

# IMPLEMENTING WEB APPLICATION

## Create IBM DB2 And Connect With Python

Team ID	PNT2022TMID17803
Project Name	Plasma Donor Application

### IBM DB2:

IBM Db2 on Cloud interface showing the 'Tables' tab. The 'REGTBL' table is selected, and its definition is shown on the right. The table has columns: USERNAME, EMAIL, PASSWORD, re-password, and PHONNUMBER.

Name	Data type	Nullable	Length	Scale
USERNAME	CHAR	Y	5	0
EMAIL	CHAR	Y	5	0
PASSWORD	CHAR	Y	5	0
re-password	CHAR	Y	5	0

IBM Db2 on Cloud interface showing the 'REGTBL' table data. The table has columns: USERNAME, EMAIL, PASSWORD, re-password, and PHONNUMBER. The data row shows: Kis, ksho, pass, pass, 999.

USERNAME	EMAIL	PASSWORD	re-password	PHONNUMBER
Kis	ksho	pass	pass	999

## Login Page:

**LOGIN PAGE**

Welcome onboard

Don't Have an account try [REGISTER](#)

Your Name  
Kishore Kumar

Password  
.....

Log In

## Registration Page:

**REGISTRATION PAGE**

New User Registration

Already Have an Account [LOGIN](#)

Your Name  
Kishore

Password  
.....

Re-Enter Password  
.....

Email-ID  
kishore123@gmail.com

Phone Number  
860

Register

## 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=#;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.bind_param(prepare_stmt, 4, re-password)

        ibm_db.bind_param(prepare_stmt, 6, phonnumber)

        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)
```