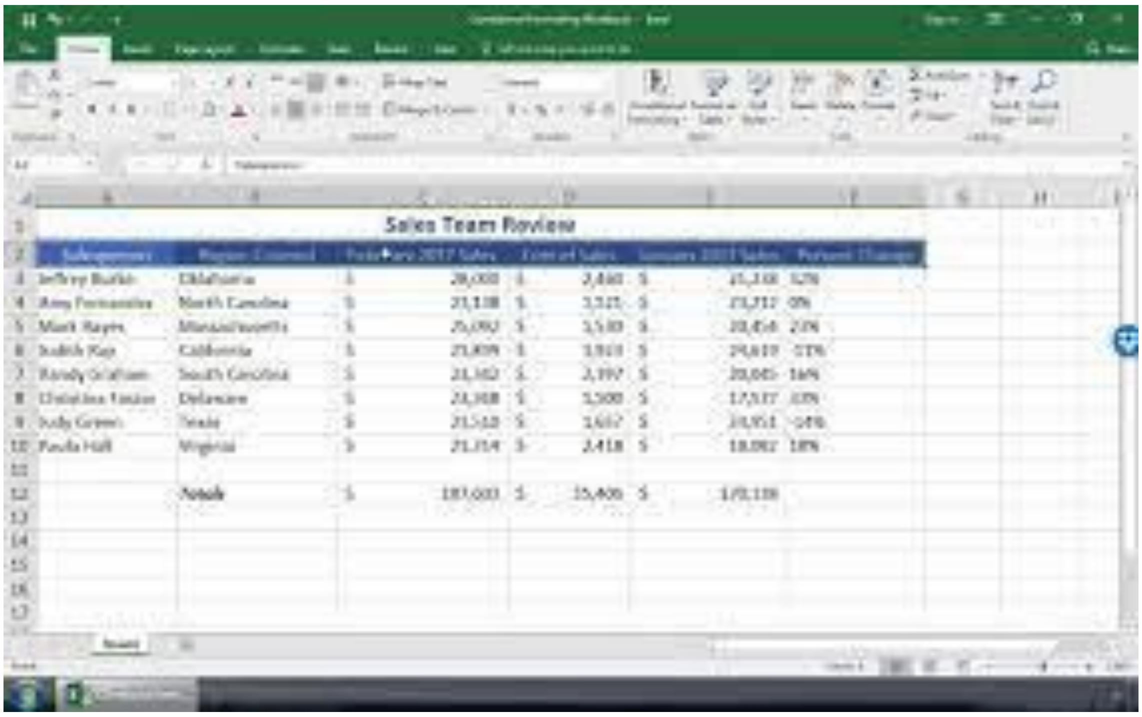
Assignment Date	29 September 2022
Student Name	Sameer Mohammed.S
Student Roll No	812419104056
Maximum Marks	2 Marks

Question-1:

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

Solution:

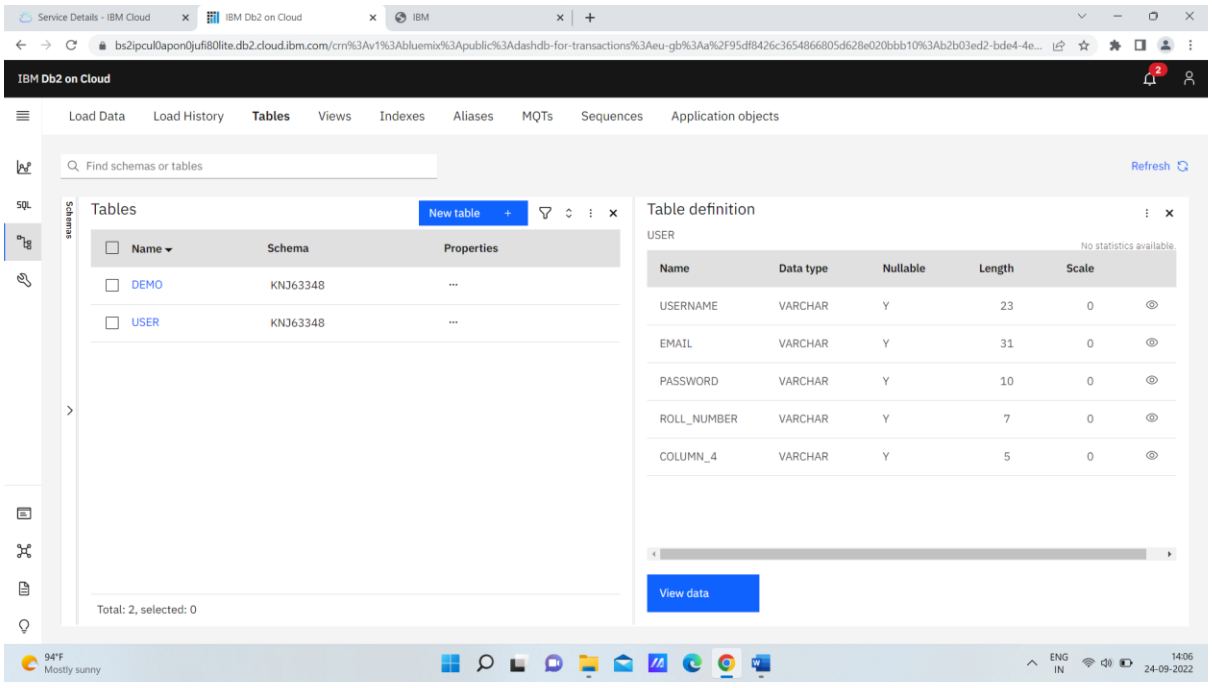
Excel comma file



Sales Team Review

Salesperson	Region Covered	February 2017 Sales		Cost of Sales		January 2017 Sales		Percent Change
Jeffrey Burke	Oklahoma	$	28,000	$	2,460	$	21,318	32%
Amy Fernandez	North Carolina	$	23,138	$	3,121	$	23,717	0%
Mark Hayes	Massachusetts	$	25,092	$	3,510	$	20,458	23%
Judith Kay	California	$	21,809	$	3,910	$	29,619	-27%
Randy Graham	South Carolina	$	23,302	$	2,197	$	20,085	16%
Christina Taylor	Delaware	$	24,308	$	3,500	$	17,517	39%
Judy Green	Texas	$	21,510	$	3,617	$	30,951	-14%
Paula Hall	Virginia	$	21,714	$	2,418	$	18,092	18%
	Totals	$	187,003	$	25,406	$	170,138	

User table in Db2


IBM Db2 on Cloud

Tables

Name	Schema	Properties
DEMO	KNJ63348	...
USER	KNJ63348	...

