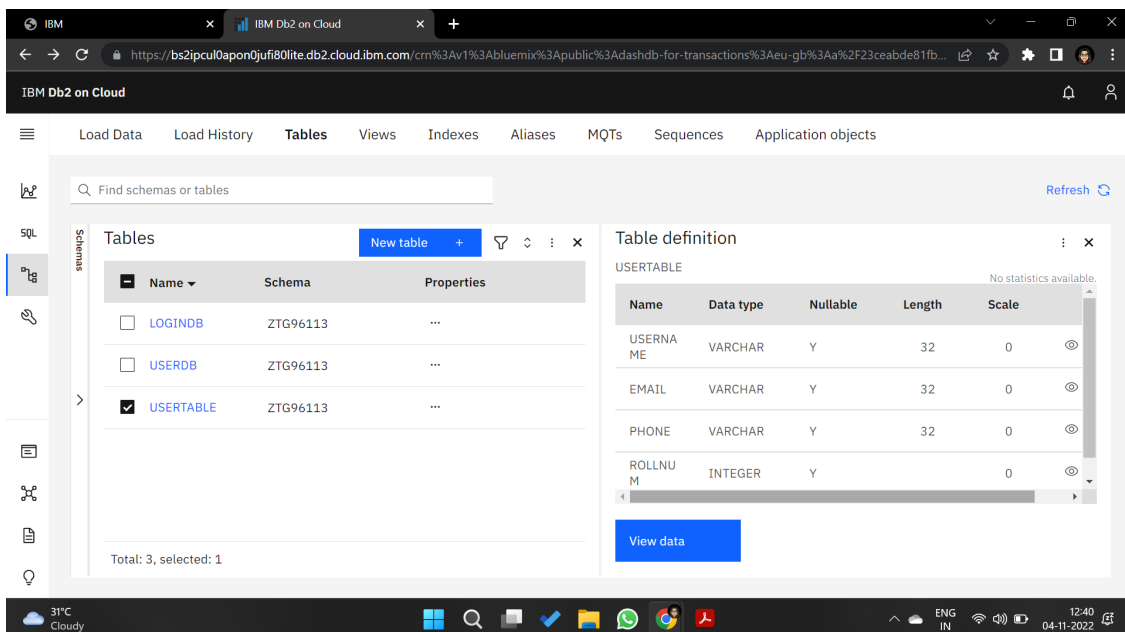
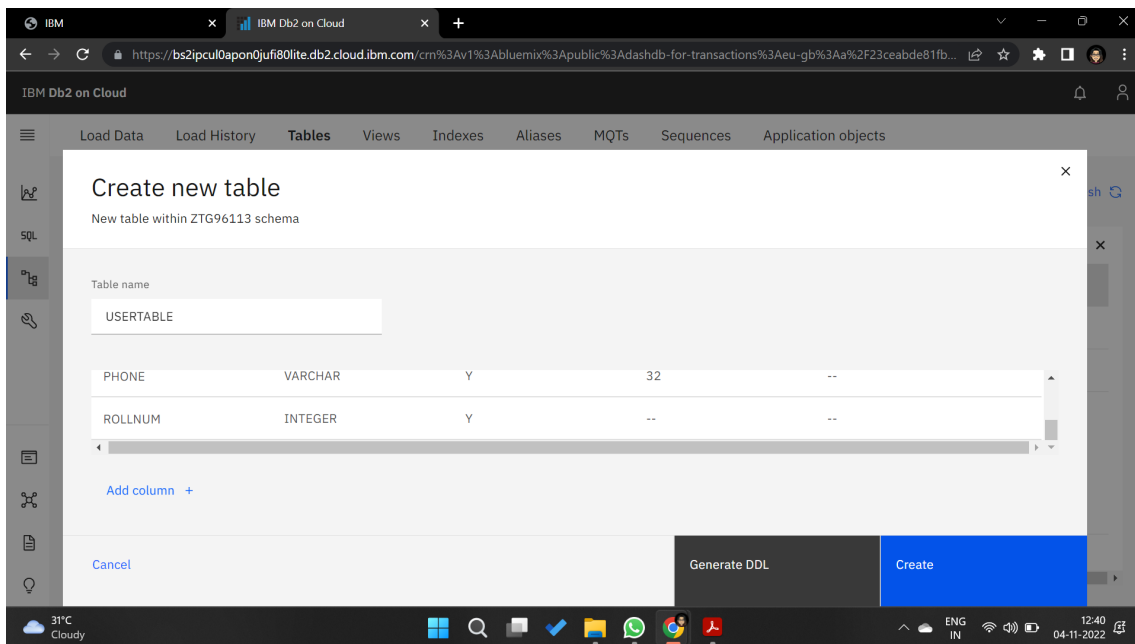


