

ASSIGNMENT-2

Project Name: Smart Fashion
Recommender Application

Team ID : PNT2022TMID29080

Roll Number:421619104025

Student Name: Hariharasudhan A

Solution:

HTML File

```
<html>
```

```
  <head>
```

```
    <meta charset="utf-8">
```

```
    <meta name="viewport" content="width=device-width, initial-scale=1,  
    shrink-to-fit=no">
```

```
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-  
    awesome/4.7.0/css/font-awesome.min.css">
```

```
    <link rel="stylesheet"  
    href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.0/css/bootstrap.min  
.css">
```

```
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/particlesjs/2.2.3/particles.min.js"
>

</head>

<style>
    body{
        background:linear-gradient(to right,#c9522d,#f64c72);
        background-attachment: fixed;
        background-repeat: no-repeat;
        background-size: cover; margin:
        0;
        padding: 0;
    }

    .form-control, .form-control:focus ,.input-group-
    addon{border-color: aliceblue;
    border-radius: 0;
    }

    .signup{
        width: 360px;
        margin: 0 auto;
        padding: 50px 0;
    }
```

```
.signup h2{
    color: #fd614d;
    text-align: center;
    margin: 0 0 20px;
}

.signup form{
    border-radius: 1px;
    margin-bottom: 15px;

    background-color: rgba(0,0,0,0.7) !important;
    border: 1px solid none;
    box-shadow: 0px 2px 2px rgba(0,0,0,0.3);
    padding: 30px;
}

.signup .form-
    group{ margin-
        bottom: 30px;
    }

.signup .form-
    control{min-height:
        38px;
        box-shadow: none !important;
        border-width: 0 0 1px 0;
    }
```

```
.form-control{
    background-color: transparent;
    color: #fd614d;
}

.form-control:hover {
    background-color: transparent;
    color: aliceblue;
    overflow: hidden;
    text-decoration: none;
}

.signup .input-group-
    addon{max-width: 42px;
    text-align: center;
    background: none;
    border-bottom: 1px solid aliceblue;
    padding-left: 5px;
}

.input-group-
    addon{font-size:
    30px;
}

.btn{
    background: linear-gradient(259deg,#f64,#f64c72);
    border: none;
```

```
    text-align: center;
    padding: 3%;
    font-size: 16px;
    font-weight: bold;
    color: aliceblue;
}
.btn:hover{
    background: aliceblue;
border: none;
color: aliceblue;
color: #fd614d;
}
.fa{
    color: aliceblue;
}
.signup .fa-
    envelope{font-size:
    23px; color:
    aliceblue;
}
a{
    color: aliceblue;
    text-decoration: none;
}
```

```

a:hover{
    color:#664a71;
    text-decoration: none;
}
</style>

<body>
    <div class="signup">
        <form action="{ {url_for(result)} }" method="post">
            <h2>LOGIN</h2>
            <div class="form-group">
                <div class="input-group">
                    <span class="input-group-addon"><i class="fa fa-envelope"></i></span>
                    <input type="email" name="email" class=" form-control"placeholder="Email" required>
                </div>
            </div>
            <div class="form-group">
                <div class="input-group">
                    <span class="input-group-addon"><i class="fa fa-lock"></i></span>
                    <input type="password" name="password" class=" form-control"placeholder="password" required>
                </div>
            </div>
        </form>
    </div>

```

```

        </div>

        <div class="form-group">

            <button type="submit" class="btn btn-primary btn-lg btn-
block">Login</button>

        </div>

    </form>

</div>

</body>

</html>

```

Database connectivity:

```

import ibm_db def
display(conn):

    sql="SELECT * FROM USER"

    stmt = ibm_db.exec_immediate(conn,sql)

    dictionary = ibm_db.fetch_both(stmt) while
dictionary != False:

        print ("The roll no is : ", dictionary["ROLLNO"]) print
        ("The Name is : ", dictionary["USERNAME"])print
        ("The Email no is : ", dictionary["EMAIL"])

        print ("The Password is : ", dictionary["PASSWORD"],"\n")

        dictionary = ibm_db.fetch_both(stmt)

```

```
def Insert(conn,rollno,name,email,password):

    sql="INSERT                                INTO                                USER
VALUES('{}','{}','{}','{}').format(rollno,name,email,password)

    stmt=ibm_db.exec_immediate(conn,sql)

    print("The Number if affected rows: ", ibm_db.num_rows(stmt),"\\n")
```

```
def Update(conn,rollno,name,email,password):

    sql="UPDATE    Users    SET            ROLLNO    ='{}',            WHERE
USERNAME='{};'.format(rollno,name)

    stmt=ibm_db.exec_immediate(conn,sql)

    print("The Number if affected rows: ", ibm_db.num_rows(stmt),"\\n")
```

try:

```
    conn=ibm_db.connect("DATABASE=bludb;HOSTNAME=2f3279a5-
73d1-4859-88f0-
a6c3e6b4b907.c3n41cmd0nqnrk39u98g.databases.appdomain.cloud;POR
T=30756;SECURITY=SSL;
SSLServerCertificateDigiCertGlobalRootCA.crt;PROTOCOL=TCPIP;UID=n
hl80748;PWD=3yD0G9e6VuQHsOBX;"," ", "")
```

```
    print("connection Success")
```

```
    display(conn)
```

```
Insert(conn,"621319205009","Gopika","gopikaramalingam09@gmail.com","
euyfuis")
```

```
    display(conn)
```


except:

```
    print("couldn't connect")
```

Python File:

```
import sys
```

```
sys.path.append('/workspace/python_db/dbconn.py')from
```

```
urllib import request
```

```
from flask import Flask,render_template,url_for
```

```
app = Flask(__name__)
```

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

```
def homepage():
```

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

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

```
send_data():
```

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

```
    #dbconn.Insert_values()
```

```
    return "hello"
```

```
if __name__=="__main__":
```

```
    app.run(debug=True)
```

Output:

