| | |
|---|---|
| **TEAM ID** | PNT2022TMID11721 |
| **PROJECT NAME** | Smart Fashion Recommender Application |

```
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import bcrypt
conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-73d1-4859-88f0-
a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;PORT=30756;SECURITY=SSL;
SSLServerCertificateDigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=nhl80748;PWD=3yD0G9e6VuQHsO
BX;", "", "")
#url_for('static', filename='style.css')



app = Flask(__name__)
app.secret_key = b'_5#y2L"F4Q8z\n\xec]/'


@app.route("/",methods=['GET'])
def home():
    if 'email' not in session:
      return redirect(url_for('index'))
    return render_template('home.html',name='Home')
@app.route("/index")
def index():
  return render_template('index.html')


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

```
def register():
  if request.method == 'POST':
    username = request.form['username']
    email = request.form['email']
    phoneno = request.form['phoneno']
    password = request.form['password']

    if not username or not email or not phoneno or not password:
      return render_template('register.html',error='Please fill all fields')
    hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())
    query = "SELECT * FROM user_detail WHERE email=? OR phoneno=?"
    stmt = ibm_db.prepare(conn, query)
    ibm_db.bind_param(stmt,1,email)
    ibm_db.bind_param(stmt,2,phoneno)
    ibm_db.execute(stmt)
    isUser = ibm_db.fetch_assoc(stmt)
    if not isUser:
      insert_sql = "INSERT INTO user_detail(username, email, phoneno, password) VALUES (?,?,?,?)"
      prep_stmt = ibm_db.prepare(conn, insert_sql)
      ibm_db.bind_param(prep_stmt, 1, username)
      ibm_db.bind_param(prep_stmt, 2, email)
      ibm_db.bind_param(prep_stmt, 3, phoneno)
      ibm_db.bind_param(prep_stmt, 4, hash)
      ibm_db.execute(prep_stmt)
      return render_template('register.html',success="You can login")
    else:
      return render_template('register.html',error='Invalid Credentials')

  return render_template('register.html',name='Home')

@app.route("/login",methods=['GET','POST'])
def login():
    if request.method == 'POST':
      email = request.form['email']
      password = request.form['password']

      if not email or not password:
        return render_template('login.html',error='Please fill all fields')
      query = "SELECT * FROM user_detail WHERE email=?"
      stmt = ibm_db.prepare(conn, query)
      ibm_db.bind_param(stmt,1,email)
      ibm_db.execute(stmt)
      isUser = ibm_db.fetch_assoc(stmt)
      print(isUser,password)

      if not isUser:
        return render_template('login.html',error='Invalid Credentials')

      isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))
```

```
   if not isPasswordMatch:
    return render_template('login.html',error='Invalid Credentials')

   session['email'] = isUser['EMAIL']
   return redirect(url_for('home'))

  return render_template('login.html',name='Home')

@app.route("/admin",methods=['GET','POST'])
def adregister():
 if request.method == 'POST':
  username = request.form['username']
  email = request.form['email']
  phoneno = request.form['phoneno']
  password = request.form['password']

  if not username or not email or not phoneno or not password:
   return render_template('adminregister.html',error='Please fill all fields')
  hash=bcrypt.hashpw(password.encode('utf-8'),bcrypt.gensalt())
  query = "SELECT * FROM admin_detail WHERE email=? OR phoneno=?"
  stmt = ibm_db.prepare(conn, query)
  ibm_db.bind_param(stmt,1,email)
  ibm_db.bind_param(stmt,2,phoneno)
  ibm_db.execute(stmt)
  isUser = ibm_db.fetch_assoc(stmt)
  if not isUser:
   insert_sql = "INSERT INTO admin_detail(username, email, phoneno, password) VALUES (?,?,?,?)"
   prep_stmt = ibm_db.prepare(conn, insert_sql)
   ibm_db.bind_param(prep_stmt, 1, username)
   ibm_db.bind_param(prep_stmt, 2, email)
   ibm_db.bind_param(prep_stmt, 3, phoneno)
   ibm_db.bind_param(prep_stmt, 4, hash)
   ibm_db.execute(prep_stmt)
   return render_template('adminregister.html',success="You can login")
  else:
   return render_template('adminregister.html',error='Invalid Credentials')

 return render_template('adminregister.html',name='Home')

@app.route("/adminlogin",methods=['GET','POST'])
def adlogin():
  if request.method == 'POST':
   email = request.form['email']
   password = request.form['password']

   if not email or not password:
    return render_template('adminlogin.html',error='Please fill all fields')
   query = "SELECT * FROM admin_detail WHERE email=?"
   stmt = ibm_db.prepare(conn, query)
   ibm_db.bind_param(stmt,1,email)
   ibm_db.execute(stmt)
   isUser = ibm_db.fetch_assoc(stmt)
   print(isUser,password)

   if not isUser:
    return render_template('adminlogin.html',error='Invalid Credentials')

   isPasswordMatch = bcrypt.checkpw(password.encode('utf-8'),isUser['PASSWORD'].encode('utf-8'))

   if not isPasswordMatch:
    return render_template('adminlogin.html',error='Invalid Credentials')

   session['email'] = isUser['EMAIL']
   return redirect(url_for('home'))

  return render_template('adminlogin.html',name='Home')

@app.route('/logout')
def logout():
  session.pop('email', None)
  return redirect(url_for('login'))
if __name__ == "__main__":
  app.run(debug=True)
```