

## Assignment -2

### IBM CLOUD DB2

|                     |                   |
|---------------------|-------------------|
| Assignment Date     | 29 September 2022 |
| Student Name        | DINESH M          |
| Student Roll Number | 611819205006      |
| Maximum Marks       | 2 Marks           |

#### Questions:

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

The first screenshot shows the IBM Db2 on Cloud console interface. The 'Tables' tab is selected, and a search for 'USERTABLE' in the 'DTP46044' schema has been performed. The 'Table definition' panel on the right displays the table's structure:

| Name       | Data type | Nullable | Length | Scale |
|------------|-----------|----------|--------|-------|
| USERNAME   | CHAR      | Y        | 5      | 0     |
| ROLLNUMBER | INTEGER   | Y        |        | 0     |
| EMAIL      | CHAR      | Y        | 5      | 0     |
| PASSWORD   | VARCHAR   | Y        | 32     | 0     |

The second screenshot shows the 'DTP46044.USERTABLE' page. It displays the table's columns: USERNAME, ROLLNUMBER, EMAIL, and PASSWORD. Below the column list, a message states 'There is no data here yet', indicating that the table has been successfully created but is currently empty.

2. Perform UPDATE,DELETE Queries with user table.

### UPDATE TABLE:-

The screenshot shows the IBM Db2 on Cloud console interface. On the left, the 'Data objects' panel displays a table named 'DTP46044'. The main editor area shows a SQL script with the following query:

```
1 INSERT INTO USERTABLE VALUES ('VK',21,'VK@gamil.com','1234')
```

The query has been executed successfully. The 'History' panel at the bottom shows the execution details:

| Script   | Date                     | Status | Runtime |
|--|--------------------------|--------|---------|
| Untitled - 1   | Oct 10, 2022 12:51:13 PM | ✓ 1    | 0.018 s |
| INSERT INTO USERTABLE VALUES ('VK',21,'VK@gamil.com','1234') |                          | ✓      | 0.018 s |

The screenshot shows the IBM Db2 on Cloud console interface displaying the contents of the 'DTP46044.USERTABLE' table. The table has the following columns: USERNAME, ROLLNO, EMAIL, and PASSWORD. The data is as follows:

| USERNAME | ROLLNO | EMAIL        | PASSWORD |
|----------|--------|--------------|----------|
| VK       | 21     | VK@gamil.com | 1234     |

## DELETE TABLE:-

The screenshot shows the IBM Db2 on Cloud web interface. The top navigation bar includes 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected, displaying the table 'DTP46044.USERTABLE'. A 'Back' button is in the top right. Below the table name, there is an 'Export to CSV' button. The table data is as follows:

| USERNAME | ROLLNO | EMAIL        | PASSWORD |
|----------|--------|--------------|----------|
| PARAMESH | 21     | VK@gamil.com | 1234     |
| VK       | 23     | VK@gamil.com | 12D34    |

The screenshot shows the IBM Db2 on Cloud web interface with the SQL editor open. The 'Data objects' panel on the left shows the table 'DTP46044'. The SQL editor contains the following query:

```
1 DELETE FROM USERTABLE WHERE ROLLNO = 23 ;
```

The 'Run all' button is visible. Below the editor, the 'History' tab is selected, showing a table of execution history:

| Script  | Date                     | Status | Runtime |
|---|--------------------------|--------|---------|
| Untitled - 1  | Oct 10, 2022 1:02:49 PM  | ✓ 1    | 0.006 s |
| DELETE FROM USERTABLE WHERE ROLLNO = 23                       |                          | ✓      | 0.006 s |
| Untitled - 1  | Oct 10, 2022 1:00:11 PM  | ✓ 1    | 0.007 s |
| INSERT INTO USERTABLE VALUES ('VK',23,'VK@gamil.com','12034') |                          | ✓      | 0.007 s |
| Untitled - 1  | Oct 10, 2022 12:59:55 PM | ✗ 1    | 0.044 s |

IBM Db2 on Cloud

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

DTP46044.USERTABLE Back

Export to CSV

| USERNAME | ROLLNO | EMAIL        | PASSWORD |
|----------|--------|--------------|----------|
| PARAMESH | 21     | VK@gamil.com | 1234     |

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
import re

app = Flask(__name__)

app.secret_key = 'a'

conn = ibm_db.connect(
    "DATABASE=bludb;HOSTNAME=fbd88901-ebdb-4a4f-a32e-9822b9fb237b.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32731;USERNAM
E=ntp46044;PASSWORD=95soX0sZGhb4ToUj;SECURITY=SSL;SSLSERVERCERTIFICATE=DigiCer
tGlobalRootCA.crt;", "", "")

