

Date	29-October 2022
Team Id	PNT2022TMID45035
Project Name	Inventory Management System for Retailers
Batch No	B1-1M3E

PROJECT DEVELOPMENT PHASE-SPRINT 1

Login.html

```
<!{ % extends 'base.html' % }  
  
{ % block head % }  
    <title>Login</title>  
  
{ % endblock % }  
  
{ % block body % }  
  
{{ msg }}  
  
    <style>  
        .divider:after,  
.divider:before {      content:  
        "" ;      flex:
```

```
1;    height: 1px;    background:
```

```
#eee;
```

```
}
```

```
.h-custom {    height:
```

```
calc(100% - 73px);
```

```
}
```

```
@media (max-width: 450px) {
```

```
.h-custom {    height:
```

```
100%;
```

```
}
```

```
}
```

```
.hlink{    text-decoration:
```

```
none;
```

```
}
```

```
</style>
```

```
<section class="vh-100">
```

```
<div class="container-fluid h-custom">
```

```
<div class="row d-flex justify-content-center align-items-center h-100">
```

```
<div class="col-md-9 col-lg-6 col-xl-5">
```

```


</div>

<div class="col-md-8 col-lg-6 col-xl-4 offset-xl-1">

  <form class="mx-1 mx-md-4" action="{{ url_for('login') }}" method="post">

    <div class="d-flex align-items-center mb-3 pb-1">

      <span class="h1 fw-bold mb-0">Login</span>

    </div>

    <!-- Email input -->

    <div class="form-outline mb-4">

      <input type="email" name="email" value="" placeholder="Email ID" class="form-control" />

    </div>

    <!-- Password input -->

    <div class="form-outline mb-3">

      <input type="password" name="password" value="" placeholder="Password" class="form-control" />

    </div>

    <div class="text-center text-lg-start mt-4 pt-2">

      <input type="submit" class="btn btn-primary btn-lg" style="width: 125px; display: block; margin-left: auto; margin-right:
auto;">

    </div>
```

```
<p class="support-p" style="margin-top: 10px;">Don't have an Account?  
    <a href="/register" style="color: rgb(0, 0, 0); text-decoration: none;"><strong style="font-size:  
large;">Register</strong></a>  
  
    </p>  
  
    </form>  
  
    </div>  
  
    </div>  
  
    </div>  
  
    </section>  
  
    {% endblock %}
```

mystyle.css

```
.right {    display:  
block;    float:  
right;    color:  
blue;  
}  
  
.bg {    background-image: url('pexels-francesco-  
ungaro281260.jpg');    background-position: center;  
backgroundrepeat: no-repeat;    background-size: cover;
```

```
}
```

DASHBOARD Users.html

```
{% extends 'dashboard/base.html' %}
```

```
{% block head %}
```

```
<title>Dashboard</title>
```

```
<style>
```

```
.container
```

```
{    padding: 20px;
```

```
border: 2px 5px 5px 5px;
```

```
border-color: #ffffff;
```

```
border-width: 5px;    border-
```

```
style: solid;    textalign:
```

```
center;    borderradius: 15px;
```

```
boxshadow: 1px 1px 10px 5px
```

```
rgba(234, 234, 234, 0.844);
```

```
margin-bottom: 10px;
```

```
margin-top: 10px;    margin-
```

```
left: 10px;    margin-right:
```

```
15px;    background-color:
```

```
#ffffff;    transition: transform
1.0s ease }    .container:hover
{    transform: scale(1.05); }

</style>

{% endblock %}

{% block body %}

<div class="home-content">

    <div class="overview-boxes">

        <section class="catogories" >

            <div class="container-fluid"> <div class="row">

                <div class="col-lg-6">

                    <div class="row">

                        <div class="col-lg-6 col-md-6 col-12 p-1">

                            <div class="catagories_item">

                                <div class="container">

                                    <a href="/productlist" style="text-decoration: none; color: black">

                                        <div class="box-topic">Products List

                                        </div></a>

                                    <a href="/productlist" style="text-decoration: none; color: black">

                                        
```

```
</a>
</div>
</div>
</div>
<div class="col-lg-6 col-md-6 col-12 p-1">
  <div class="catagories_item">
    <div class="container">
      <a href="/addproduct" style="text-decoration: none; color: black">
        <div class="box-topic">Add Products
      </div></a>
      <a href="/addproduct" style="text-decoration: none; color: black">
        
      </a>
    </div>
  </div>
</div>
</div>
</div>
<div class="col-lg-6">
  <div class="row">
```

```
<div class="col-lg-6 col-md-6 col-12 p-1">
  <div class="catagories_item">
    <div class="container">
      <a href="/movement" style="text-decoration: none; color: black">
        <div class="box-topic">Products Movement
      </div></a>
      <a href="/movement" style="text-decoration: none; color: black">
        </a>
      </div>
    </div>
  </div>
<div class="col-lg-6 col-md-6 col-12 p-1">
  <div class="catagories_item">
    <div class="container">
      <a href="/report" style="text-decoration: none; color: black">
        <div class="box-topic">Product Report
      </div></a>
      <a href="/report" style="text-decoration: none; color: black">
        
      </a>
    </div>
  </div>
</div>
```


