

## Assignment -2

### IBM CLOUD DB2

|                     |                 |
|---------------------|-----------------|
| Assignment Date     | 26 OCTOBER 2022 |
| Student Name        | PRAGADISH SA    |
| Student Roll Number | 713319CS106     |
| Maximum Marks       | 2 Marks         |

#### Questions:

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

IBM Db2 on Cloud

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

Find schemas or tables: Refresh

**Tables** New table +

| Name  | Schema   | Properties |
|-------|----------|------------|
| JOB   | TBC72932 | ...        |
| USERD | TBC72932 | ...        |

Total: 2, selected: 0

**Table definition**

USERD No statistics available.

| Name     | Data type | Nullable | Length | Scale |
|----------|-----------|----------|--------|-------|
| USERNAME | CHAR      | Y        | 244    | 0     |
| EMAIL    | CHAR      | Y        | 244    | 0     |
| PASSWORD | CHAR      | Y        | 244    | 0     |
| NUMBER   | CHAR      | Y        | 244    | 0     |

View data

IBM Db2 on Cloud

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

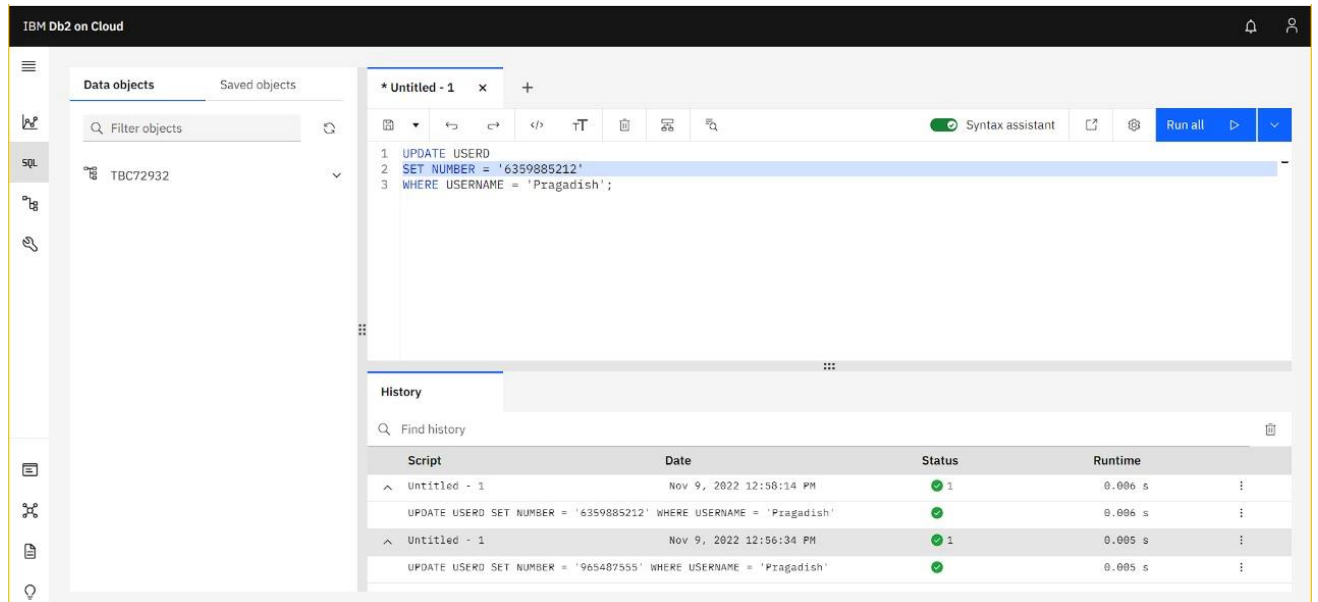
TBC72932.USERD Back

Export to CSV

| USERNAME         | EMAIL                    | PASSWORD       | NUMBER     |
|------------------|--------------------------|----------------|------------|
| Mridhul          | mridhulvinu123@gmail.com | D7M5%lw4n6Oi   | 6380951390 |
| Pragadish        | pragadish@gmail.com      | VfrA&83Hn/U6   | 965487555  |
| Priyadharshika R | Priyadharshika@gmail.com | \$yi\$qVVN1z07 | 9566482047 |

