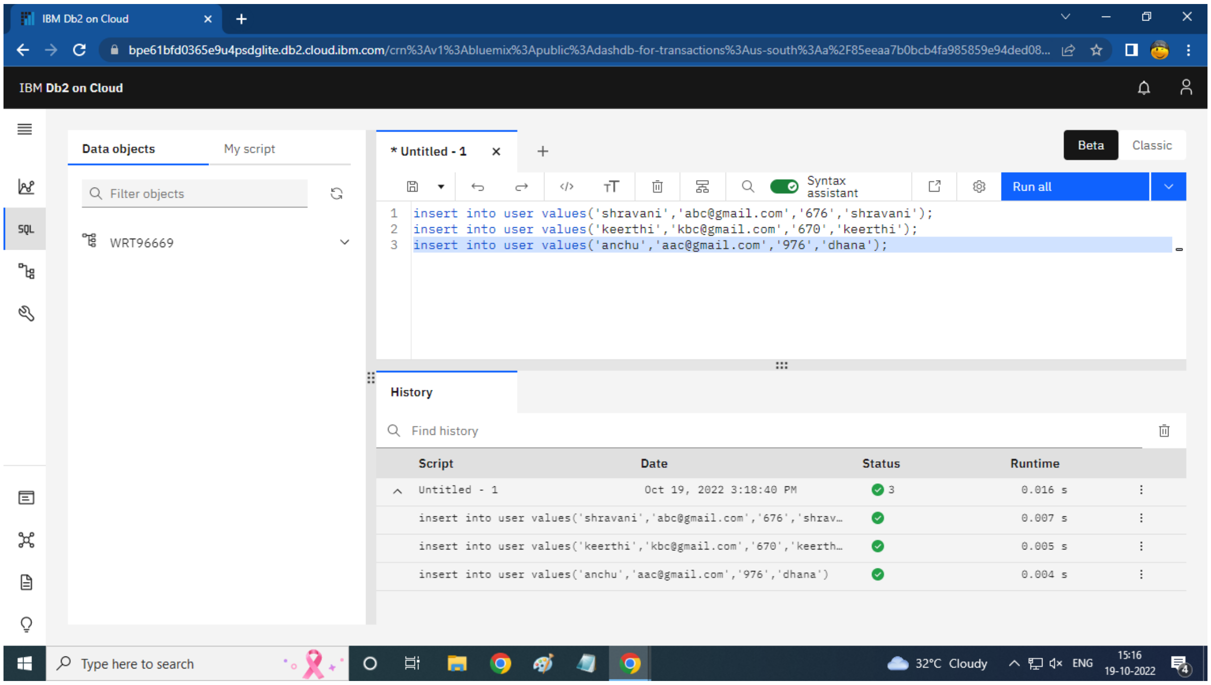
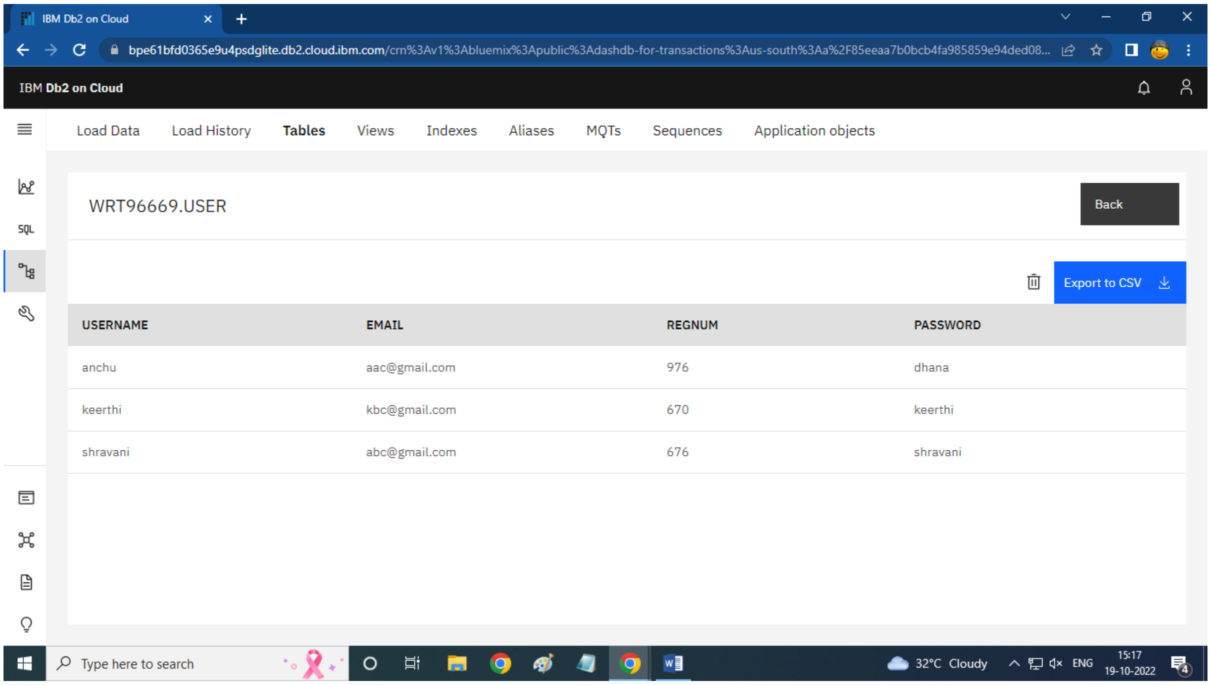
1.



```
insert into user values('shravani','abc@gmail.com','676','shravani');
insert into user values('keerthi','kbc@gmail.com','670','keerthi');
insert into user values('anchu','aac@gmail.com','976','dhana');
```