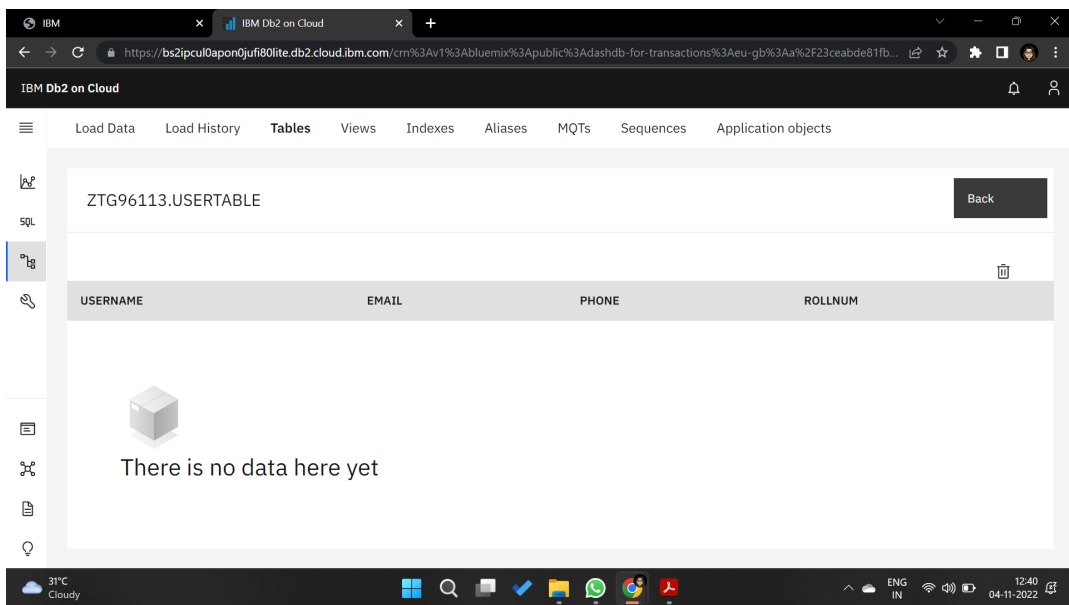
ASSIGNMENT 2

Team ID	PNT2022TMID17803
Project Name	Plasma Donor Application
Student ID	713319CS003
Student Name	Abhirami S S

