

ASSIGNMENT 2

NAME: FARHANA SABREEN M

REGISTER NUMBER: 311519104016

PROJECT TOPIC: CUSTOMER CARE REGISTRY

COLLEGE NAME: MEENAKSHI SUNDARARAJAN ENGINEERING COLLEGE

TEAM NUMBER: PNT2022TMID27825

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

The screenshot shows the IBM Db2 on Cloud console. The 'Tables' tab is selected, displaying a list of tables. The 'USER' table is highlighted. The 'Table definition' panel on the right shows the table's structure:

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

The screenshot shows the IBM Db2 on Cloud console with the SQL editor open. The editor contains the following SQL statements:

```
1 insert into USER values('carolina@gmail.com','Carolina',101,'carol@123');
2 insert into USER values('fathima@gmail.com','Fathima',102,'fathima@123');
3 insert into USER values('prachi@gmail.com','Prachi',103,'prachi@123');
4 insert into USER values('sayonika@gmail.com','Sayonika',104,'sayonika@123');
```

The 'History' tab is selected, showing the execution results of the SQL statements:

Script	Date	Status	Runtime
Untitled - 1	Nov 3, 2022 8:45:49 PM	✓ 4	0.009 s
insert into USER values('carolina@gmail.com','Carolina',101,'carol@123')		✓	0.002 s
insert into USER values('fathima@gmail.com','Fathima',102,'fathima@123')		✓	0.002 s
insert into USER values('prachi@gmail.com','Prachi',103,'prachi@123')		✓	0.003 s
insert into USER values('sayonika@gmail.com','Sayonika',104,'sayonika@123')		✓	0.002 s

IBM Db2 on Cloud				
Load Data Load History Tables Views Indexes Aliases MQTs Sequences Application objects				
MMZ64024.USER				Back
				Export to CSV
EMAIL	USERNAME	ROLLNUMBER	PASSWORD	
carolina@gmail.com	Carolina	101	carol@123	
fathima@gmail.com	Fathima	102	fathima@123	
prachi@gmail.com	Prachi	103	prachi@123	
sayonika@gmail.com	Sayonika	104	sayonika@123	

2. Perform UPDATE, DELETE queries with user table

UPDATE:

QUERY:

update USER set PASSWORD='sayonika123' where ROLLNUMBER=104;

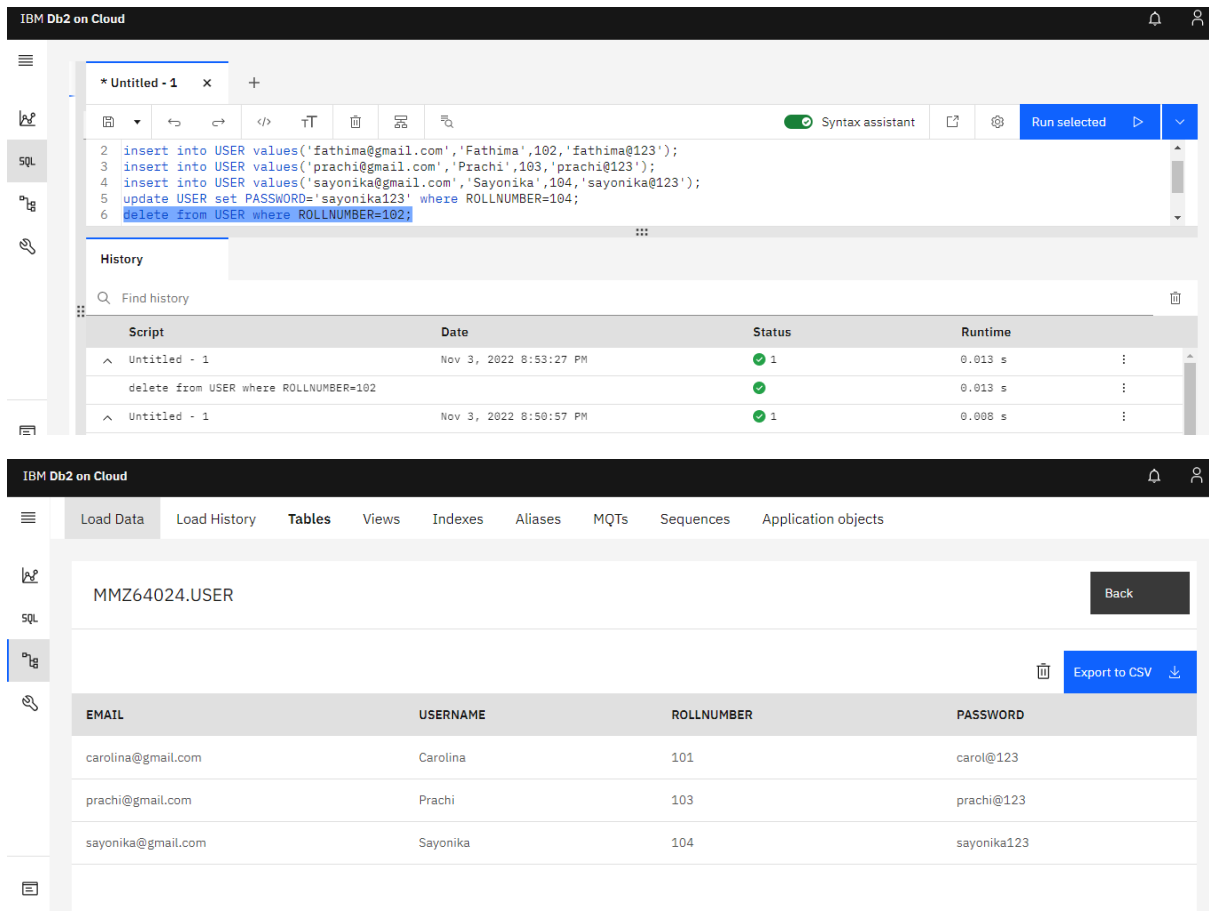
IBM Db2 on Cloud				
*Untitled - 1 x +				
<pre> 2 insert into USER values('fathima@gmail.com','Fathima',102,'fathima@123'); 3 insert into USER values('prachi@gmail.com','Prachi',103,'prachi@123'); 4 insert into USER values('sayonika@gmail.com','Sayonika',104,'sayonika@123'); 5 update USER set PASSWORD='sayonika123' where ROLLNUMBER=104; 6 </pre>				
History				
Find history				
Script	Date	Status	Runtime	
Untitled - 1	Nov 3, 2022 8:50:57 PM	1	0.008 s	:
update USER set PASSWORD='sayonika123' where ROLLNUMBER=104			0.008 s	:
Untitled - 1	Nov 3, 2022 8:48:48 PM	4	0.027 s	:

IBM Db2 on Cloud				
Load Data Load History Tables Views Indexes Aliases MQTs Sequences Application objects				
MMZ64024.USER				Back
				Export to CSV
EMAIL	USERNAME	ROLLNUMBER	PASSWORD	
carolina@gmail.com	Carolina	101	carol@123	
fathima@gmail.com	Fathima	102	fathima@123	
prachi@gmail.com	Prachi	103	prachi@123	
sayonika@gmail.com	Sayonika	104	sayonika123	

DELETE:

QUERY:

delete from USER where ROLLNUMBER=102;



3. Connect python code to DB2

PYTHON FLASK CODE:

```
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=54a2f15b-5c0f-46df-8954-7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=mmz64024;PWD=FY3HANoJIBWv5mpO","")
```

```
@app.route('/')
```

```
def home():
```


```
    return render_template('register.html')
```

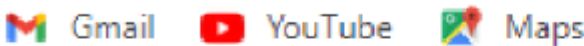
```
@app.route('/register',methods=['GET','POST'])
```

```
def register():
```

```
    global userid
```

```
msg="
if request.method=='POST':
    username=request.form['NAME']
    email=request.form['EMAIL']
    mobno=request.form['MOBILENUMBER']
    insert_sql="INSERT INTO DETAILS VALUES(?,?,?)"
    prep_stmt=ibm_db.prepare(conn,insert_sql)
    ibm_db.bind_param(prepare_stmt,1,username)
    ibm_db.bind_param(prepare_stmt,2,email)
    ibm_db.bind_param(prepare_stmt,3,mobno)
    ibm_db.execute(prepare_stmt)
    return render_template('result.html')
else:
    msg='Registration Unsuccessful! Please try again later!'
    return render_template('register.html')
if __name__=='__main__':
    app.run(debug=True)
```

localhost:5000/home



NAME

EMAIL

PHONE NUMBER



IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

MMZ64024.REGISTER

Back

Export to CSV

NAME	EMAIL	Phone Number
Sabreen	sabreen@gmail.com	9002233441
Sayonika	sayonika@gmail.com	9992133941
farhana	farhana@gmail.com	1234567890

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

FLASK CODE:

```
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.connnext("DATABASE=bludb;HOSTNAME=fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32731;Security=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=rng73403;PWD=PXxhpaVaRcxT6Qu0;")
@app.route('/')
def home():
    return render_template(register.html)
@app.route('/register',methods=['GET','POST'])
def register():
```

```

global userid
msg=" "
if request.method=='POST':
    name=request.form['name']
    email=request.form['email']
    mobile=request.form['mobilenumber']
    insert_sql="INSERT INTO REGISTER values(?,?,?)"
    prep_stmt=ibm_db.prepare(conn,insert_sql)
    ibm_db.bind_param(prepare_stmt,1,name)
    ibm_db.bind_param(prepare_stmt,2,mobile)
    ibm_db.bind_param(prepare_stmt,3,email)
    ibm_db.execute(prepare_stmt)
    return render_template('login.html',msg=msg)
else:
    msg='REGISTRATION UNSUCCESSFUL'
    return render_template('register.html',msg=msg)
@app.route('/login',methods=['GET','POST'])
def login():
    msg=" "
    if request.method=='POST':
        username=request.form['username']
        pswd=request.form['pswd']
        sql="SELECT * FROM REGISTER WHERE USERNAME=? AND
PASSWORD=?"
        stmt=ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,pswd)
        ibm_db.execute(stmt)
        acc=ibm_db.fetch_assoc(stmt)
        if acc:
            session['loggedin']=True
            msg="LOGIN SUCCESSFULLY! WELCOME"
            return render_template('welcome.html')
        else:
            msg="PLEASE ENTER VALID USERNAME/PASSWORD"
            return redirect(url_for('login',msg=msg))
if __name__=='__main__':
    app.run(debug=True)