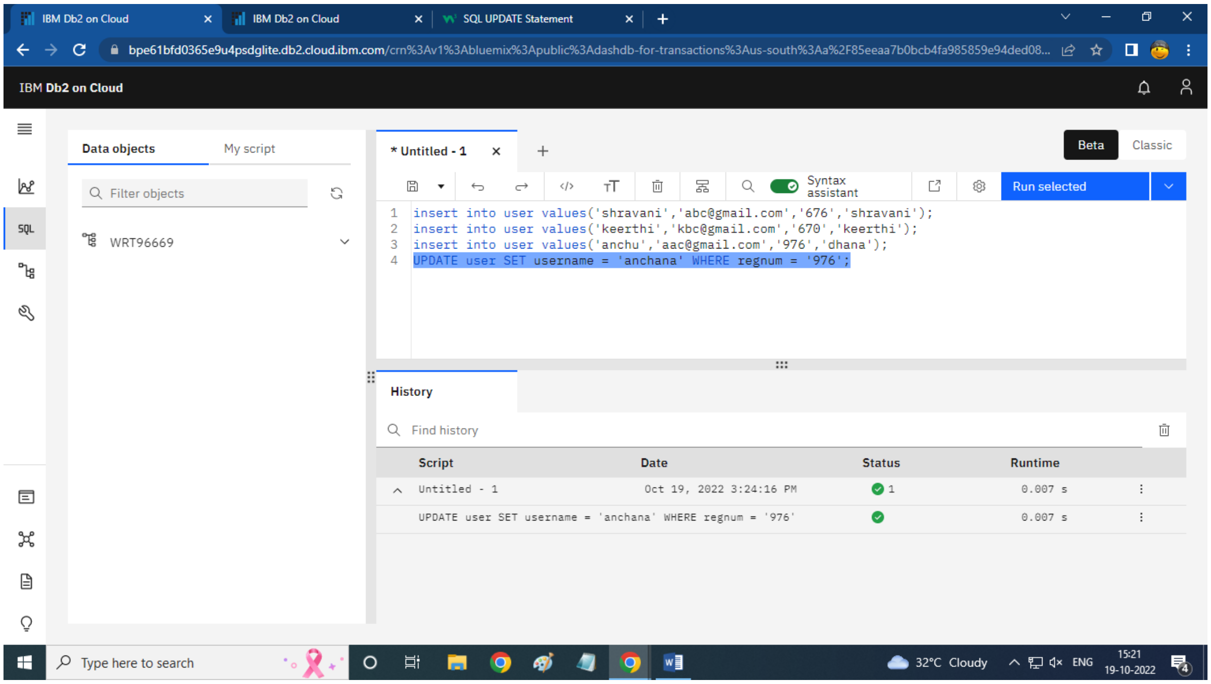
Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 3:18:40 PM	✓ 3	0.016 s
insert into user values('shravani','abc@gmail.com','676','shrav…		✓	0.007 s
insert into user values('keerthi','kbc@gmail.com','670','keerth…		✓	0.005 s
insert into user values('anchu','aac@gmail.com','976','dhana')		✓	0.004 s