@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(prepare_stmt, 1, username)
    ibm_db.bind_param(prepare_stmt, 2, email)
    ibm_db.bind_param(prepare_stmt, 3, password)
    ibm_db.execute(prepare_stmt)
    msg = 'You have successfully registered !'
elif request.method == 'POST':
    msg = 'Please fill out 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 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:
```

```
            session['loggedin'] = True
```

```
            session['id'] = account['USERNAME']
```

```
            userid = account['USERNAME']
```

```
            session['username'] = account['USERNAME']
```

```
            msg = 'Logged in successfully !'
```

```

        msg = 'Logged in successfully !'
        return render_template('dashboard.html', msg=msg)
    else:
        msg = 'Incorrect username / password !'
        return render_template('login.html', msg=msg)

if __name__ == '__main__':
    app.run(host='0.0.0.0')
    # app.run(debug=True)

```

## Register.html

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Smart Fashion register</title>

    <!-- bootstrap css cdn -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
"
        integrity="sha384-
JcKb8q3iqJ61gNV9KGb8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">
    <!-- css stylesheet -->
    <link rel="stylesheet" href="/static/css/style.css">
    <!-- font styles cdn -->
    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link
href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap"
rel="stylesheet">
    <link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swap"
rel="stylesheet">
</head>

<body>
    <!-- bootstrap navbar -->
    <!-- navbar ends -->
    <!-- Login form -->
    <div class="login text-center mt-5">

```

```

        <form action="/" method="post">
            <h2 class="form-text-h2"> Register Form </h2>
            <div class="msg">{{ msg }}</div>
            <input type="text" class="form-control" name="username"
placeholder="Enter Your Username" id="username"
                required>
            <input type="email" class="form-control" name="email"
placeholder="Enter Your Email ID" id="email" required>
            <input type="password" class="form-control" name="password"
placeholder="Enter Your Password" id="password"
                required>
            <button type="submit" id="button" class="btn btn-primary">
Register </button>
            <div class="note mt-3 text-center">
                <!--Register form -->
                <p> already have an account ? please login <a
href="/login">login! </a> </p>

            </div>
        </form>

    </div>

</body>

</html>

```

### Login.html

```

<!DOCTYPE html>
<html lang="en">

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Smart Fashion Login</title>
    <!-- favicon -->
    <!-- <link rel="shortcut icon" href="/assets/img/favicon.ico"
type="image/x-icon"> -->
    <!-- <link rel="icon" href="/assets/img/favicon.ico" type="image/x-icon">
-->
    <link rel="icon" type="image/png" sizes="16x16" href="/assets/img/favicon-
32x32.png">
    <!-- bootstrap css cdn -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css"
    "

```

```

        integrity="sha384-
JcKb8q3iqJ61gNV9KgB8thSsNjpSL0n8PARn9HuZOnIxN0hoP+VmmDGMN5t9UJ0Z"
crossorigin="anonymous">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/4.7.0/css/font-awesome.css">
    <!-- css stylesheet -->
    <link rel="stylesheet" href="/static/css/style2.css">
    <!-- font styles cdn -->
    <link rel="preconnect" href="https://fonts.gstatic.com">
    <link
href="https://fonts.googleapis.com/css2?family=Alegreya&display=swap"
rel="stylesheet">
    <link
href="https://fonts.googleapis.com/css2?family=Alegreya:wght@600&display=swap"
rel="stylesheet">
</head>

<body>

    <div class="login text-center mt-5">

        <form action="/login" method="post">
            <h2 class="form-text-h2"> Login Form </h2>
            <div class="msg">{{ msg }}</div>
            <div class="form-group mb-3">
                <label class="label" for="username">Username</label>
                <input type="text" class="form-control" placeholder="Username"
name="username" required>
            </div>
            <div class="form-group mb-3">
                <label class="label" for="password">Password</label>
                <input type="password" class="form-control"
placeholder="Password" name="password" required>
            </div>

            <button type="submit" id="button" class="btn btn-primary"> Login
</button>

            <div class="note mt-3 text-center">
                <!--Register form -->
                <p> Don't have an account yet? Click here to <a
href="/">register! </a> </p>
            </div>
        </form>
    </div>

</body>

</html>

```



## Style.css

```
* {
  margin: 0;
  padding: 0;
}

body {
  background-color: aqua !important;
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
}

form {
  display: flex;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  background-color: white;
  width: 40%;
  height: 450px;
  border-radius: 50px;
  margin: 100px;
  gap: 25px;
}

h2 {
  font-size: large;
  font-weight: bolder;
}

input {
  width: 80% !important;
}

div {
  display: flex;
  align-items: center;
  justify-content: center;
  flex-direction: column;
  width: 90%;
}

.form-text-vk {
  font-size: small;
  padding-bottom: 10px;
}
```

```
}  
  
.form-text-h2 {  
  font-size: x-large;  
  padding-top: 10px;  
}
```