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

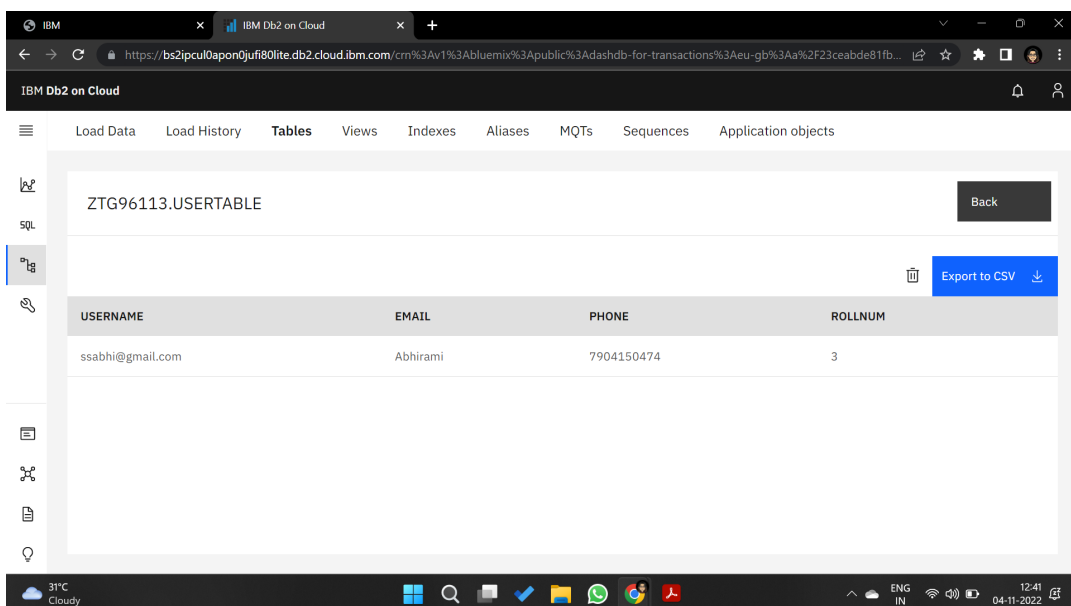
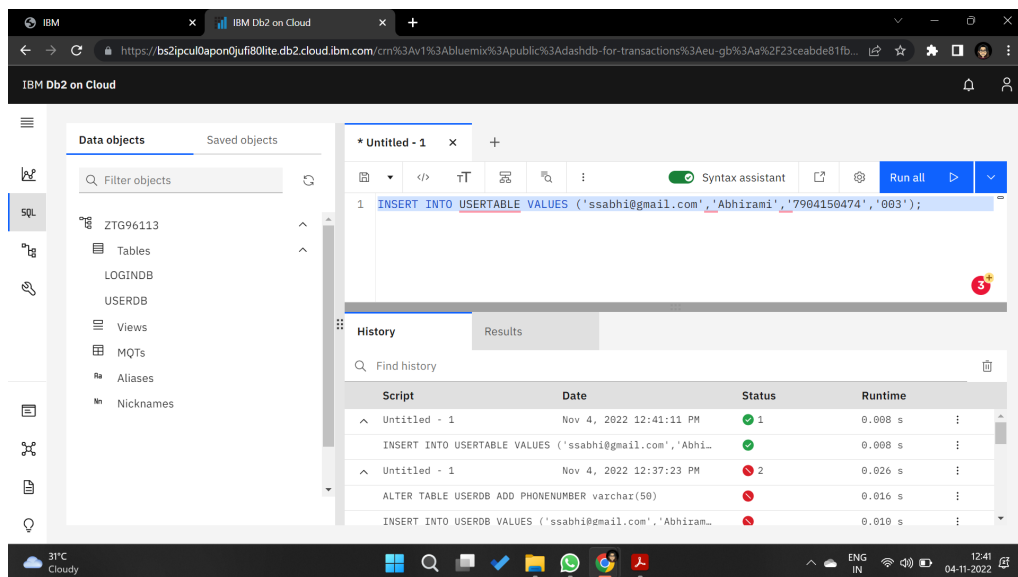
Using IBM DB2 Cloud USERTABLE is created as follows:





2. Perform UPDATE,DELETE Queries with user table

Inserting Values to USERTABLE



Using UPDATE to update values to the table:

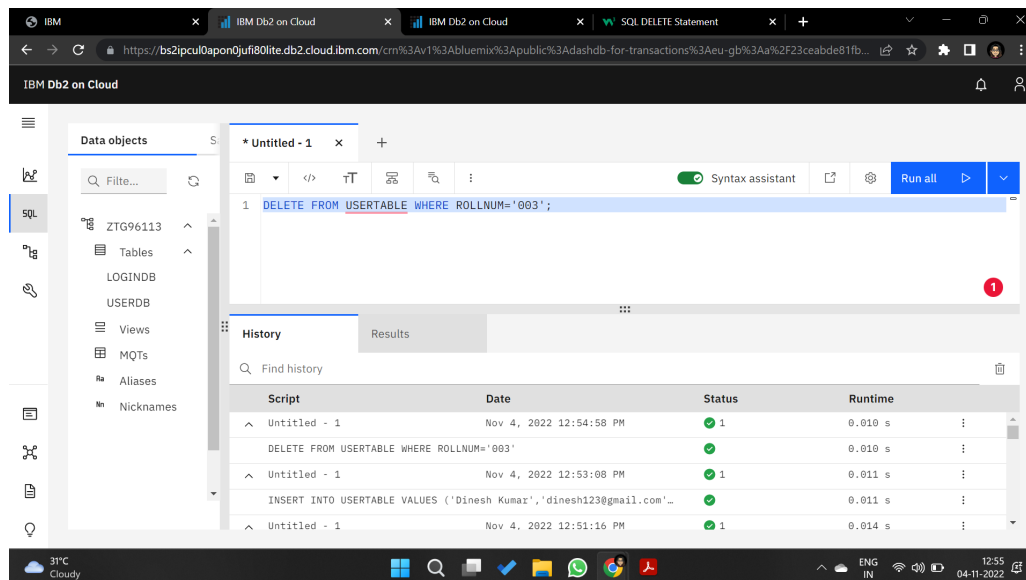
The screenshot shows the IBM Db2 on Cloud SQL editor interface. The top navigation bar includes 'Data objects' and 'SQL'. The left sidebar shows a tree view of the database structure, including 'ZTG96113', 'Tables', 'LOGINDB', 'USERDB', 'Views', 'MQTs', 'Aliases', and 'Nicknames'. The main editor area displays an SQL statement: `UPDATE USERTABLE SET EMAIL='kishore136@gmail.com', USERNAME = 'Kishore Kumar', PHONE='8608583693' WHERE ROLL`. Below the editor, the 'History' tab is active, showing a table of execution history.