Total: 2, selected: 0

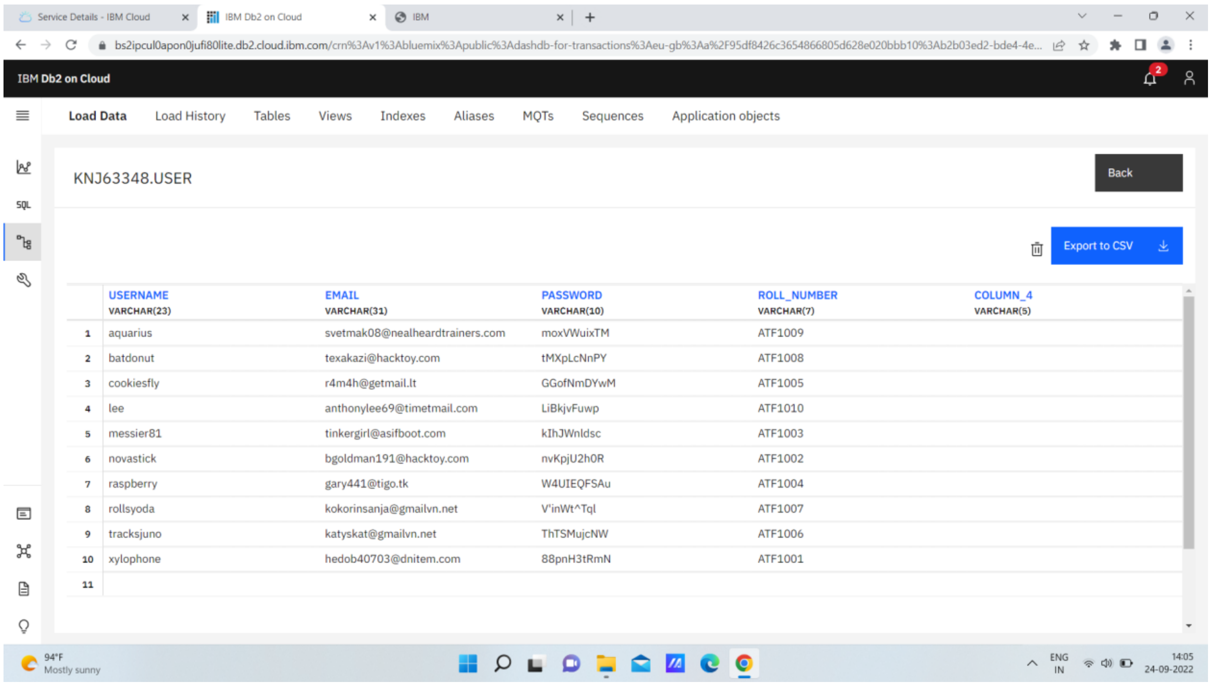
Table definition

USER

No statistics available

Name	Data type	Nullable	Length	Scale
USERNAME	VARCHAR	Y	23	0
EMAIL	VARCHAR	Y	31	0
PASSWORD	VARCHAR	Y	10	0
ROLL_NUMBER	VARCHAR	Y	7	0
COLUMN_4	VARCHAR	Y	5	0

Data in User table


IBM Db2 on Cloud

KNJ63348.USER

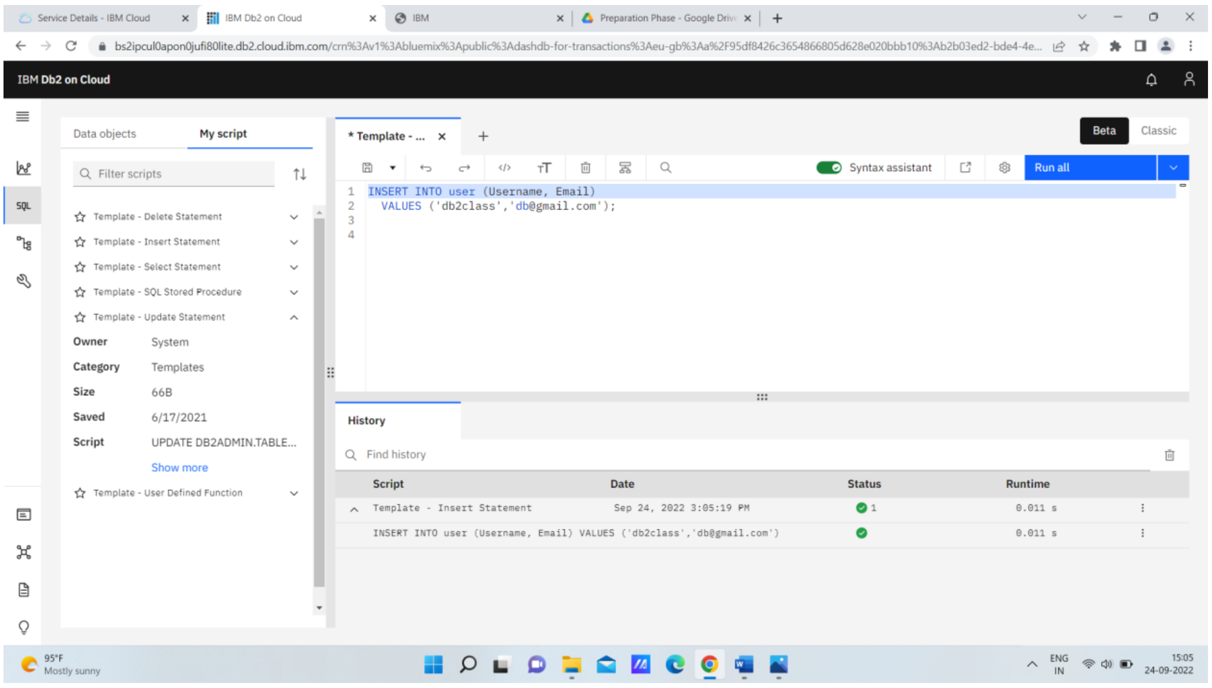
	USERNAME VARCHAR(23)	EMAIL VARCHAR(31)	PASSWORD VARCHAR(10)	ROLL_NUMBER VARCHAR(7)	COLUMN_4 VARCHAR(5)
1	aquarius	svetmak08@nealheardtrainers.com	moxVWuixTM	ATF1009	
2	batdonut	texakazi@hacktoy.com	tMXpLcNnPY	ATF1008	
3	cookiesfly	r4m4h@getmail.lt	GGofNmDYwM	ATF1005	
4	lee	anthonylee69@timetmail.com	LiBkjvFuwp	ATF1010	
5	messier81	tinkergirl@asifboot.com	kIhJWnldsc	ATF1003	
6	novastick	bgoldman191@hacktoy.com	nvKpjU2h0R	ATF1002	
7	raspberry	gary441@tigo.tk	W4UIEQFSAu	ATF1004	
8	rollsyoda	kokorinsanja@gmailvn.net	V'inWt^Tql	ATF1007	
9	tracksjuno	katyskat@gmailvn.net	ThTSMujcNW	ATF1006	
10	xylophone	hedob40703@dnitem.com	88pnH3tRmN	ATF1001	
11					