## 2. Perform UPDATE, DELETE Queries with user table.

### UPDATE TABLE:

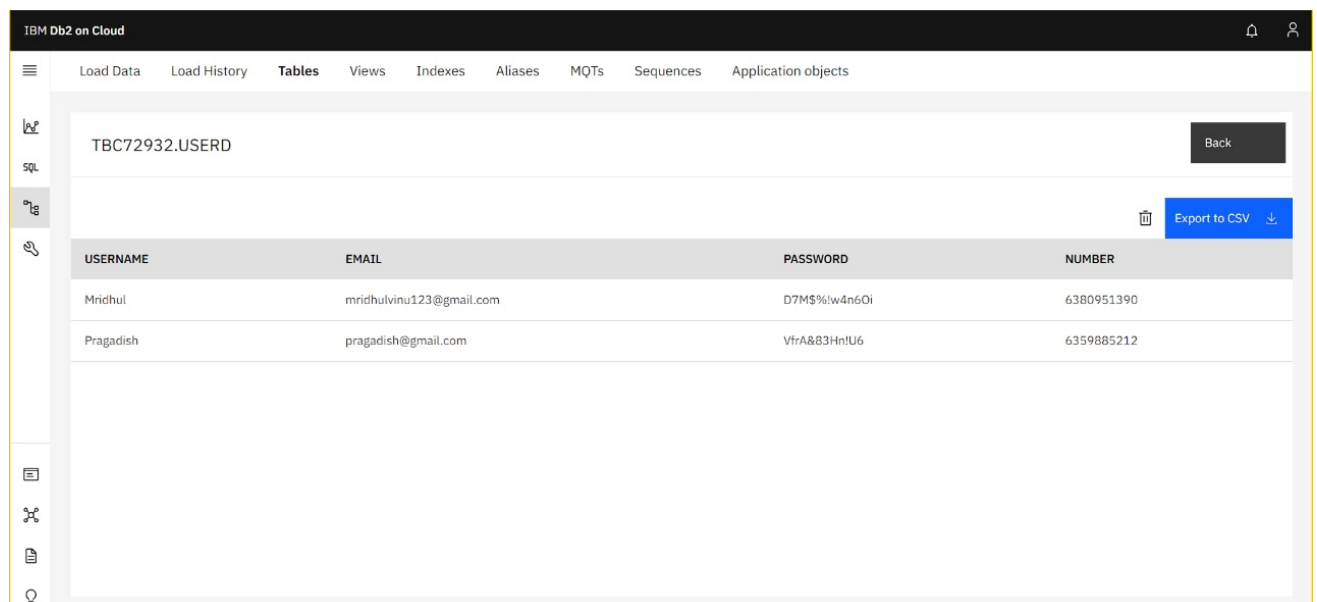


The screenshot shows the IBM Db2 on Cloud SQL editor interface. On the left, the 'Data objects' pane shows a table named 'TBC72932'. The main editor area displays an SQL script with the following lines:

```
1 UPDATE USERD
2 SET NUMBER = '6359885212'
3 WHERE USERNAME = 'Pragadish';
```

Below the script, the 'History' pane shows the execution results:

| Script  | Date                    | Status | Runtime |
|---|-------------------------|--------|---------|
| Untitled - 1  | Nov 9, 2022 12:58:14 PM | ✓ 1    | 0.006 s |
| UPDATE USERD SET NUMBER = '6359885212' WHERE USERNAME = 'Pragadish' |                         | ✓      | 0.006 s |
| Untitled - 1  | Nov 9, 2022 12:56:34 PM | ✓ 1    | 0.005 s |
| UPDATE USERD SET NUMBER = '965487555' WHERE USERNAME = 'Pragadish'  |                         | ✓      | 0.005 s |



The screenshot shows the IBM Db2 on Cloud 'Tables' view for the table 'TBC72932.USERD'. The table structure is as follows:

| USERNAME  | EMAIL                    | PASSWORD      | NUMBER     |
|-----------|--------------------------|---------------|------------|
| Mridhul   | mridhulvinu123@gmail.com | D7M\$%lw4n6Qi | 6380951390 |
| Pragadish | pragadish@gmail.com      | VfrA&83Hn!U6  | 6359885212 |

Buttons for 'Back' and 'Export to CSV' are visible in the top right corner.

