IMPLEMENTING WEB APPLICATION

Date	11 November 2022
Team ID	PNT2022TMID09659
Project Name	Smart Fashion Recommended Application

- Create IBM DB2 And Connect with Python.

```
    import ibm_db dictionary={}

def printTableData(conn):sql = "SELECT * FROM

    userdetails" out = ibm_db.exec_immediate(conn, sql)

    document = ibm_db.fetch_assoc(out) while document != False:

    dictionary.update({document['USERNAME']:

    document['PASSWORD']})document = ibm_db.fetch_assoc(out)

    def insertTableData(conn,rollno,username,email,password):

    sql="INSERT INTO userdetails(rollno,username,email,password)

    VALUES({},'{}','{}','{}')".format(rollno,username,email,password)

        out = ibm_db.exec_immediate(conn,sql)

        print('Number of affected rows :',ibm_db.num_rows(out),"\n")

    def updateTableData(conn,rollno,username,email,password):

        sql = "UPDATE userdetails

        SET (username,email,password)=('{}','{}','{}')

        WHERE rollno={}".format(username,email,password,rollno)
    out = ibm_db.exec_immediate(conn,sql)

    print('Number of affected rows : ', ibm_db.num_rows(out), "\n")

def deleteTableData(conn,rollno):

    sql = "DELETE FROM userdetails

    WHERErollno={}".format(rollno)

out = ibm_db.exec_immediate(conn, sql)

print('Number of affected rows : ', ibm_db.num_rows(out), "\n")

try:

conn=ibm_db.connect("DATABASE=bludb;

HOSTNAME=0c77d6f2-5da9-48a9-81f8-86b520b87518.bs2io90l

08kqb1od8lcg.databases.appdomain.cloud;PORT=31198;

SECURITY=SSL;

SSLServerCertificate=DigiCertGlobalRootCA.crt;

PROTOCOL=TCPIP;

UID=bjn03696;

PWD=ef96tLJX2VjzaCPX;", "", "")

    print("Db connected")except:

print("Error")

from flask import Flask,render_template,request,url_for,session

app=Flask(_name_)

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

```
      @app.route("/login",methods=['POST','GET'])def login():
            if request.method=="POST":
      printTableData(conn)

username=request.form['username'] password=request.form['password']
try:
      if dictionary[username] == password   and username in dictionary:
          return "Logged in successfully"except:
          return "Invalidusername or password"
          return render_template('loginpage.html')
      @app.route("/register",methods=['POST','GET'])def register():
if request.method=="POST":
rollno = request.form['rollno']
      username = request.form['username']
      email = request.form['email']
      password = request.form['password']
insertTableData(conn, rollno, username, email, password)
return render_template('loginpage.html')
return render_template('registerpage.html')
if_name_=="_main_":
      app.run(debug=True)
      qlate,request,url_for,sessionapp=Flask(_name_)
```

```
@app.route("/&quot;)

@app.route("/login",methods=['POST','GET'])

def login():

    if request.method=="POST":

printTableData(conn)


username=request.form['username']

password=request.form['password']

try:

if dictionary[username] == password and username in dictionary:

     return "Logged in successfully"

except:

     return "Invalidusername or password"

return render_template('log)
```