Script	Date	Status	Runtime
Untitled - 1	Nov 4, 2022 12:46:03 PM	✓ 1	0.009 s
UPDATE USERTABLE SET EMAIL='kishore136@gmail.com', USERNAME = 'Kis...		✓	0.009 s
Untitled - 1	Nov 4, 2022 12:45:17 PM	⚠ 1	0.008 s
UPDATE USERTABLE SET EMAIL='kishore136@gmail.com', USERNAME = 'Kis...		⚠	0.008 s
Untitled - 1	Nov 4, 2022 12:41:11 PM	✓ 1	0.008 s

The screenshot shows the IBM Db2 on Cloud interface displaying the data in the 'ZTG96113.USERTABLE' table. The table has four columns: 'USERNAME', 'EMAIL', 'PHONE', and 'ROLLNUM'. The data row shows 'Kishore Kumar' with email 'kishore136@gmail.com', phone '8608583693', and roll number '67'. The interface includes a 'Back' button and an 'Export to CSV' button.

USERNAME	EMAIL	PHONE	ROLLNUM
Kishore Kumar	kishore136@gmail.com	8608583693	67

Using DELETE to delete values to the table:



Before deletion :

The screenshot shows the IBM Db2 on Cloud interface displaying the data in the ZTG96113.USERTABLE table. The table has the following columns: USERNAME, EMAIL, PHONE, ROLLNUM, and ID.

USERNAME	EMAIL	PHONE	ROLLNUM	ID
Charu Nethra	charu@gmail.com	8937465328	25	4
Dinesh Kumar	dinesh123@gmail.com	9452830234	508	3
Kishore Kumar	kishore136@gmail.com	8608583693	67	1
ssabhi@gmail.com	Abhirami	7904150474	3	2

After deletion :

The screenshot shows the IBM Db2 on Cloud interface displaying the data in the ZTG96113.USERTABLE table after the deletion of the record with ROLLNUM='003'. The table has the following columns: USERNAME, EMAIL, PHONE, ROLLNUM, and ID.

USERNAME	EMAIL	PHONE	ROLLNUM	ID
Charu Nethra	charu@gmail.com	8937465328	25	4
Dinesh Kumar	dinesh123@gmail.com	9452830234	508	3
Kishore Kumar	kishore136@gmail.com	8608583693	67	1

3. Connect python code to db2.

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

App.py to connect python-flask with IBM DB2

```
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=#;HOSTNAME=#;PORT=32731;USERNAME=#PASSWORD=#;SECURITY=SSL;SSL
    SERVERCERTIFICATE=DigiCertGlobalRootCA.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)

