

SPRINT 1

TEAM ID	PNT2022TMID22144
PROJECT NAME	Skill Based Job Recommender Application

LOGIN PANEL

CODE:

```
1  import ibm_db
2  from flask import Flask, render_template, request, redirect, url_for, session
3  import os
4  import bcrypt
5
6  # os.add_dll_directory('~app-packs/lib/python3.8/site-packages/clidriver/bin')
7
8
9  conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=ba99a9e6-d59e-4883-8fc0-d6a8c9f7a08f.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=313
10
11
12  # url_for('static', filename='style.css')
13
14  app = Flask(__name__)
15  app.secret_key = 'job-key-pk'
16
17
18  @app.route("/", methods=['GET'])
19  def home():
20      if 'email' not in session:
21          return redirect(url_for('login'))
22      return render_template('home.html', name='Home')
23
24  @app.route("/register", methods=['GET', 'POST'])
25  def register():
26      if request.method == 'POST':
27          email = request.form['email']
28          username = request.form['username']
29          password = request.form['password']
30
31          if not email or not username or not password:
```

```
30
31      if not email or not username or not password:
32          return render_template('register.html', error='Please fill all fields')
33
34      hash = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt())
35
36      query = "SELECT * FROM USER WHERE email=? OR rollNo=?"
37      stmt = ibm_db.prepare(conn, query)
38      ibm_db.bind_param(stmt, 1, email)
39      ibm_db.execute(stmt)
40      isUser = ibm_db.fetch_assoc(stmt)
41
42      if not isUser:
43          insert_sql = "INSERT INTO User(username,email,PASSWORD,rollNo) VALUES (?, ?, ?, ?)"
44          prep_stmt = ibm_db.prepare(conn, insert_sql)
45          ibm_db.bind_param(prepare_stmt, 1, username)
46          ibm_db.bind_param(prepare_stmt, 2, email)
47          ibm_db.bind_param(prepare_stmt, 3, hash)
48          ibm_db.execute(prepare_stmt)
49          return render_template('register.html', success="You can login")
50      else:
51          return render_template('register.html', error='Invalid Credentials')
52
53      return render_template('register.html', name='Home')
54
55  @app.route("/login", methods=['GET', 'POST'])
56  def login():
57      if request.method == 'POST':
58          email = request.form['email']
59          password = request.form['password']
```

```

58     email = request.form['email']
59     password = request.form['password']
60
61     if not email or not password:
62         return render_template('login.html',error='Please fill all fields')
63     query = "SELECT * FROM USER WHERE email=?"
64     stmt = ibm_db.prepare(conn, query)
65     ibm_db.bind_param(stmt,1,email)
66     ibm_db.execute(stmt)
67     isUser = ibm_db.fetch_assoc(stmt)
68     print(isUser,password)
69
70     if not isUser:
71         return render_template('login.html',error='Invalid Credentials')
72
73     isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))
74
75     if not isPasswordMatch:
76         return render_template('login.html',error='Invalid Credentials')
77
78     session['email'] = isUser['EMAIL']
79     return redirect(url_for('home'))
80
81     return render_template('login.html',name='Home')
82
83
84 @app.route('/logout')
85 def logout():
86     session.pop('email', None)
87     return redirect(url_for('login'))

```