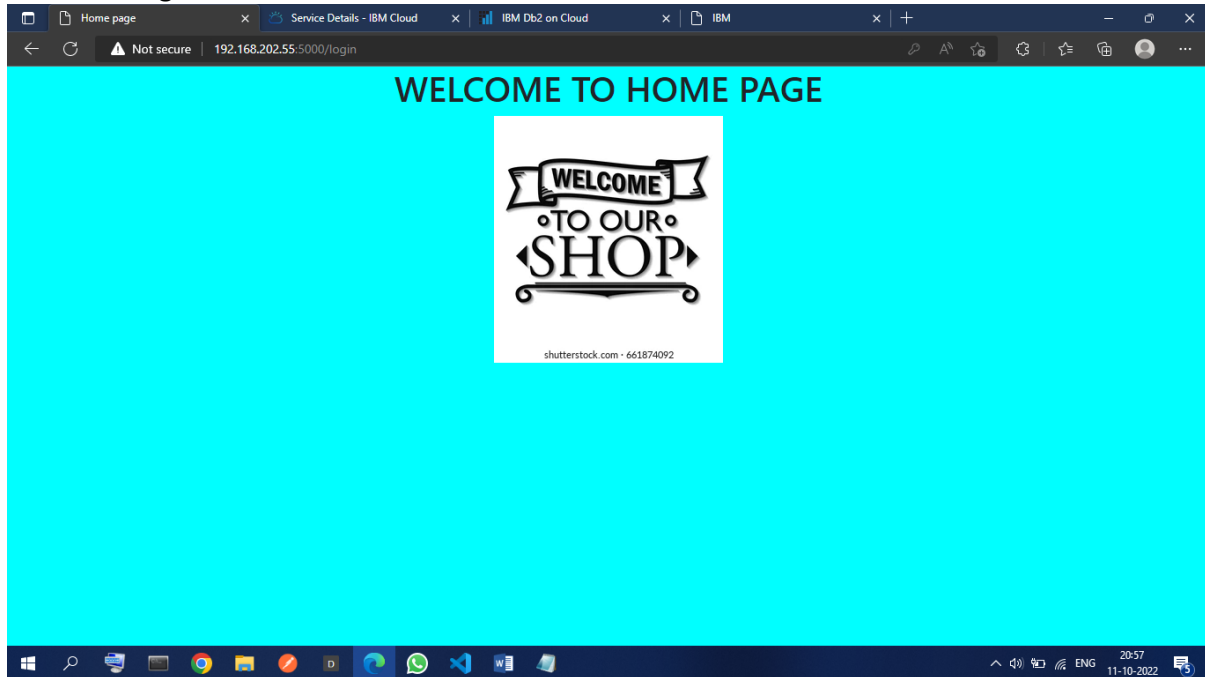
## Output:- registration page:-

The screenshot shows a web browser window with the URL `192.168.202.55:5000`. The page has a solid blue background. In the center, there is a white rounded rectangle titled "Register Form". Inside the form, there are three input fields: the first contains "Parameshwaran", the second contains "parameshvk3232@gmail.com", and the third is labeled "Enter Your Password". Below these fields is a blue "Register" button. At the bottom of the form, there is a link: "already have an account ? please login [login!](#)". The browser's taskbar at the bottom shows various icons and the system clock indicating 20:56 on 11-10-2022.

## Login Page:-

The screenshot shows a web browser window with the URL `192.168.202.55:5000/login`. The page has a solid blue background. In the center, there is a white rounded rectangle titled "Login Form". Inside the form, there are two input fields: the first is labeled "Username" and contains "Parameshwaran", and the second is labeled "Password" and contains "\*\*\*\*\*" with a toggle icon. Below these fields is a blue "Login" button. At the bottom of the form, there is a link: "Don't have an account yet? Click here to [register!](#)". The browser's taskbar at the bottom shows various icons and the system clock indicating 20:57 on 11-10-2022.

## Welcome Page:-



## Data Base Table :-

The screenshot shows the IBM Db2 on Cloud console. The top navigation bar includes 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MQTs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected, and the table 'DTP46044.USERS' is displayed. The table has three columns: 'USERNAME', 'EMAIL', and 'PASSWORD'. The data row shows 'Parameshwaran' as the username, 'parameshvk3232@gmail.com' as the email, and '123432' as the password. There are 'Back' and 'Export to CSV' buttons in the top right corner of the table view.

| USERNAME      | EMAIL                    | PASSWORD |
|---------------|--------------------------|----------|
| Parameshwaran | parameshvk3232@gmail.com | 123432   |