WRT96669.USER

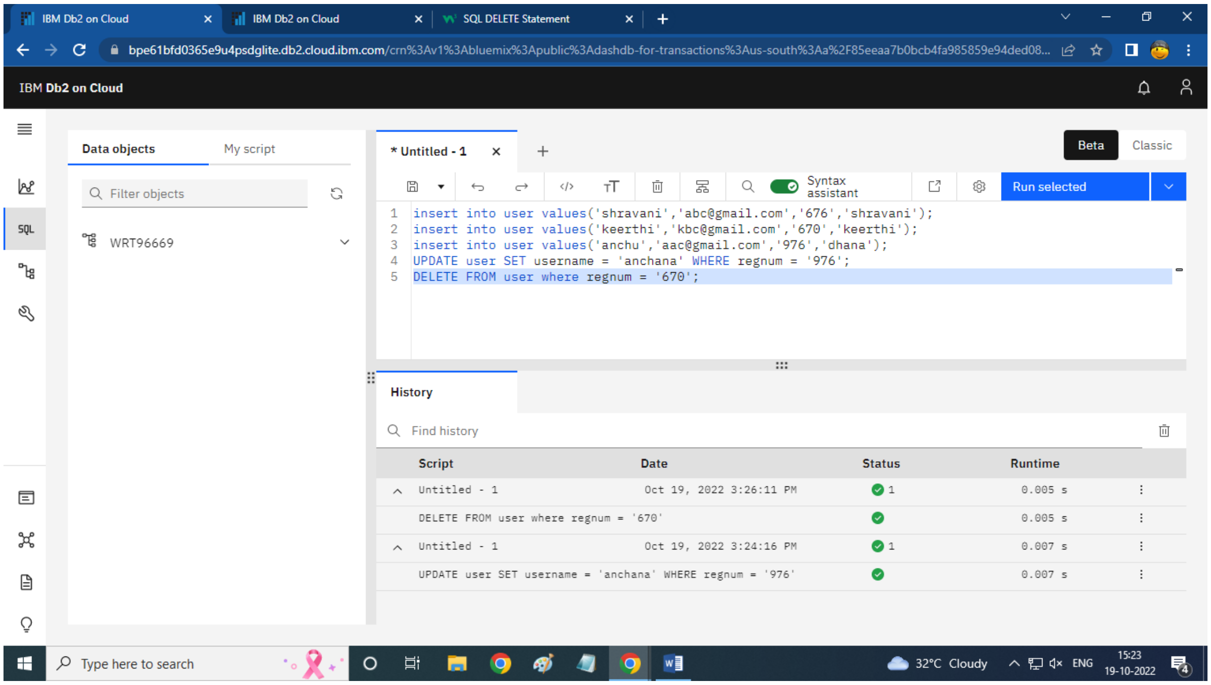
USERNAME	EMAIL	REGNUM	PASSWORD
anchu	aac@gmail.com	976	dhana
keerthi	kbc@gmail.com	670	keerthi
shravani	abc@gmail.com	676	shravani

2.



```
insert into user values('shravani','abc@gmail.com','676','shravani');
insert into user values('keerthi','kbc@gmail.com','670','keerthi');
insert into user values('anchu','aac@gmail.com','976','dhana');
UPDATE user SET username = 'anchana' WHERE regnum = '976';
```

Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 3:24:16 PM	✓ 1	0.007 s
UPDATE user SET username = 'anchana' WHERE regnum = '976'		✓	0.007 s



```
insert into user values('shravani','abc@gmail.com','676','shravani');
insert into user values('keerthi','kbc@gmail.com','670','keerthi');
insert into user values('anchu','aac@gmail.com','976','dhana');
UPDATE user SET username = 'anchana' WHERE regnum = '976';
DELETE FROM user where regnum = '670';
```