```

register.html

```

<!DOCTYPE html>
<html>
<head>

```

```
<meta http-equiv="Content-Type" content="text/html; charset="UTF-8">
<meta name="viewport" content="width=device-width,initial-scale=1">
<meta http-equiv="X-UA-Compatible" content="ie=edge">
<title>SIGN UP</title>
<style>
body,html{
height: 100%;
margin:0;
}
*{
box-sizing:border-box;
}
.bck{
background-image:url("https://images.indianexpress.com/2020/07/Rail-
tunnel.jpg");
height: 100%;
background-position: center;
background-repeat: no-repeat;
background-size: cover;
filter: blur(12px);
-webkit-filter: blur(12px);
}
.bckbox {
background-color: rgb(0,0,0);
background-color: rgba(0,0,0, 0.09);
color: white;
font-weight: bolder;
border: 1px solid #f1f1f1;
position: absolute;
top: 50%;
left: 50%;
transform: translate(-50%, -50%);
z-index: 2;
width: 60%;
height:80%;
padding: 35px;
text-align: left;
font-size:25px;
}
.bcktext{
position: absolute;
top:20%;
left:20%;
```

[illegible]

[illegible]

```

return false;
}
if( document.f1.mob.value == "" || isNaN( document.f1.mob.value ) ) {
alert( "Please provide mobile number." );
document.f1.mob.focus() ;
return false;
}
if( document.f1.user.value == "-1" ) {
alert( "Please provide your username" );
document.f1.user.focus()
return false;
}
if( document.f1.pwd.value == "" ) {
alert( "Please provide your password" );
document.f1.pwd.focus()
return false;
}
if( document.f1.cpwd.value == "" ) {
alert( "Please provide your confirmation password" );
document.f1.cpwd.focus()
return false;
}
if( document.f1.cpwd.value != document.f1.pwd.value ) {
alert( "Password mismatch!" );
document.f1.cpwd.focus()
return false;
}
}
}
}
</script>
</form>
</body>
</html>

```

Login.html

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<title> Login Page </title>
<style>
Body {
font-family: Calibri, Helvetica, sans-serif;

```

```
    background-color: pink;
}
button {
    background-color: #4CAF50;
    width: 100%;
    color: orange;
    padding: 15px;
    margin: 10px 0px;
    border: none;
    cursor: pointer;
}
form {
    border: 3px;
}
input[type=text], input[type=password] {
    width: 100%;
    margin: 8px 0;
    padding: 12px 20px;
    display: inline-block;
    border: 2px solid green;
    box-sizing: border-box;
}
button:hover {
    opacity: 0.7;
}
.cancelbtn {
    width: auto;
    padding: 10px 18px;
    margin: 10px 5px;
}

.container {
    padding: 25px;
}


</style>
</head>
<body>
    <center> <h1> Student Login Form </h1> </center>
    <form>
        <div class="container">
            <label>Username : </label>
```




```

        <input type="text" placeholder="Enter Username" name="username"
required>
        <label>Password : </label>
        <input type="password" placeholder="Enter Password"
name="password" required>
        <button type="submit">Login</button>

    </div>
</form>
</body>
</html>

```


localhost:5000/home

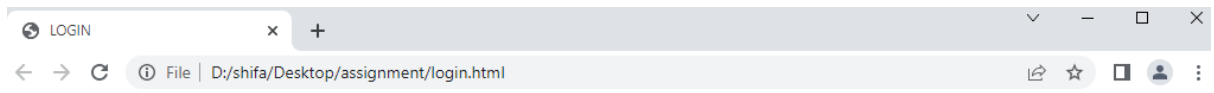
 Gmail
  YouTube
  Maps

NAME

EMAIL

PHONE NUMBER

IBM Db2 on Cloud			
<div> <div>Load Data</div> <div>Load History</div> <div>Tables</div> <div>Views</div> <div>Indexes</div> <div>Aliases</div> <div>MQTs</div> <div>Sequences</div> <div>Application objects</div> </div>			
MMZ64024.REGISTER			Back
			Export to CSV
NAME	EMAIL	Phone Number	
Sabreen	sabreen@gmail.com	9002233441	
Sayonika	sayonika@gmail.com	9992133941	
farhana	farhana@gmail.com	1234567890	



LOGIN

USERNAME :

PASSWORD :