DELETE TABLE:

IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

TBC72932.USERD

Back

Export to CSV

| USERNAME         | EMAIL                    | PASSWORD       | NUMBER     |
|------------------|--------------------------|----------------|------------|
| Mridhul          | mridhulvinu123@gmail.com | D7M\$%lw4n6Oi  | 6380951390 |
| Pragadish        | pragadish@gmail.com      | VfrA&83HnIU6   | 965487555  |
| Priyadharshika R | Priyadharshika@gmail.com | \$yi\$qVVN1z07 | 9566482047 |

IBM Db2 on Cloud

Data objectsSaved objects

Filter objects

TBC72932

\*Untitled - 1

1DELETE FROM USERD WHERE USERNAME='Priyadharshika R'

History

Find history

| Script  | Date                    | Status | Runtime |
|---|-------------------------|--------|---------|
| Untitled - 1  | Nov 9, 2022 12:59:56 PM | 1      | 0.00s   |
| DELETE FROM USERD WHERE USERNAME='Priyadharshika R' |                         |        | 0.00s   |

IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

TBC72932.USERD

Back

Export to CSV

| USERNAME  | EMAIL                    | PASSWORD      | NUMBER     |
|-----------|--------------------------|---------------|------------|
| Mridhul   | mridhulvinu123@gmail.com | D7M\$%lw4n6Oi | 6380951390 |
| Pragadish | pragadish@gmail.com      | VfrA&83HnIU6  | 6359885212 |

3.Connect python code to db2.

**NOTE:- Question 4 contains Question 3 answer**

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

## Answers

### App.py

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

app = Flask(__name__)

dsn_hostname = "815fa4db-dc03-4c70-869a-
a9cc13f33084.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"
dsn_uid = "tbc72932"
dsn_pwd = "4LioQzrIBnfCt6c"
dsn_driver = "{IBM DB2 ODBC DRIVER}"
dsn_database = "BLUDB"
dsn_port = "30367"
dsn_security = "SSL"
dsn = ("DRIVER={0};"
"DATABASE={1};"
"HOSTNAME={2};"
"PORT={3};"
"UID={4};"
"PWD={5};"
"SECURITY={6};").format(dsn_driver,dsn_database,dsn_hostname,dsn_port,dsn_uid,dsn_pwd,dsn_security)
print(dsn)
try:
    conn = ibm_db.pconnect(dsn,"","")
    print("success")
except:
    print(ibm_db.conn_errormsg())

@app.route("/") , methods=['GET', 'POST'])
def register():
    if request.method == 'POST':
        sql_stmt = "insert into USER values(?,?,?,?)"
        stmt = ibm_db.prepare(conn, sql_stmt)
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        number = request.form['number']
        ibm_db.bind_param(stmt, 1, username)
        ibm_db.bind_param(stmt, 2, email)
        ibm_db.bind_param(stmt, 3, password)
        ibm_db.bind_param(stmt, 4, number)
        try:
```

```

        ibm_db.execute(stmt)
        return redirect('/')
    except:
        print(ibm_db.stmt_errormsg())

    return render_template('index.html')

@app.route("/login", methods=('GET', 'POST'))
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']
        query = "select COUNT(*) from user where username='"+username+"' and
password='"+password+"'"
        stmt5 = ibm_db.exec_immediate(conn, query)
        row = ibm_db.fetch_tuple(stmt5)
        if(row[0] ==1 ):
            return redirect("/shop/")
    return render_template("index.html")

@app.route("/shop/" , methods=['GET', 'POST'])
def shop():
    sql = "SELECT * FROM job"
    job_name = []
    job_location = []
    job_role = []
    job_number = []
    job_salary = []
    job_image = []
    stmt = ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_assoc(stmt)
    while dictionary != False:
        job_name.append(dictionary["NAME"])
        job_location.append(dictionary["LOCATION"])
        job_role.append(dictionary["ROLE"])
        job_number.append(dictionary["NUMBER"])
        job_salary.append(dictionary["SALARY"])
        job_image.append(dictionary["IMAGE"])
        dictionary = ibm_db.fetch_assoc(stmt)
    return render_template('shop.html', len = len(job_name), job_name = job_name,
job_location = job_location, job_role = job_role, job_number = job_number,
job_salary = job_salary, job_image = job_image)
if __name__ == "__main__":
    app.run(debug=True, host='0.0.0.0')

```

## Index.html

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Sign Up | By Code Info</title>

```

```

<link rel="stylesheet" href="log.css" />
<link
  href="https://fonts.googleapis.com/css2?family=Roboto:wght@300&display=swap"
  rel="stylesheet"
/>
</head>
<body>
  <div class="signup-box">
    <h1>Sign Up</h1>
    <form>
      <label>First Name</label>
      <input type="text" placeholder="" />
      <label>Last Name</label>
      <input type="text" placeholder="" />
      <label>Email</label>
      <input type="email" placeholder="" />
      <label>Password</label>
      <input type="password" placeholder="" />
      <label>Confirm Password</label>
      <input type="password" placeholder="" />
      <input type="button" value="Submit" />
    </form>
    <p>
      By clicking the Sign Up button, you agree to our <br />
      <a href="#">Terms and Condition</a> and <a href="#">Policy Privacy</a>
    </p>
  </div>
  <p class="para-2">
    Already have an account? <a href="signin.html">Login here</a>
  </p>
</body>
</html>