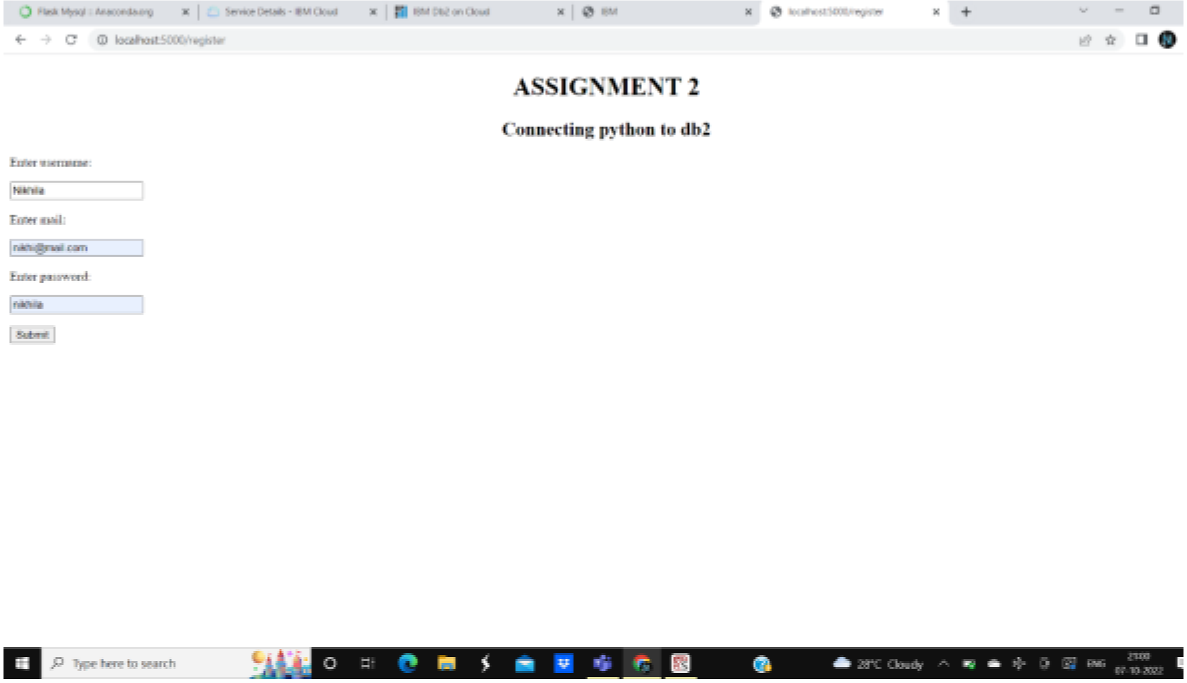
Script	Date	Status	Runtime
Untitled - 1	Oct 19, 2022 3:26:11 PM	✓ 1	0.005 s
DELETE FROM user where regnum = '670'		✓	0.005 s
Untitled - 1	Oct 19, 2022 3:24:16 PM	✓ 1	0.007 s
UPDATE user SET username = 'anchana' WHERE regnum = '976'		✓	0.007 s