$$\{\{\text{msg}\}\}$$

```
{ % endblock % }
```

special.html

```
{% extends 'dashboard/base.html' %}
```

```
{ % block head % }
```

```
<title>List of Users</title>

{% endblock %}

{% block body %}

<link href="//maxcdn.bootstrapcdn.com/bootstrap/3.3.0/css/bootstrap.min.css" rel="stylesheet" id="bootstrap-css">

<div > </div>

<div class="home-content">

<div class="overview-boxes">

<section class="catagories" >

<div class="container-fluid">

<div class="row">

<div class="col-lg-6">

<div class="row">

<div class="col-lg-6 col-md-6 col-12 p-1">

<div class="catagories_item">

</div>

</div>

<div class="col-lg-6 p-0" style=" margin-top: 25px;">

<div class="catagories_item">

<div class="catagories_item catagories_large_item" >

</div>
```

</div>

</div>

<div class="col-lg-6 col-md-6 col-12 p-1">

<div class="catagories_item">

</div>

</div>

<div class="col-lg-6 col-md-6 col-12 p-1">

<div class="catagories_item">

</div>

</div>

</div>

</div>

<div class="container register-form">

<div class="form">

<div class="d-flex align-items-center mb-3 pb-1">

{{msg}}

Product List

</div>

<table class="table table-hover">

```
<thead>

<tr>

<th>ID</th>

<th>PRODUCT NAME</th>

<th>QUANTITY</th>

<th>DATE</th>

</tr>

</thead>

{% for row in products %}

<tr>

<td>{{ row["ID"] }}</td>

<td>{{ row["PRODUCTNAME"] }}</td>

<td>{{ row["QUANTITY"] }}</td>

<td>{{ row["DATE"] }}</td>

</tr>

{% endfor %}

</table>

</div>

</div>

</div>
```

</div>

</div>

</div>

</section>

{% endblock % }

app.py

```
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=54a2f15b-5c0f-46df-8954-
7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32733;SECURITY=SSL;SSLServerCertificate=DigiCertGlo
balRootCA.crt;UID=lkc93724;PWD=zAzNGa6DaNk6xvle",",") import smtplib, ssl ## email.mime subclasses from
email.mime.multipart import MIMEMultipart from email.mime.text import MIMEText
```

```
## The pandas library is only for generating the current date, which is not necessary for sending emails import
```

```
pandas as pd app = Flask(__name__)
```

```
var_list = [] app.secret_key =
```

```
'your secret key'
```

```
@app.route('/') def home():
```

```
if not session.get("name"):
```

```
    return render_template('home.html') return
```

```
render_template('home.html', session = session)
```

```

@app.route('/register') def new_student():
    return render_template('Register.html')

@app.route('/addrec',methods = ['POST', 'GET']) def
addrec():    if request.method == 'POST':        fname =
request.form['fname']        lname = request.form['lname']
cname = request.form['cname']                state =
request.form['state']        city = request.form['city']
mobilenos = request.form['mobilenos']        emailid =
request.form['emailid']                password =
request.form['password']                pincode =
request.form['pincode']    sql = "SELECT * FROM Users
WHERE EMAILID=?"

    stmt = ibm_db.prepare(conn, sql)
    ibm_db.bind_param(stmt,1,emailid)
    ibm_db.execute(stmt)                account =
    ibm_db.fetch_assoc(stmt) if account:        users
= []    sql = "SELECT * FROM Users"    stmt
= ibm_db.exec_immediate(conn, sql)
    dictionary = ibm_db.fetch_both(stmt)    while
dictionary != False:

```

```

        # print ("The Name is : ", dictionary)      users.append(dictionary)      dictionary = ibm_db.fetch_both(stmt)
return render_template('list.html', msg="You are already a member, please login using your details", users = users)
else:
    var_list.append(fname)
var_list.append(lname)      var_list.append(cname)
var_list.append(state)      var_list.append(city)
var_list.append(mobilenos)      var_list.append(emailid)
var_list.append(password)
var_list.append(pincode)

    bodytemp = r'D:\IBM\GUIDED PROJECT\INVENTORY MANAGEMENT SYSTEM FOR RETAILERS\templates\email.html'
with open(bodytemp, "r", encoding='utf-8') as f:      html= f.read()

    # Set up the email addresses and password. Please replace below with your email address and password
email_from = 'padhu10a@gmail.com'      epassword = 'rbjibzkssszsbrjo'      email_to = emailid

    # Generate today's date to be included in the email Subject      date_str
= pd.Timestamp.today().strftime('%Y-%m-%d')

    # Create a MIMEMultipart class, and set up the From, To, Subject fields      email_message
= MIMEMultipart()      email_message['From'] = email_from      email_message['To'] =
email_to      email_message['Subject'] = f'Report email - {date_str}'

    # Attach the html doc defined earlier, as a MIMEText html content type to the MIME message
email_message.attach(MIMEText(html, "html"))

```

```

    # Convert it as a string    email_string = email_message.as_string() #
Connect to the Gmail SMTP server and Send Email    context =
ssl.create_default_context()    with smtplib.SMTP_SSL("smtp.gmail.com", 465,
context=context) as server:

    server.login(email_from, epassword)

server.sendmail(email_from, email_to, email_string)    return render_template('notify.html')

@app.route('/confirm') def confirmation():

    insert_sql = "INSERT INTO Users (FIRSTNAME, LASTNAME, COMPANYNAME, STATE, CITY, MOBILENO, EMAILID,
PASSWORD, PINCODE) VALUES (?, ?, ?, ?, ?, ?, ?, ?)"    prep_stmt = ibm_db.prepare(conn, insert_sql)
ibm_db.bind_param(prepare_stmt, 1, var_list[0])    ibm_db.bind_param(prepare_stmt, 2, var_list[1])    ibm_db.bind_param(prepare_stmt, 3,
var_list[2])    ibm_db.bind_param(prepare_stmt, 4, var_list[3])    ibm_db.bind_param(prepare_stmt, 5, var_list[4])
ibm_db.bind_param(prepare_stmt, 6, var_list[5])    ibm_db.bind_param(prepare_stmt, 7, var_list[6])    ibm_db.bind_param(prepare_stmt, 8,
var_list[7])    ibm_db.bind_param(prepare_stmt, 9, var_list[8])    ibm_db.execute(prepare_stmt)    return render_template('confirm.html')

@app.route('/login', methods =['POST', 'GET']) def login():    msg = "    if

request.method == 'POST' and 'email' in request.form and 'password' in request.form:

    email = request.form['email']

password = request.form['password']

    sql = "SELECT * FROM Users WHERE EMAILID =? 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:
    session['loggedin'] = True      session['id'] =
account['ID']      session['email'] = account['EMAILID']
session['name'] = account['FIRSTNAME']      msg =
'Logged in successfully !'      return
render_template('home.html', msg = msg)
else:
    msg = 'Incorrect email / password !'      return
render_template('login.html', msg = msg)
@app.route('/logout') def
logout():
    session.pop('loggedin', None)
session.pop('id', None)
session.pop('email', None)
session.pop('name', None)      return
redirect(url_for('home'))
@app.route('/list') def list():
users = []

```

```
sql = "SELECT * FROM Users" stmt =  
ibm_db.exec_immediate(conn, sql)  
dictionary = ibm_db.fetch_both(stmt) while  
dictionary != False: # print ("The Name is  
: ", dictionary) users.append(dictionary)  
dictionary = ibm_db.fetch_both(stmt) if  
users:  
    return render_template("list.html", users = users , session = session) return  
"No users..."
```