Question-2:

Perform INSERT, SELECT, DELETE Queries in user table

Solution:

Insert Query

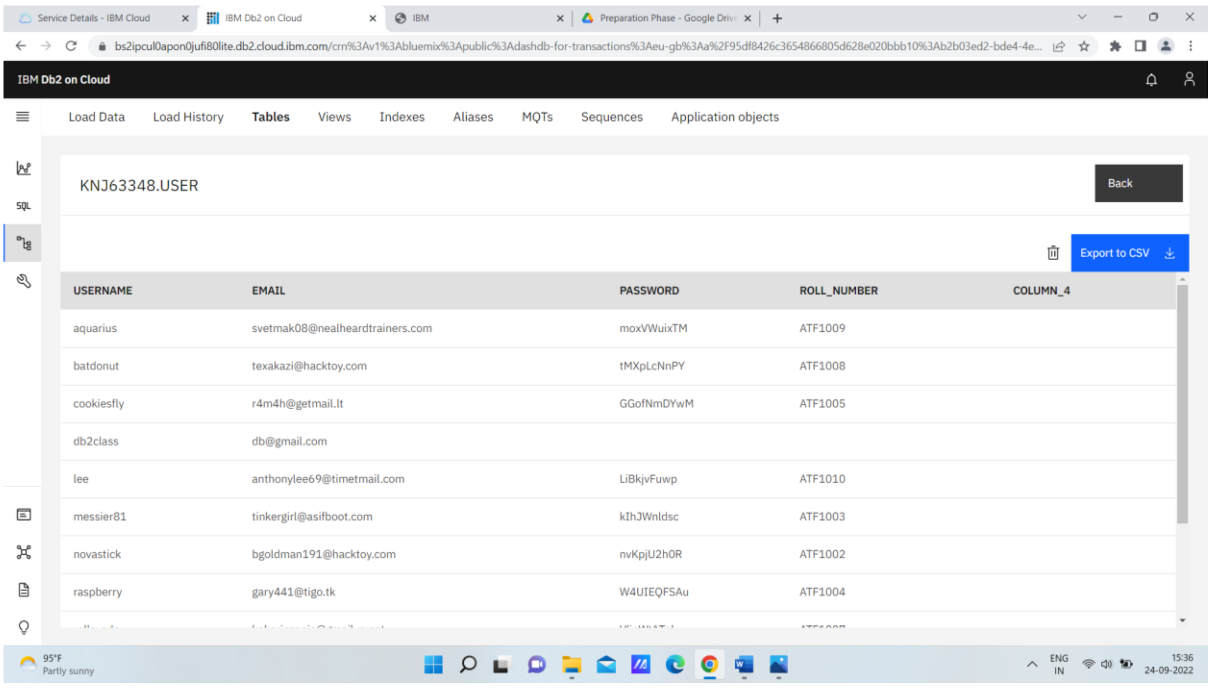


```
INSERT INTO user (Username, Email)
  VALUES ('db2class','db@gmail.com');
```

History

Script	Date	Status	Runtime
Template - Insert Statement	Sep 24, 2022 3:05:19 PM	1	0.011 s
INSERT INTO user (Username, Email) VALUES ('db2class','db@gmail.com')		✓	0.011 s