```

Login.html for first vist registration:

```

<!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>Signin page </title>

<style>
body{

    background-color:#eacefa !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:#fff;

    width: 40%;

    height: 430px;

    border-radius: 50px;

    margin: 100px;

}

.text1{

font-family: "Arial Black", Gadget, sans-serif;

font-size: 15px;

text-align: left;

letter-spacing: 1.2px;

word-spacing: 2px;

color: #eacefa;

font-weight: 400;

text-transform: capitalize;

padding: 1.6%;
```



```
}

h1{

font-family: "Arial Black", Gadget, sans-serif;

font-size: 45px;

letter-spacing: 2px;

word-spacing: 2px;

color: #9999ff;

text-shadow: 2px 2px 0px #FFFFFF, 5px 4px 0px rgba(0,0,0,0.15), 2px 7px
2px rgba(230,204,255,0);

}


div{

    display: flex;

    align-items: center;

    justify-content: center;

    flex-direction: column;

    width: 90%;

}

.form-box{

    width: 90% !important;

    height:90% ;

    border-radius: 25px 25px;

    padding: 2.9%;

    border: 3px solid #A4A4A4;

}

.button

{

    background: #D3F6FA;

    border: #FAE5D3;

    border-radius: 28px;

    color: grey;

    font-size: 20px;

    padding: 10px 20px 10px 20px;
```

```
text-decoration: none;

margin: 5%;
}

.button:hover {

background: #FAE5D3;

color: black;
}

</style>

</head>

<body>

<form method="post" action="login">

    <form action="/form_login" method="post">

        <h1>LOG-IN</h1>

        <div class="box">

            <label for="name" class="text1">Name</label>

            <input type="text" class="form-box" id="name " name="name"
placeholder="Enter Your Name">

        </div>

        <div class="box">

            <label for="exampleInputEmail1"
class="text1">Mail-Id</label>

            <input type="email" class="form-box" name="email"
id="exampleInputEmail1" placeholder="Enter Your Mail Id">

        </div>

        <div class="box">

            <label for="phone" class="text1">Phone Number</label>

            <input type="text" class="form-box" name="phone" id="phone"
placeholder="Enter Your Mobile Number">

        </div>

    </form>

</form>

</body>

</html>
```

```

        <div class="box">

            <label for="phone" class="text1">Set Password</label>

            <input type="password" class="form-box" name="phone"
id="phone" placeholder="Enter 6-digit password">

        </div>


        <button type="submit" class="button">Submit</button>

    </form>
</body>

</html>

```

Signin.html for already registered users:

```

<!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>Login page </title>

    <style>
body{

    background-color:#eacefa !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:#fff;

    width: 40%;

    height: 430px;

    border-radius: 50px;

    margin: 100px;

}

.text1{

font-family: "Arial Black", Gadget, sans-serif;

font-size: 15px;

text-align: left;

letter-spacing: 1.2px;

word-spacing: 2px;

color: #eacefa;

font-weight: 400;

text-transform: capitalize;

padding: 1.6%;

}

h1{

font-family: "Arial Black", Gadget, sans-serif;

font-size: 45px;

letter-spacing: 2px;

word-spacing: 2px;

color: #9999ff;

text-shadow: 2px 2px 0px #FFFFFF, 5px 4px 0px rgba(0,0,0,0.15), 2px 7px

2px rgba(230,204,255,0);

}

h3{
```

```
font-family: "Arial Black", Gadget, sans-serif;

font-size: 14px;

letter-spacing: 1.2px;

color: grey;

}

div{

    display: flex;

    align-items: center;

    justify-content: center;

    flex-direction: column;

    width: 90%;

}

.form-box{

    width: 90% !important;

    height: 90% ;

    border-radius: 25px 25px;

    padding: 2.9%;

    border: 3px solid #A4A4A4;

}

.button

{

    background: #D3F6FA;

    border: #FAE5D3;

    border-radius: 28px;

    color: gray;

    font-size: 20px;

    padding: 10px 20px 10px 20px;

    text-decoration: none;

    margin: 5%;

}

.button:hover {

    background: #FAE5D3;

    color: black;
```

```

}

</style>
</head>

<body>

    <form method="post" action="login">

        <form action="/form_login" method="post">

            <h1>SIGN-IN</h1>

            <div class="box">

                <label for="name" class="text1">Username</label>

                <input type="text" class="form-box" id="name " name="name"
placeholder="Enter Your Name">

            </div>

            <div class="box">

                <label for="password" class="text1">Password</label>

                <input type="password" class="form-box" name="phone"
id="phone" placeholder="Enter Your Password">

            </div>

            <button type="submit" class="button">Sign-in</button>

            <h3>If you are a new user try logging-in first</h3>

        </form>
    </body>

</html>

```

Index.html as a welcoming dashboard redirecting after singing in:

```

<!DOCTYPE html>

<html>

<head>

```

```
<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Welcome!</title>

<style>

@import url(https://fonts.googleapis.com/css?family=Khula:700);

body {

    background: #111;

    background-color: antiquewhite;

}

.hidden {

    opacity:0;

}

.console-container {

    font-family:Khula;

    font-size:4em;

    text-align:center;

    height:200px;

    width:600px;

    display:block;

    position:absolute;

    color:white;

    top:0;

    bottom:0;

    left:0;

    right:0;

    margin:auto;

}

.console-underscore {

    display:inline-block;

    position:relative;

    top:-0.14em;

    left:10px;

}
```

```
</style>

</head>

<body>

<div class="main">

    <div class='console-container'><span id='text'></span>

        <div class='console-underscore' id='console'>&#95;</div></div>

</div>

<script>

    consoleText(['Welcome to the website','Team : PNT2022TMID17803',
'Wait for sometime.'], 'text',['tomato','rebeccapurple','lightblue']);

function consoleText(words, id, colors) {

    if (colors === undefined) colors = ['#fff'];

    var visible = true;

    var con = document.getElementById('console');

    var letterCount = 1;

    var x = 1;

    var waiting = false;

    var target = document.getElementById(id)

    target.setAttribute('style', 'color:' + colors[0])

    window.setInterval(function() {

        if (letterCount === 0 && waiting === false) {

            waiting = true;

            target.innerHTML = words[0].substring(0, letterCount)

            window.setTimeout(function() {

                var usedColor = colors.shift();

                colors.push(usedColor);

                var usedWord = words.shift();

                words.push(usedWord);

                x = 1;

                target.setAttribute('style', 'color:' + colors[0])
```



```
        letterCount += x;

        waiting = false;

    }, 1000)

    } else if (letterCount === words[0].length + 1 && waiting ===
false) {

        waiting = true;

        window.setTimeout(function() {

            x = -1;

            letterCount += x;

            waiting = false;

        }, 1000)

    } else if (waiting === false) {

        target.innerHTML = words[0].substring(0, letterCount)

        letterCount += x;

    }

}, 120)

window.setInterval(function() {

    if (visible === true) {

        con.className = 'console-underscore hidden'

        visible = false;

    } else {

        con.className = 'console-underscore'

        visible = true;

    }

}, 400)

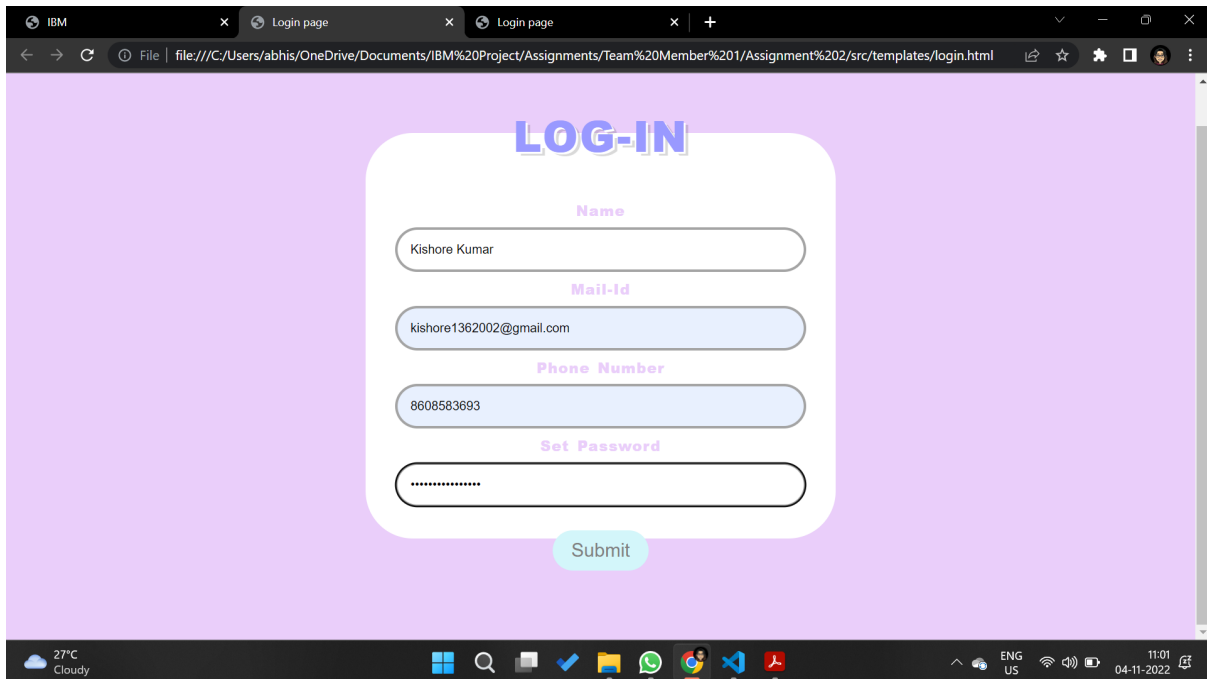
}

</script>

</body>

</html>
```

OUTPUT :



IBM

File | file:///C:/Users/abhis/OneDrive/Documents/IBM%20Project/Assignments/Team%20Member%201/Assignment%202/src/templates/login.html

LOG-IN

Name

Kishore Kumar

Mail-Id

kishore1362002@gmail.com

Phone Number

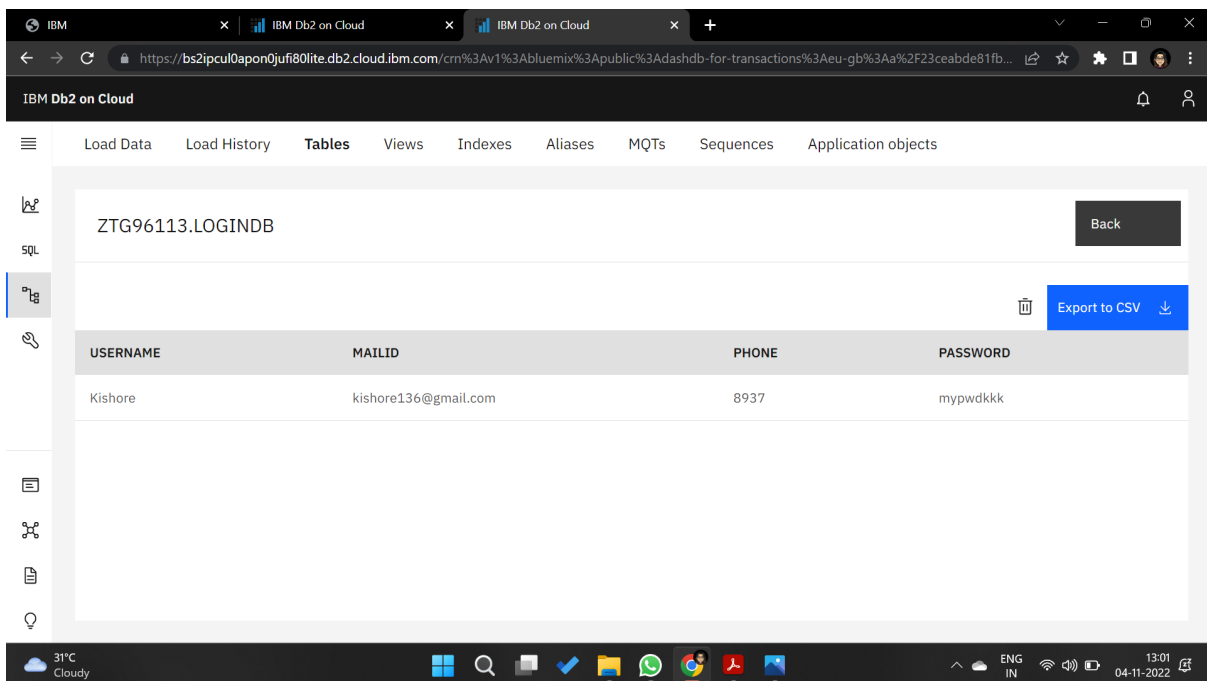
8608583693

Set Password

Submit

27°C Cloudy

11:01 04-11-2022



IBM

https://bs2jpcul0apon0jufi80lite.db2.cloud.ibm.com/crm%3Av1%3Abluemix%3Apublic%3Adashdb-for-transactions%3Aeu-gb%3Aa%2F23ceabde81fb...

IBM Db2 on Cloud

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

ZTG96113.LOGINDB

Back

Export to CSV

USERNAME	MAILID	PHONE	PASSWORD
Kishore	kishore136@gmail.com	8937	mypwdkkk

31°C Cloudy

13:01 04-11-2022

