

```
from flask import Flask,render_template,request,redirect,session,make_response,url_for
import sqlite3 as sql
from functools import wraps
import re
```

```
import ibm_db
app=Flask(__name__)
app.secret_key = 'jackiechan'
```

```
hostname = "815fa4db-dc03-4c70-869a-
a9cc13f33084.bs2io90l08kqb1od8lcg.databases.appdomain.cloud"
```

```
uid = "tkx67682"
```

```
pwd = "o7CLe1RKTEB89jC8"
```

```
driver = "{IBM DB2 ODBC DRIVER}"
```

```
db = "bludb"
```

```
port = "30367"
```

```
protocol = "TCPIP"
```

```
cert = "DigiCertGlobalRootCA.crt"
```

```
dsn = (
```

```
    "DATABASE={0};"
```

```
    "HOSTNAME={1};"
```

```
    "PORT={2};"
```

```
    "UID={3};"
```

```
    "SECURITY=SSL;"
```

```
    "SSLServerCertificate={4};"
```

```
    "PWD={5};"
```

```
).format(db, hostname, port, uid, cert, pwd)
```

```
print(dsn)
```

```
conn = ibm_db.connect(dsn, "", "")
```

```
email = 'harish19gmail.com'
```

```
sql = "SELECT * FROM USERS WHERE email = ?"
```

```
stmt = ibm_db.prepare(conn, sql)
```

```
ibm_db.bind_param(stmt, 1, email)
```

```
ibm_db.execute(stmt)
```

```
account = ibm_db.fetch_assoc(stmt)
```

```
print(account)
```

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

```
def root():
```

```
    return render_template("login.html")
```

```
@app.route('/signup', methods=['POST', 'GET'])
```

```
def signup():
```

```
    mg = "
```

```
    if request.method == "POST":
```

```
        username = request.form['username']
```

```
        email = request.form['email']
```

```
        pw = request.form['password']
```

```
        sql = 'SELECT * FROM USERS WHERE email =?'
```

```
        stmt = ibm_db.prepare(conn, sql)
```

```
        ibm_db.bind_param(stmt, 1, email)
```

```
        ibm_db.execute(stmt)
```

```
        acnt = ibm_db.fetch_assoc(stmt)
```

```
        print(acnt)
```

```
    if acnt:
```

```

        mg = 'Account already exists!!'
elif not re.match(r'^[@]+\.[^@]+', email):
    mg = 'Please enter the avalid email address'
elif not re.match(r'[A-Za-z0-9]+', username):
    mg = 'name must contain only character and number'
else:
    insert_sql = 'INSERT INTO USERS (USERNAME,EMAIL,PASSWORD) VALUES (?,?,?)'
    pstmt = ibm_db.prepare(conn, insert_sql)
    ibm_db.bind_param(pstmt, 1, username)
    # ibm_db.bind_param(pstmt,4,"123456789")
    ibm_db.bind_param(pstmt, 2, email)
    ibm_db.bind_param(pstmt, 3, pw)
    print(pstmt)
    ibm_db.execute(pstmt)
    mg = 'You have successfully registered click login!'
    return render_template("login.html", meg=mg)

```

```

return render_template("signup.html", meg=mg)

```

```

def rewrite(url):
    view_func, view_args = app.create_url_adapter(request).match(url)
    return app.view_functions[view_func](**view_args)

```

```

def login_required(f):
    @wraps(f)
    def decorated_function(*args, **kwargs):
        if "id" not in session:
            return redirect(url_for('login'))
        return f(*args, **kwargs)
    return decorated_function

```

```
@app.route('/dashboard', methods=['POST', 'GET'])
@login_required
def dashBoard():
    headings = ("id", "name", "order_id", "location")
    data = (
        ("1", "lorem", "ipsum", "dolor"),
        ("2", "lorem", "ipsum", "dolor"),
        ("3", "lorem", "ipsum", "dolor"),
        ("1", "lorem", "ipsum", "dolor"),
        ("2", "lorem", "ipsum", "dolor"),
        ("3", "lorem", "ipsum", "dolor"),
    )
    return render_template("dashboard.html", headings=headings, data=data)
```

```
@app.route('/orders', methods=['POST', 'GET'])
@login_required
def orders():
    return render_template("orders.html")
```

```
@app.route('/suppliers', methods=['POST', 'GET'])
@login_required
def suppliers():
    return render_template("suppliers.html")
```

```
@app.route('/profile', methods=['POST', 'GET'])
@login_required
```

```
def profile():  
    return render_template("profile.html")
```

```
@app.route('/logout', methods=['GET'])
```

```
@login_required
```

```
def logout():  
    print(request)  
    resp = make_response(render_template("login.html"))  
    session.clear()  
    return resp
```

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

```
def login():
```

```
    global userid
```

```
    msg = "
```

```
    if request.method == 'POST':
```

```
        un = request.form['username']
```

```
        pd = request.form['password_1']
```

```
        print(un, pd)
```

```
        sql = "SELECT * FROM USERS WHERE email =? AND password=?"
```

```
        stmt = ibm_db.prepare(conn, sql)
```

```
        ibm_db.bind_param(stmt, 1, un)
```

```
        ibm_db.bind_param(stmt, 2, pd)
```

```
        ibm_db.execute(stmt)
```

```
        account = ibm_db.fetch_assoc(stmt)
```

```
        print(account)
```

```

if account:

    session['loggedin'] = True

    session['id'] = account['EMAIL']

    userid = account['EMAIL']

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

    msg = 'Logged in successfully !'


    return rewrite('/dashboard')
else:

    msg = 'Incorrect username / password !'

    return render_template('login.html', msg=msg)
else:

    return render_template('login.html')

```

```

@app.route('/user/<id>')
@login_required
def user_info(id):

    with sql.connect('inventorymanagement.db') as con:

        con.row_factory = sql.Row

        cur = con.cursor()

        cur.execute(f'SELECT * FROM USERS WHERE email="{id}"')

        user = cur.fetchall()

    return render_template("user_info.html", user=user[0])


if __name__ == '__main__':

    app.run(debug=True)

```