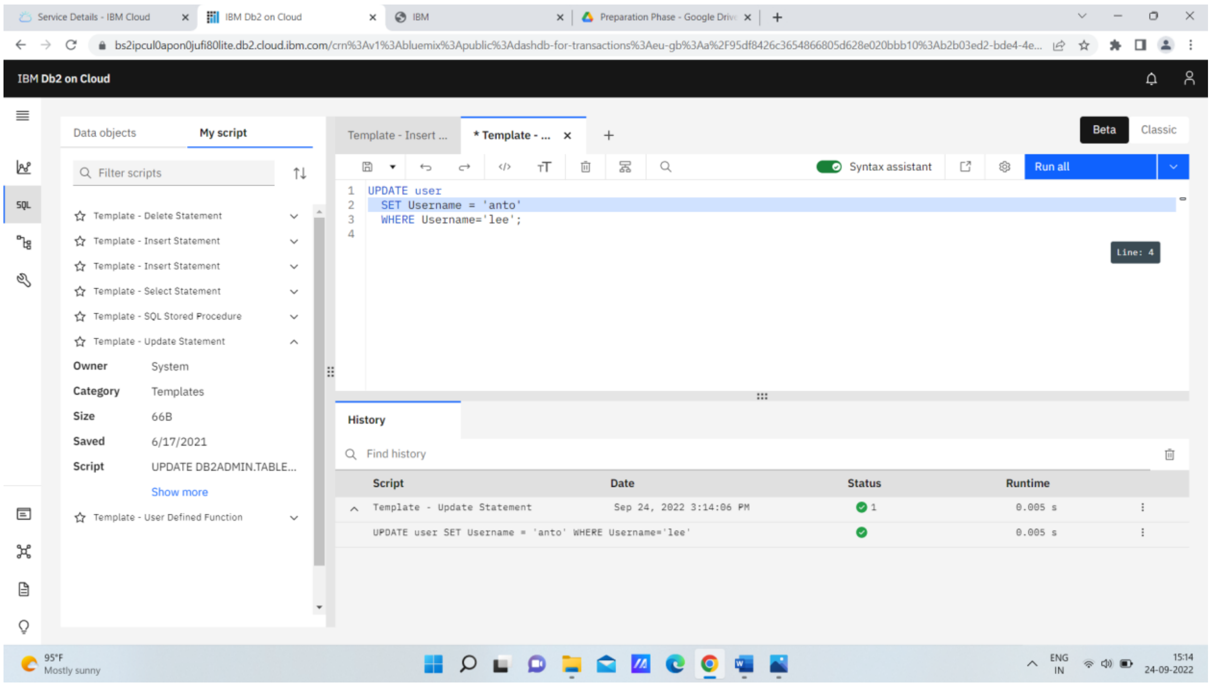
Insert Query Output



KNJ63348.USER

USERNAME	EMAIL	PASSWORD	ROLL_NUMBER	COLUMN_4
aquarius	svetmak08@nealheardtrainers.com	moxVWuixTM	ATF1009	
batdonut	texakazi@hacktoy.com	tMXpLcNnPY	ATF1008	
cookiesfly	r4m4h@getmail.lt	GGofNmDYwM	ATF1005	
db2class	db@gmail.com			
lee	anthonylee69@timetmail.com	LiBkjvFuwp	ATF1010	
messier81	tinkergirl@asifboot.com	kIhJWnldsc	ATF1003	
novastick	bgoldman191@hacktoy.com	nvKpjU2h0R	ATF1002	
raspberry	gary441@tigo.tk	W4UIEQFSAu	ATF1004	

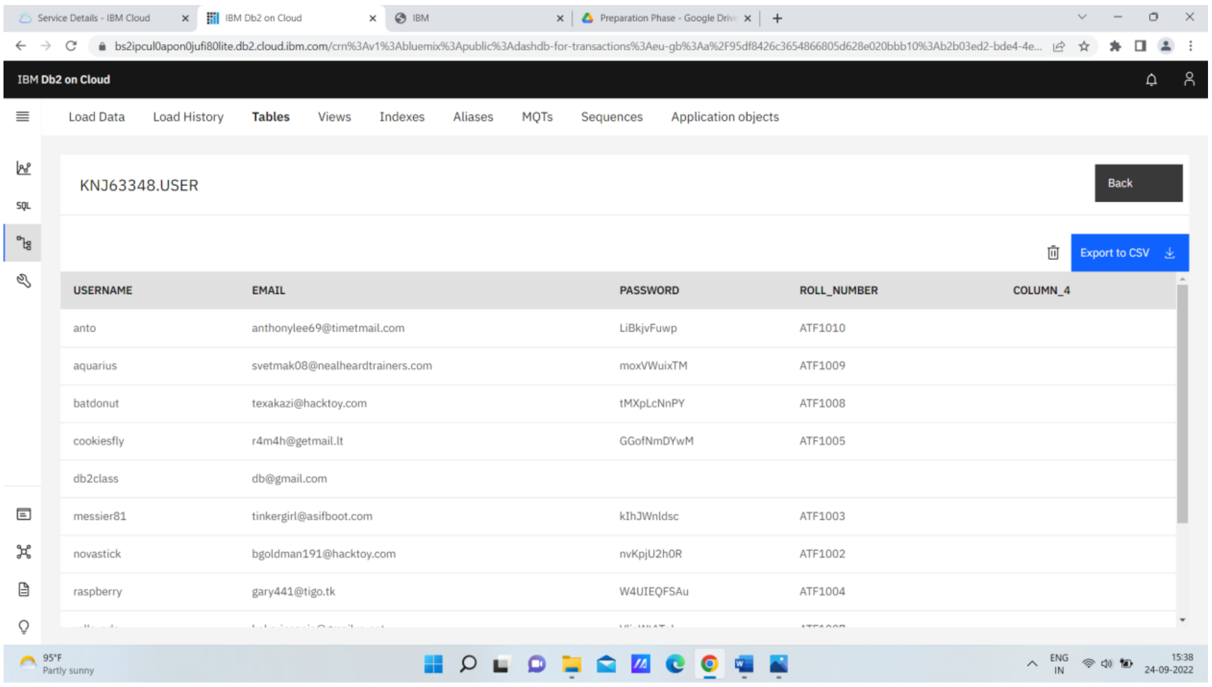
Update Query



```
UPDATE user
  SET Username = 'anto'
  WHERE Username='lee';
```

Script	Date	Status	Runtime
Template - Update Statement	Sep 24, 2022 3:14:06 PM	✓ 1	0.005 s
UPDATE user SET Username = 'anto' WHERE Username='lee'		✓	0.005 s

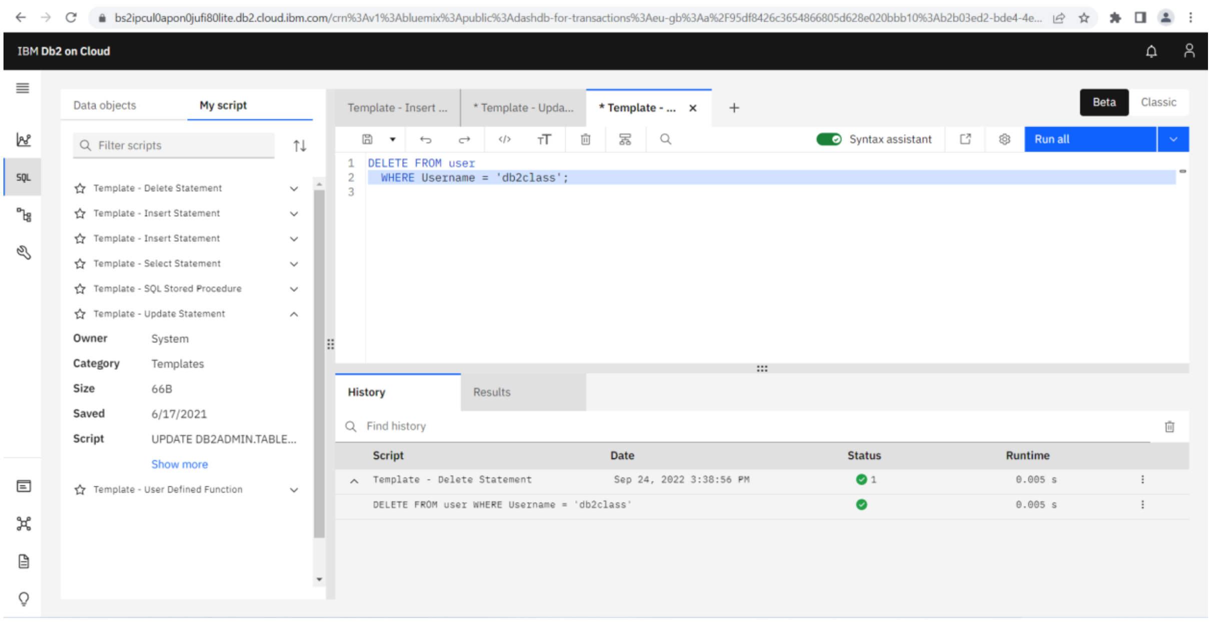
Update Query Output



KNJ63348.USER

USERNAME	EMAIL	PASSWORD	ROLL_NUMBER	COLUMN_4
anto	anthonylee69@timetmail.com	LiBkjvFuwp	ATF1010	
aquarius	svetmak08@nealheardtrainers.com	moxVWuixTM	ATF1009	
batdonut	texakazi@hacktoy.com	tMXpLcNnPY	ATF1008	
cookiesfly	r4m4h@getmail.lt	GGofNmDYwM	ATF1005	
db2class	db@gmail.com			
messier81	tinkergirl@asifboot.com	kIhJWnldsc	ATF1003	
novastick	bgoldman191@hacktoy.com	nvKpjU2h0R	ATF1002	
raspberry	gary441@tigo.tk	W4UIEQFSAu	ATF1004	

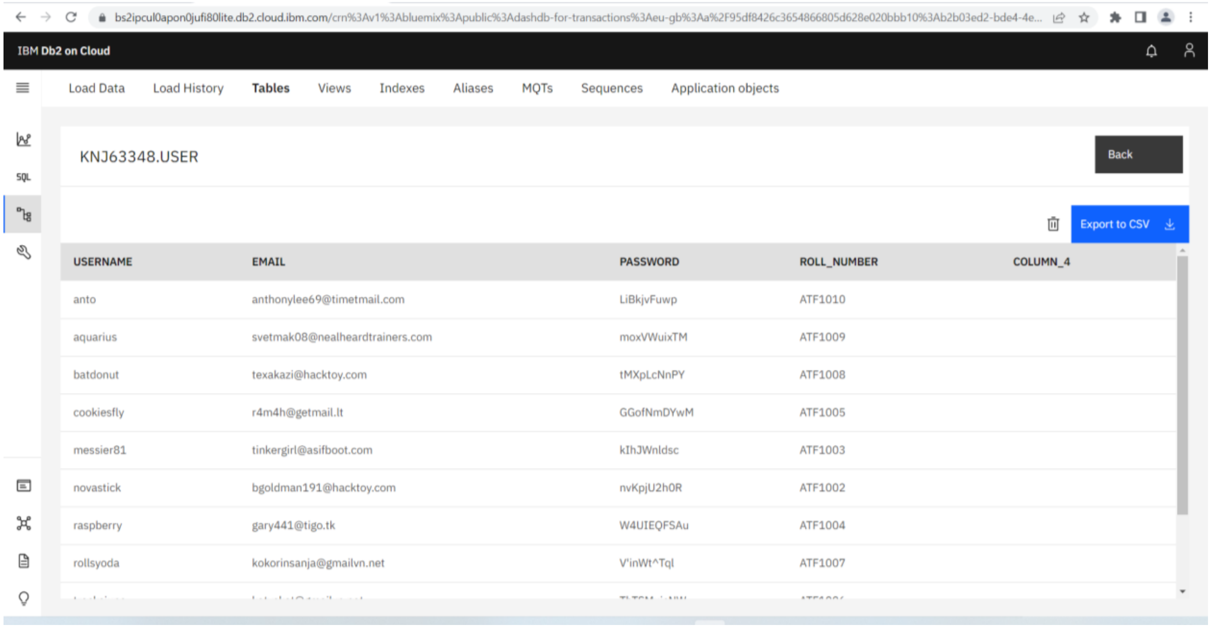
Delete Query



```
DELETE FROM user
 WHERE Username = 'db2class';
```

Script	Date	Status	Runtime
Template - Delete Statement	Sep 24, 2022 3:38:56 PM	✓ 1	0.005 s
DELETE FROM user WHERE Username = 'db2class'		✓	0.005 s

