

SPRINT - 1

Date	19 November 2022
Team ID	PNT2022TMID15812
Project Name	Project- Smart fashion Recommendation system
Story Points	20

1. Functional Requirements: Registration.

User story: USN-1.

Solution:

```
from turtle import st
from flask import Flask, render_template, request, redirect, url_for, session
from markupsafe import escape

import ibm_db
conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=824dfd4d-99de-440d-9991-629c01b3832d.bs2io90l08kqb1od8lcg.databases.appdomain.cloud;PORT=30119;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=xhx40038;PWD=BDz5ow7439yj5PEd",'','')
print ("Database connection established", conn)

app = Flask(__name__)

@app.route('/')
def home():
    return render_template('home.html')

@app.route('/addstudent')
def new_student():
```

```

    return render_template('add_student.html')
@app.route('/list')
def list():
    return render_template('list.html')

@app.route('/addrec', methods = ['POST', 'GET'])
def addrec():
    if request.method == 'POST':

        name = request.form['name']
        email = request.form['email']
        password = request.form['password']

        sql = "SELECT * FROM userdata WHERE name=? "
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)

        if account:
            return render_template('list.html', msg="You are already a user, please login using your details")
        else:
            insert_sql = "INSERT INTO userdata VALUES (?, ?, ?)"
            prep_stmt = ibm_db.prepare(conn, insert_sql)
            ibm_db.bind_param(prepare_stmt, 1, name)
            ibm_db.bind_param(prepare_stmt, 2, email)
            ibm_db.bind_param(prepare_stmt, 3, password)

            ibm_db.execute(prepare_stmt)

            return render_template('home.html', msg="Registered successfully")

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

```

```

def check():

    if request.method == 'POST':

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

        sql = "SELECT * FROM userdata WHERE email=? and password= ?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,email)
        ibm_db.bind_param(stmt,2,password)

        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            return render_template('result.html', msg="")
        else:
            return render_template('list.html', msg="Please check your credentials!")

# # while student != False:
# #     print ("The Name is : ", student)

# print(student)

# @app.route('/posts/edit/<int:id>', methods=['GET', 'POST'])
# def edit(id):

#     post = BlogPost.query.get_or_404(id)

#     if request.method == 'POST':
#         post.title = request.form['title']

```

```
#         post.author = request.form['author']
#         post.content = request.form['content']
#         db.session.commit()
#         return redirect('/posts')
#     else:
#         return render_template('edit.html', post=post)
```

2. Functional Requirements: Verification.

User story: USN-2.

[Solution:](#)

```
CREATE TABLE userdata (
    name VARCHAR(250) NOT NULL,
    email VARCHAR(100) NOT NULL,
    password CHAR(200) NOT NULL);
```

3. Functional Requirements: login Process.

User story: USN-3.

[Solution:](#)

Login

Register

Username :

Email Address:

Password :

Register

e-CART