```

### Style.css

```

body {
  background-color: #344a72;
  font-family: "Roboto", sans-serif;
}

.signup-box {
  width: 360px;
  height: 620px;
  margin: auto;
  background-color: white;
  border-radius: 3px;
}

.login-box {
  width: 360px;

```

```
height: 280px;
margin: auto;
border-radius: 3px;
background-color: white;
}

h1 {
  text-align: center;
  padding-top: 15px;
}

h4 {
  text-align: center;
}

form {
  width: 300px;
  margin-left: 20px;
}

form label {
  display: flex;
  margin-top: 20px;
  font-size: 18px;
}

form input {
  width: 100%;
  padding: 7px;
  border: none;
  border: 1px solid gray;
  border-radius: 6px;
  outline: none;
}

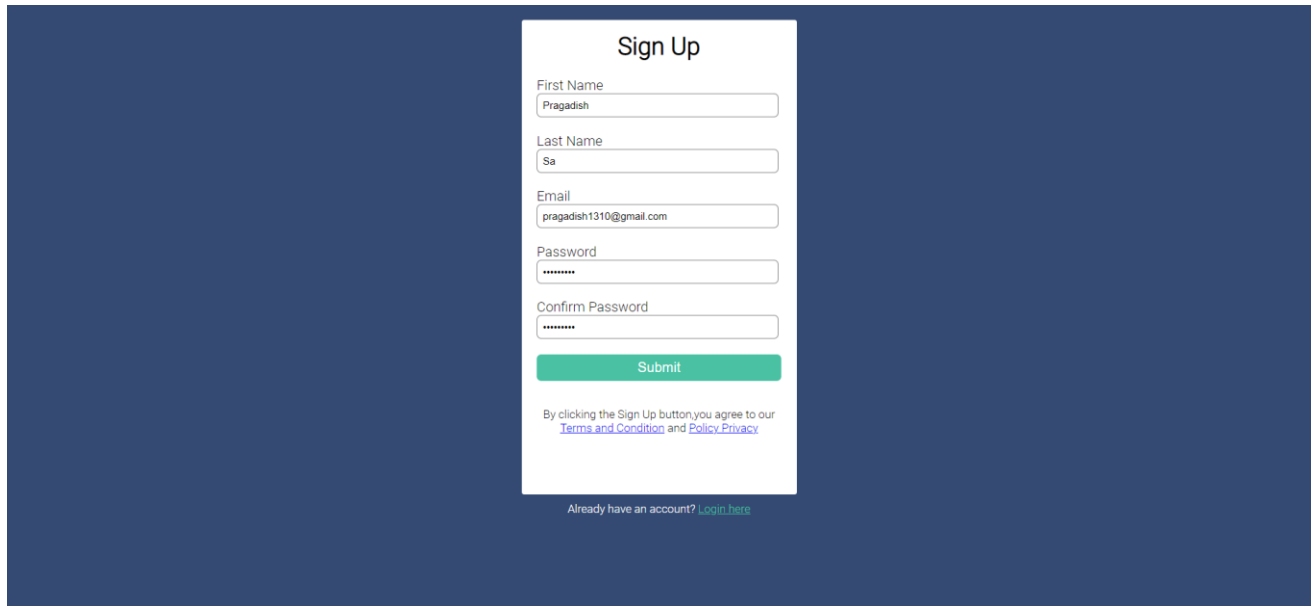
input[type="button"] {
  width: 320px;
  height: 35px;
  margin-top: 20px;
  border: none;
  background-color: #49c1a2;
  color: white;
  font-size: 18px;
}

p {
  text-align: center;
  padding-top: 20px;
  font-size: 15px;
}

.para-2 {
  text-align: center;
  color: white;
  font-size: 15px;
}
```