Delete Query Output



KNJ63348.USER

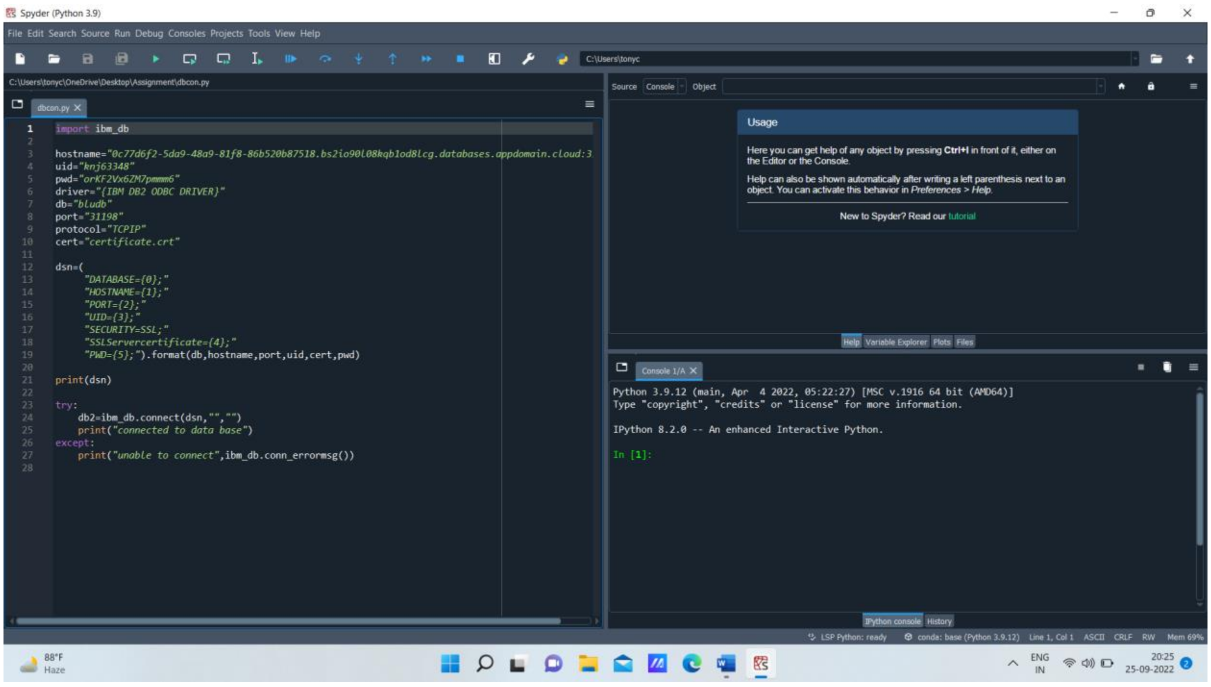
USERNAME	EMAIL	PASSWORD	ROLL_NUMBER	COLUMN_4
anto	anthonylee69@timetmail.com	LiBkjvFuwp	ATF1010	
aquarius	svetmak08@nealheardtrainers.com	moxVWuixTM	ATF1009	
batdonut	texakazi@hacktoy.com	tMXpLcNnPY	ATF1008	
cookiesfly	r4m4h@getmail.lt	GGofNmDYwM	ATF1005	
messier81	tinkergirl@asifboot.com	kIhJWnldsc	ATF1003	
novastick	bgoldman191@hacktoy.com	nvKpjU2h0R	ATF1002	
raspberry	gary441@tigo.tk	W4UIEQFSAu	ATF1004	
rollsyoda	kokorinsanja@gmailvn.net	V'inWt^Tql	ATF1007	

Question-3:

Connect Python to DB2

Solution:

Source Code for Python to DB2 Connection


Spyder (Python 3.9)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\tonyc
C:\Users\tonyc\OneDrive\Desktop\Assignment\dbcon.py
dbcon.py
Source Console Object
Help Variable Explorer Plots Files
Console 1/A
IPython console History
LSP Python: ready conda: base (Python 3.9.12) Line 1, Col 1 ASCII CRLF RW Mem 69%
88°F Haze
ENG IN 20:25 25-09-2022

```python
import ibm_db

hostname="0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l08kqb1od8lcg.databases.appdomain.cloud:3
uid="knj63348"
pwd="orKF2Vx6ZM7pmmm6"
driver="{IBM DB2 ODBC DRIVER}"
db="bludb"
port="31198"
protocol="TCPIP"
cert="certificate.crt"

dsn=(
    "DATABASE={0};"
    "HOSTNAME={1};"
    "PORT={2};"
    "UID={3};"
    "SECURITY=SSL;"
    "SSLServercertificate={4};"
    "PWD={5};").format(db,hostname,port,uid,cert,pwd)

print(dsn)

try:
    db2=ibm_db.connect(dsn,"","")
    print("connected to data base")
except:
    print("unable to connect",ibm_db.conn_errormsg())
```

Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

New to Spyder? Read our tutorial

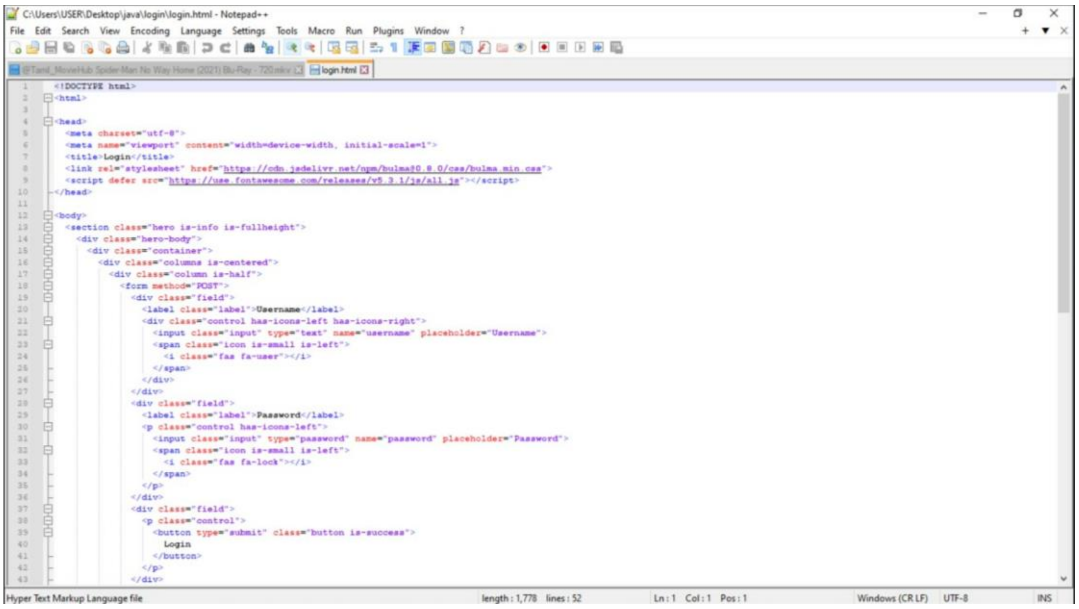
```
Python 3.9.12 (main, Apr  4 2022, 05:22:27) [MSC v.1916 64 bit (AMD64)]
Type "copyright", "credits" or "license" for more information.

IPython 8.2.0 -- An enhanced Interactive Python.

In [1]:
```

Question-4:

Create a flask app with 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

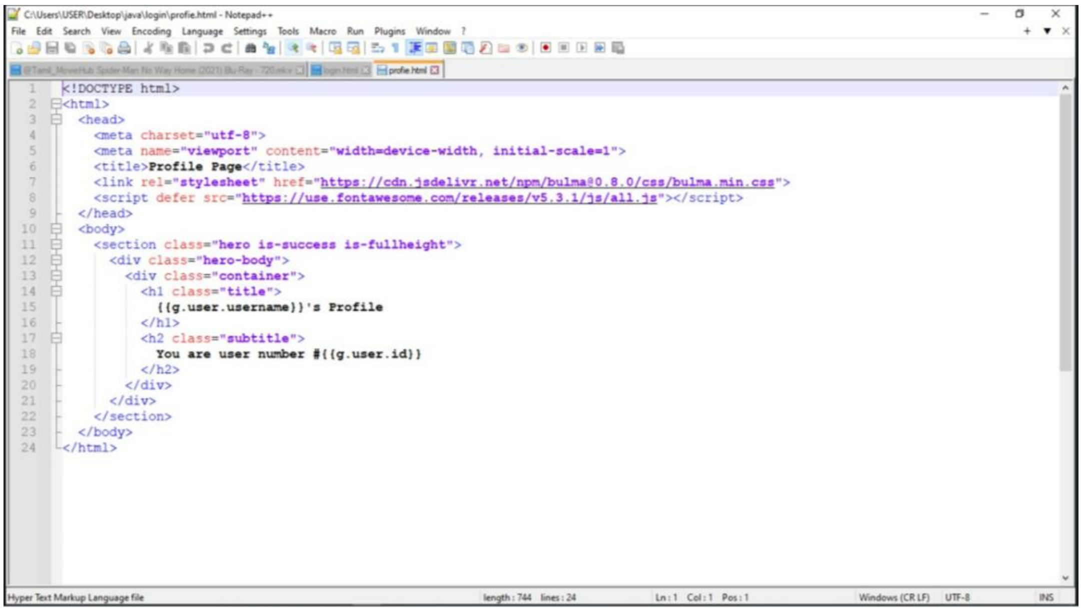
Solution:

Source code for login page



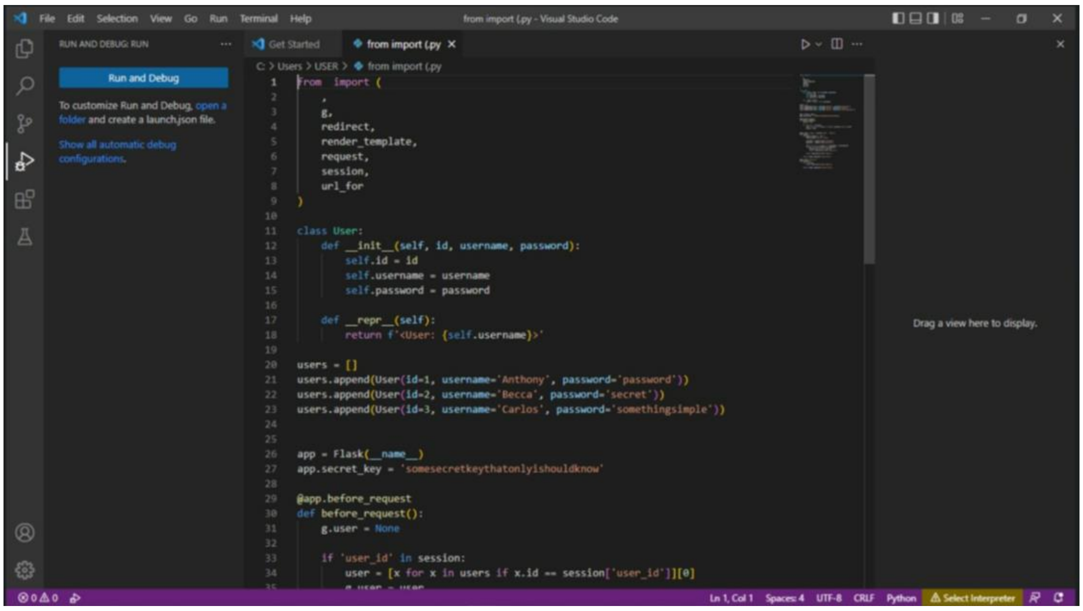
```
<!DOCTYPE html>
<html>

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <title>Login</title>
  <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bulma@0.8.0/css/bulma.min.css">
  <script defer src="https://use.fontawesome.com/releases/v5.3.1/js/all.js"></script>
</head>

<body>
  <section class="hero is-info is-fullheight">
    <div class="hero-body">
      <div class="container">
        <div class="columns is-centered">
          <div class="column is-half">
            <form method="POST">
              <div class="field">
                <label class="label">Username</label>
                <div class="control has-icons-left has-icons-right">
                  <input class="input" type="text" name="username" placeholder="Username">
                  <span class="icon is-small is-left">
                    <i class="fas fa-user"></i>
                  </span>
                </div>
              </div>
              <div class="field">
                <label class="label">Password</label>
                <p class="control has-icons-left">
                  <input class="input" type="password" name="password" placeholder="Password">
                  <span class="icon is-small is-left">
                    <i class="fas fa-lock"></i>
                  </span>
                </p>
              </div>
              <div class="field">
                <p class="control">
                  <button type="submit" class="button is-success">
                    Login
                  </button>
                </p>
              </div>
```

Source code for welcome page



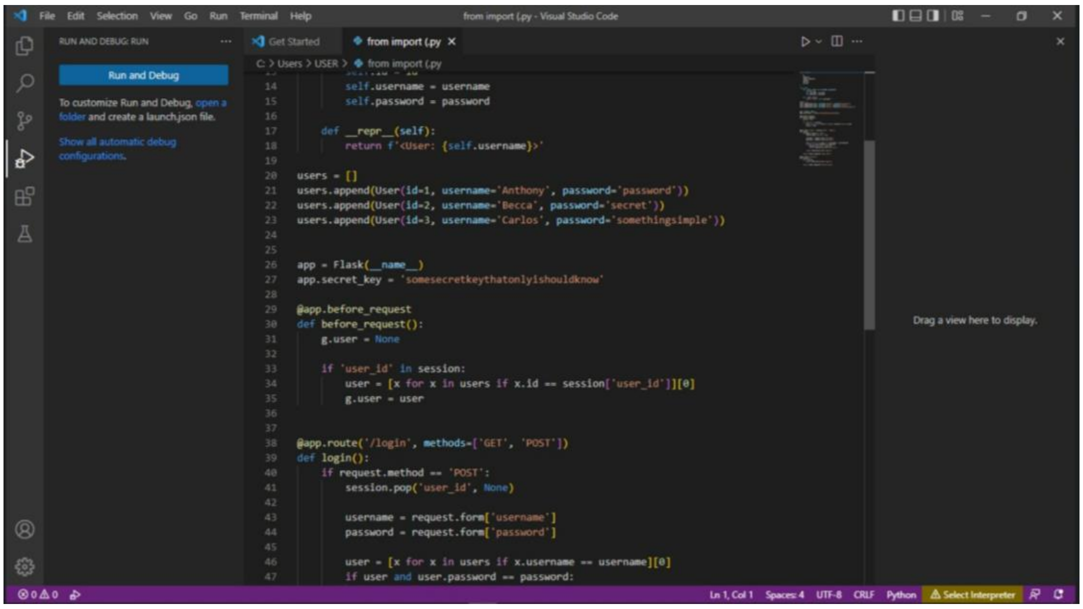
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Profile Page</title>
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bulma@0.8.0/css/bulma.min.css">
    <script defer src="https://use.fontawesome.com/releases/v5.3.1/js/all.js"></script>
  </head>
  <body>
    <section class="hero is-success is-fullheight">
      <div class="hero-body">
        <div class="container">
          <h1 class="title">
            {{g.user.username}}'s Profile
          </h1>
          <h2 class="subtitle">
            You are user number #{{g.user.id}}
          </h2>
        </div>
      </div>
    </section>
  </body>
</html>
```

Flask code



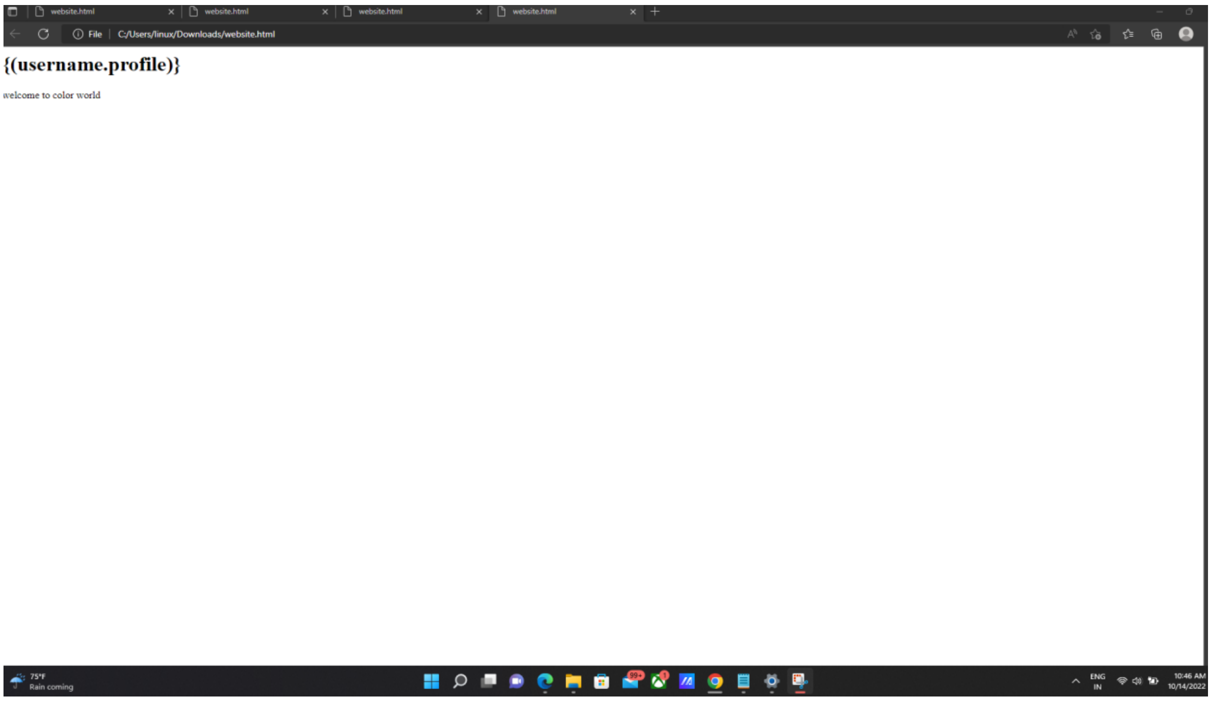
```python
from  import (
    ,
    g,
    redirect,
    render_template,
    request,
    session,
    url_for
)

class User:
    def __init__(self, id, username, password):
        self.id = id
        self.username = username
        self.password = password

    def __repr__(self):
        return f'<User: {self.username}>'

users = []
users.append(User(id=1, username='Anthony', password='password'))
users.append(User(id=2, username='Becca', password='secret'))
users.append(User(id=3, username='Carlos', password='somethingsimple'))


app = Flask(__name__)
app.secret_key = 'somesecretkeythatonlyishouldknow'

@app.before_request
def before_request():
    g.user = None

    if 'user_id' in session:
        user = [x for x in users if x.id == session['user_id']][0]
```

Flask code



```python
        self.username = username
        self.password = password

    def __repr__(self):
        return f'<User: {self.username}>'

users = []
users.append(User(id=1, username='Anthony', password='password'))
users.append(User(id=2, username='Becca', password='secret'))
users.append(User(id=3, username='Carlos', password='somethingsimple'))


app = Flask(__name__)
app.secret_key = 'somesecretkeythatonlyishouldknow'

@app.before_request
def before_request():
    g.user = None

    if 'user_id' in session:
        user = [x for x in users if x.id == session['user_id']][0]
        g.user = user


@app.route('/login', methods=['GET', 'POST'])
def login():
    if request.method == 'POST':
        session.pop('user_id', None)

        username = request.form['username']
        password = request.form['password']

        user = [x for x in users if x.username == username][0]
        if user and user.password == password:
```

Login Page Output



C:/Users/linux/Downloads/website.html

{(username.profile)}

welcome to color world

Welcome Page Output