3. Connect python code to db2

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=6667d8e9-9d4d-4ccb-ba32-
21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;
SECURITY=SSL;
SSLServerCertificate=DigiCertGlobalRootCA.crt;
UID=tbl66129
;PWD=LPzyHjRGcm XprQoN",",")
 @app.route('/', methods =['GET', 'POST'])
def register(): msg = ''
 if request.method == 'POST' :
username = request.form['username']
 email = request.form['email']
 password = request.form['password'] sql = "SELECT * FROM users WHERE username =?"
stmt = ibm_db.prepare(conn, sql)
 ibm_db.bind_param(stmt,1,username)
 ibm_db.execute(stmt) account = ibm_db.fetch_assoc(stmt)
print(account)
if account:
insert_sql = "INSERT INTO users VALUES (?, ?, ?)"
 prep_stmt = ibm_db.prepare(conn, insert_sql)
 ibm_db.bind_param(prep_stmt, 1, username)
ibm_db.bind_param(prep_stmt, 2, email)
 ibm_db.bind_param(prep_stmt, 3, password)
ibm_db.execute(prep_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.run(debug = True)
```

Register.html



ASSIGNMENT 2

Connecting python to db2

Enter username:

Nikhila

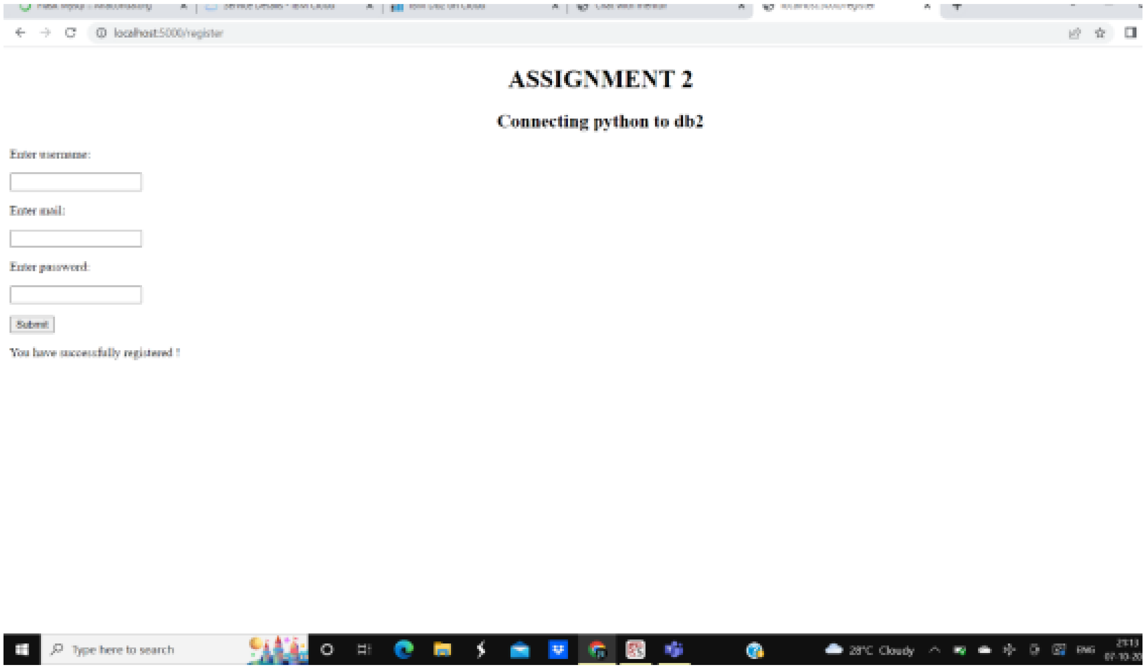
Enter mail:

nikhi@mail.com

Enter password:

nikhila

Submit



ASSIGNMENT 2

Connecting python to db2

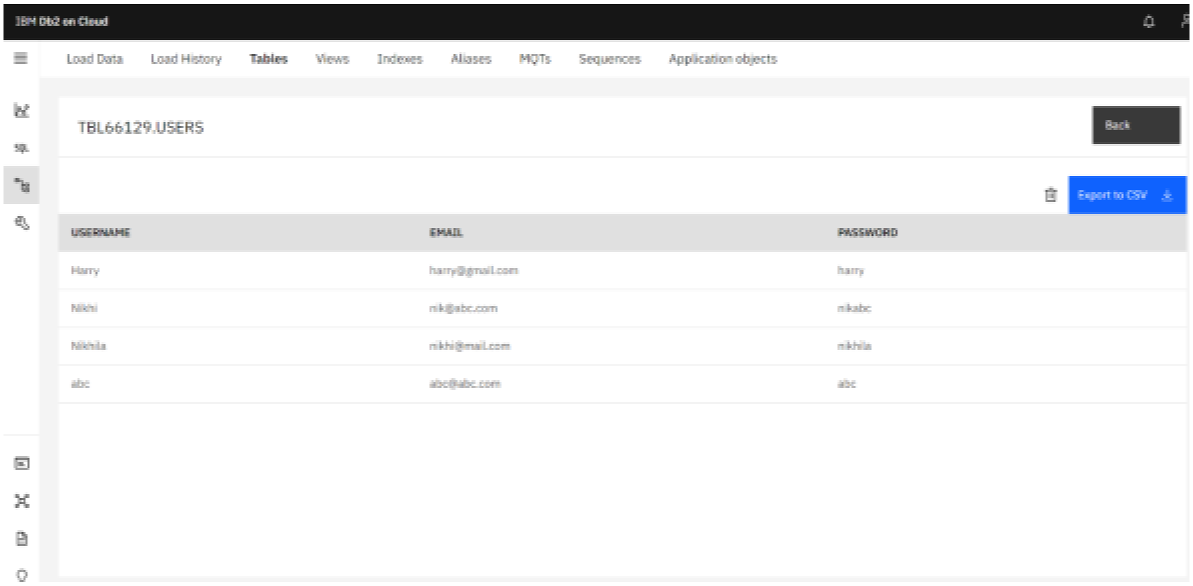
Enter username:

Enter mail:

Enter password:

Submit

You have successfully registered !



TBL66129.USERS

USERNAME	EMAIL	PASSWORD
Harry	harry@gmail.com	harry
Nikhi	nik@abc.com	nikabc
Nikhila	nikhi@mail.com	nikhila
abc	abc@abc.com	abc

4. 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 the database and navigate to the login page to authenticate user username and password. If the user is valid show the welcome page

Register.html

```
<!DOCTYPE html>

<html>

<body>

<center><h1>ASSIGNMENT 2</h1></center>

<center><h2>REGISTRATION</h2></center>

<form action="http://localhost:5000" method="POST">

<p>Enter username:</p>

<p><input type="text" name="username"/></p>

<p>Enter mail:</p>

<p><input type="email" name="email"/></p>

<p>Enter password:</p>
```

```
<p><input type="password" name="password"/></p>
<p><input type="submit" value="Submit"/></p>
{{msg}}
</form>
</body>
</html>
```

Login.html

Login

```
<!DOCTYPE html>
<html>
<body>
<center><h1>ASSIGNMENT 2</h1></center>
<center><h2>LOGIN</h2></center>
<form action="http://localhost:5000/login" method="POST">
<p>Enter username:</p>
<p><input type="text" name="username"/></p>
<p>Enter password:</p>
<p><input type="password" name="password"/></p>
<p><input type="submit" value="Submit"/></p>
{{msg}}
</form>
</body>
</html>
```

WELCOME.HTML

```
<!DOCTYPE html>
<html>
<body>
<center><h1>ASSIGNMENT 2</h1></center>
<center><h2>LOGIN</h2></center>
<form action="http://localhost:5000/login" method="POST">
```

```
<p>Enter username:</p>

<p><input type="text" name="username"/></p>

<p>Enter password:</p>

<p><input type="password" name="password"/></p>

<p><input type="submit" value="Submit"/></p>

{{msg}}

</form>

</body>

</html>
```

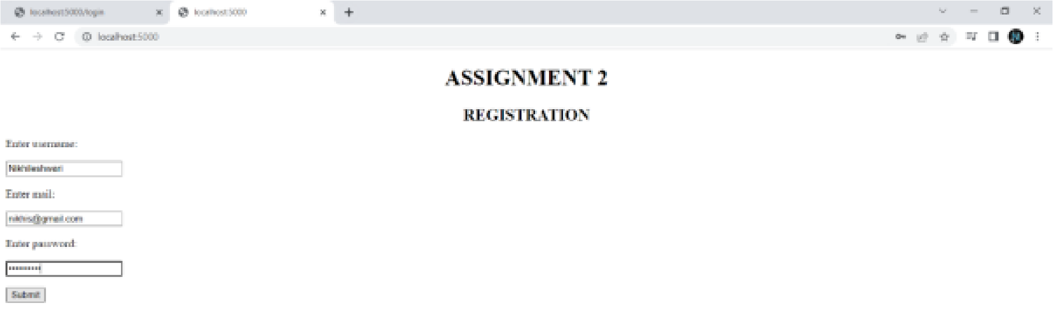
app.py

```
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=6667d8e9-9d4d-4ccb-ba32-

21da3bb5aafc.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376

;SECURITY=SSL;

SSLServerCertificate=DigiCertGlobalRootCA.crt;

UID=tbl66129;PWD=LPzyHjRGcm XprQoN",'','')

@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: print("hi")
 session['logged in'] = True
session['id'] = account['USERNAME']
userid= account['USERNAME']
 session['username'] = account['USERNAME']
msg = 'Logged in successfully !'
return render_template('login.html', msg = msg)

else:
 msg = 'Incorrect username / password !'
 return render_template('login.html', msg = msg)
 @app.route('/', methods =['GET', 'POST'])
def register(): msg = '' if request.method == 'POST'
: username = request.form['username']
 email = request.form['email']
password = request.form['password']
 sql = "SELECT * FROM users WHERE username =?" stmt = ibm_db.prepare(conn, sql)
ibm_db.bind_param(stmt,1,username)
 ibm_db.execute(stmt) account = ibm_db.fetch_assoc(stmt)
```

```
print(account) if account: msg = 'Account already exists !'
 elif not re.match(r'[^@]+@[^@]+\.[^@]+', email):
 msg = 'Invalid email address !
' elif not re.match(r'[A-Za-z0-9]+', username):
 msg = 'name must contain only characters and numbers !'
else: insert_sql = "INSERT INTO users VALUES (?, ?, ?)" prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prep_stmt, 1, username)
 ibm_db.bind_param(prep_stmt, 2, email)
ibm_db.bind_param(prep_stmt, 3, password)
ibm_db.execute(prep_stmt)
```

```
if __name__ == '__main__':
   app.run(debug = True)
```



ASSIGNMENT 2

REGISTRATION

Enter username:

Nikhileshwari

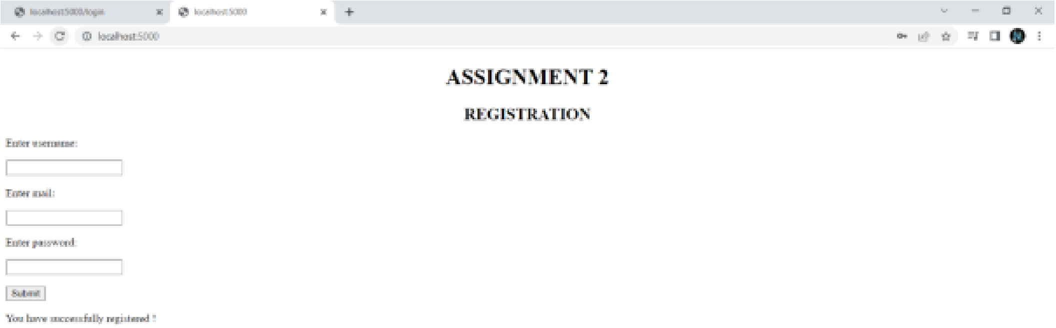
Enter mail:

nikhis@gmail.com

Enter password:

•••••••••

Submit



ASSIGNMENT 2

REGISTRATION

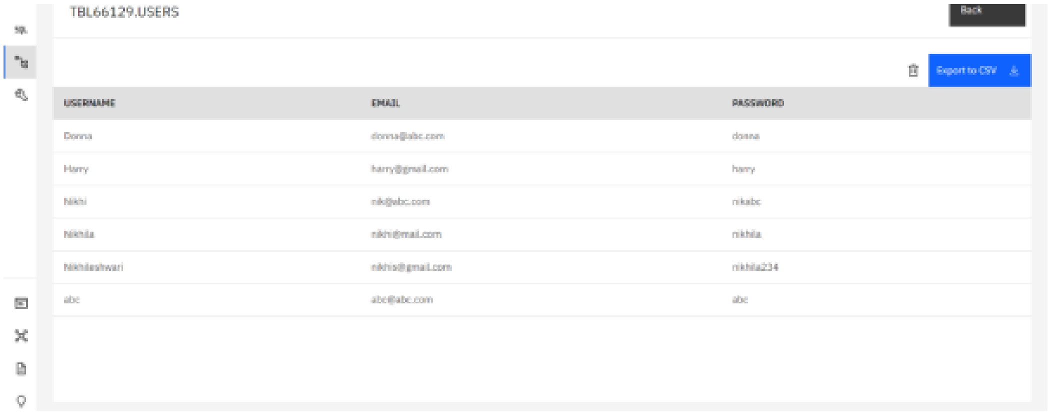
Enter username:

Enter mail:

Enter password:

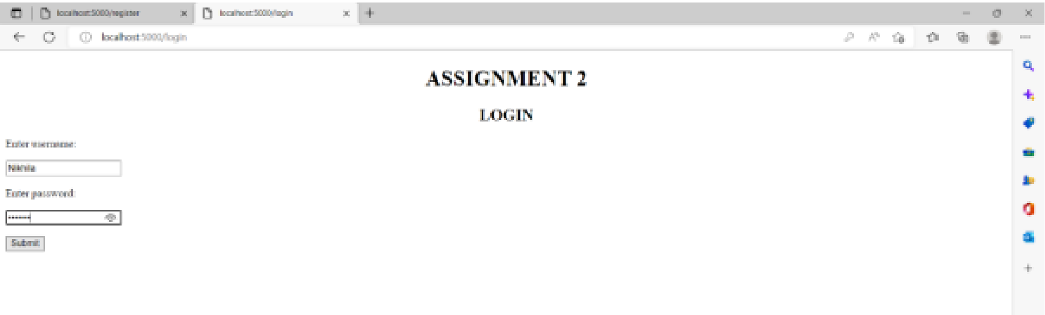
Submit

You have successfully registered !



TBL66129.USERS

USERNAME	EMAIL	PASSWORD
Donna	donna@abc.com	donna
Harry	harry@gmail.com	harry
Nikhi	nik@abc.com	nikabc
Nikhila	nikhi@mail.com	nikhila
Nikhileshwari	nikhis@gmail.com	nikhila234
abc	abc@abc.com	abc



ASSIGNMENT 2

LOGIN

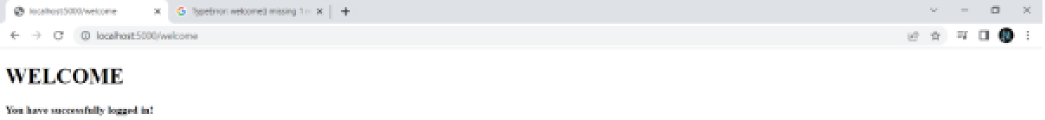
Enter username:

Nikhila

Enter password:

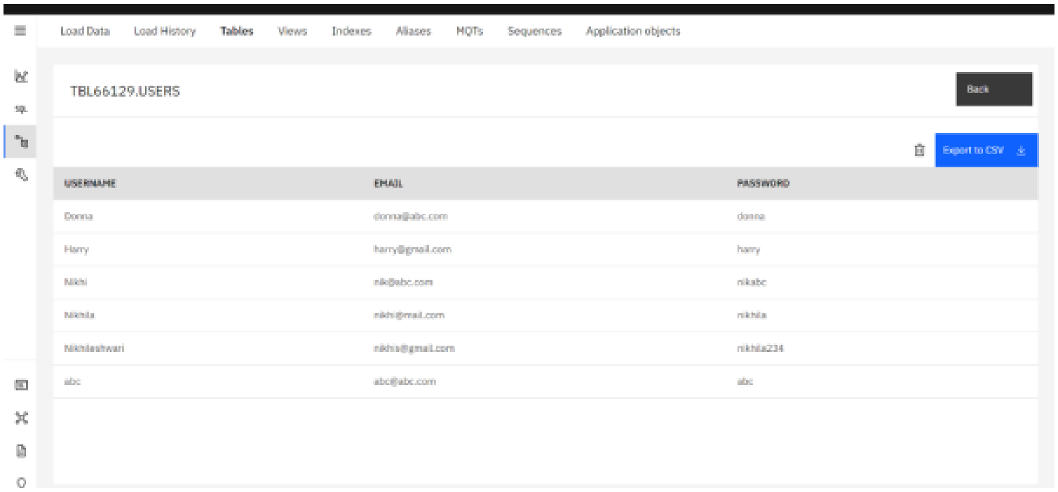
••••••

Submit



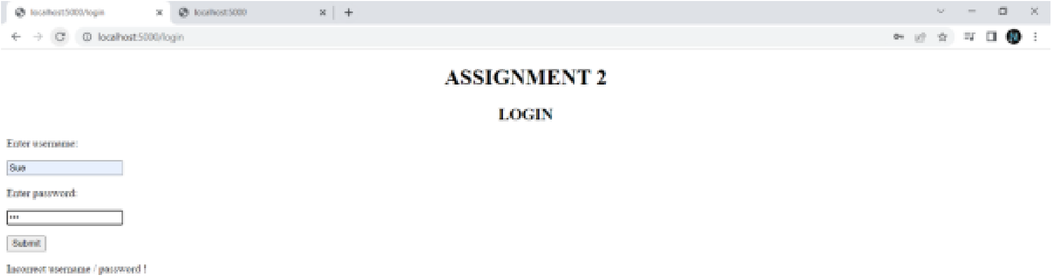
WELCOME

You have successfully logged in!



TBL66129.USERS

USERNAME	EMAIL	PASSWORD
Donna	donna@abc.com	donna
Harry	harry@gmail.com	harry
Nikhi	nik@abc.com	nikabc
Nikhila	nikhi@mail.com	nikhila
Nikhileshwari	nikhis@gmail.com	nikhila234
abc	abc@abc.com	abc



ASSIGNMENT 2

LOGIN

Enter username:

Sue

Enter password:

•••

Incorrect username / password !