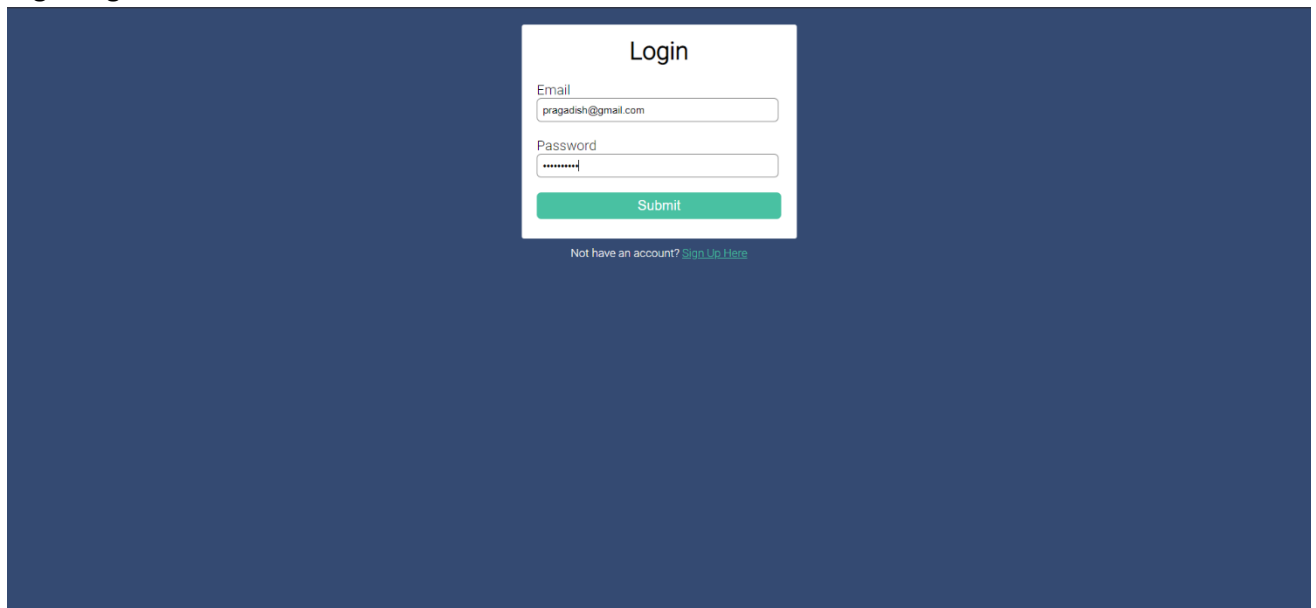
```
margin-top: -10px;
}
.para-2 a {
  color: #49c1a2;
}
```

### Output page:



The image shows a 'Sign Up' form centered on a dark blue background. The form is white with a title 'Sign Up' at the top. It contains five input fields: 'First Name' (with 'Pragadish'), 'Last Name' (with 'Sa'), 'Email' (with 'pragadish1310@gmail.com'), 'Password' (with masked characters), and 'Confirm Password' (with masked characters). Below the fields is a green 'Submit' button. At the bottom, there is a line of text: 'By clicking the Sign Up button, you agree to our [Terms and Condition](#) and [Policy Privacy](#)'. Below that, it says 'Already have an account? [Login here](#)'.

### Login Page:



The image shows a 'Login' form centered on a dark blue background. The form is white with a title 'Login' at the top. It contains two input fields: 'Email' (with 'pragadish@gmail.com') and 'Password' (with masked characters). Below the fields is a green 'Submit' button. At the bottom, there is a line of text: 'Not have an account? [Sign Up Here](#)'.




## Welcome Page:


# Job Recommendation

HomeAboutContactProfile


Search

**Wipro - Bengaluru (5.0 LPA )**  
**Software Engineer**  
Contact: +91(80)46827999

REGISTER

**ACCENTURE - Chennai (4.5 LPA )**  
**Software Engineer**  
Contact: +918061180002

REGISTER

**ACCENTURE - Chennai (4.0 LPA )**  
**Programmer Analyst**  
Contact: +91638051390

REGISTER

## Data Base Table

IBM Db2 on Cloud

Load DataLoad HistoryTablesViewsIndexesAliasesMQTsSequencesApplication objects

TBC72932.USERD

Back

Export to CSV

| USERNAME         | EMAIL                    | PASSWORD       | NUMBER     |
|------------------|--------------------------|----------------|------------|
| Mridhul          | mridhulvinu123@gmail.com | D7M\$%lw4n6Oi  | 6380951390 |
| Pragadish        | pragadish@gmail.com      | VfrA&83HntU6   | 6359885212 |
| Priyadharshika R | Priyadharshika@gmail.com | \$yi\$qVVN1z07 | 9566482047 |