

SPRINT - 2

DATE	05 November 2022
Team ID	PNT2022TMID49588
Project Name	Project-Personal Expense Tracker

CODE:

app.py

```
from flask import Flask, redirect, url_for, render_template, request, flash, session
import ibm_db
import re
```

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

```
app.secret_key = 'a'
```

```
conn =
```

```
ibm_db.connect("DATABASE=bludb;HOSTNAME=6667d8e9-9d4d-4ccb-ba32-21da3bb5aaf
c.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=30376;SECURITY=SSL;SSLSe
rverCertificate=DigiCertGlobalRootCA.crt;UID=pht72647;PWD=IILWoOw2JKv8XKkj","")
```

```
#HOME--PAGE
```

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

```
def home():
```

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

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

```
def add():
```

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

```
#SIGN--UP--OR--REGISTER
```

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

```
def signup():
```

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

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

```

def register():
    msg = ""
    if request.method == 'POST' :
        username = request.form['username']
        email = request.form['email']
        password = request.form['password']
        stmt = ibm_db.prepare(conn,'SELECT * FROM register WHERE username = ?')
        ibm_db.bind_param(stmt,1,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        if account:
            msg = 'Account already exists !'
        elif not re.match(r'^@]+@^[^@]+\.[^@]+', email):
            msg = 'Invalid email address !'
        elif not re.match(r'[A-Za-z0-9]+', username):
            msg = 'name must contain only characters and numbers !'
        else:
            insert_sql = "INSERT INTO register VALUES (?, ?, ?)"
            stmt = ibm_db.prepare(conn,insert_sql)
            ibm_db.bind_param(stmt, 1, username)
            ibm_db.bind_param(stmt, 2, email)
            ibm_db.bind_param(stmt, 3, password)
            ibm_db.execute(stmt)
            msg = 'You have successfully registered !'
    return render_template('signup.html', msg = msg)

```

#LOGIN--PAGE

```

@app.route("/signin")
def signin():
    return render_template("login.html")

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

```

```

    if request.method == 'POST' :
        username = request.form['username']
        password = request.form['password']
        stmt = ibm_db.prepare(conn,'SELECT * FROM register WHERE username = ? AND
password = ?')
        ibm_db.bind_param(stmt,1,username)
        ibm_db.bind_param(stmt,2,password)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)
        print (account)

```

```

if account:
    session['loggedin'] = True
    session['username'] = account['USERNAME']
    session['password'] = account['PASSWORD']
    msg = 'Logged in successfully !'

    return redirect('/home')
else:
    msg = 'Incorrect username / password !'
return render_template('login.html', msg = msg)

```

#ADDING----DATA

```

@app.route("/add")
def adding():
    return render_template('add.html')

@app.route('/addexpense',methods=['GET', 'POST'])
def addexpense():

```

```

    date = request.form['date']
    expensename = request.form['expensename']
    amount = request.form['amount']
    paymode = request.form['paymode']
    category = request.form['category']

    insert_sql = "(INSERT INTO expenses VALUES (NULL, % s, % s, % s, % s, % s, % s),(session['id'] ,date, expensename, amount, paymode, category)"
    stmt = ibm_db.prepare( conn , insert_sql)
    ibm_db.bind_param(stmt, 1, session['id'])
    ibm_db.bind_param(stmt, 2, date)
    ibm_db.bind_param(stmt, 3, expensename)
    ibm_db.bind_param(stmt, 4, amount)
    ibm_db.bind_param(stmt, 5, paymode)
    ibm_db.bind_param(stmt, 6, category)
    ibm_db.execute(stmt)
    print(date + " " + expensename + " " + amount + " " + paymode + " " + category)

    return redirect("/display")

```

#DISPLAY---graph

```

@app.route("/display")
def display():

```

```

print(session["username"],session['id'])
stmt = ibm_db.prepare(conn,'SELECT * FROM expenses WHERE userid = % s AND date
ORDER BY `expenses`.`date` DESC',(str(session['id'])))
ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)

return render_template('display.html' , expense = expense)

```

#delete---the--data

```

@app.route('/delete/<string:id>', methods = ['POST', 'GET' ])
def delete(id):
    stmt = ibm_db.prepare(conn,'DELETE FROM expenses WHERE id = {0}'.format(id))
    ibm_db.execute(stmt)
    print('deleted successfully')
    return redirect("/display")

```

#UPDATE---DATA

```

@app.route('/edit/<id>', methods = ['POST', 'GET' ])
def edit(id):
    stmt = ibm_db.prepare(conn,'SELECT * FROM expenses WHERE id = %s', (id,))
    row = ibm_db.fetch_tuple()

    print(row[0])
    return render_template('edit.html', expenses = row[0])

```

```

@app.route('/update/<id>', methods = ['POST'])
def update(id):
    if request.method == 'POST' :

        date = request.form['date']
        expensename = request.form['expensename']
        amount = request.form['amount']
        paymode = request.form['paymode']
        category = request.form['category']

        stmt = ibm_db.prepare(conn,"UPDATE `expenses` SET `date` = % s , `expensename` =
% s , `amount` = % s , `paymode` = % s , `category` = % s WHERE `expenses`.`id` = % s
",(date, expensename, amount, str(paymode), str(category),id))
        ibm_db.execute(stmt)

```

```
print('successfully updated')
return redirect("/display")
```

```
#limit
@app.route("/limit" )
def limit():
    return redirect('/limitn')

@app.route("/limitnum" , methods = ['POST' ])
def limitnum():
    if request.method == "POST":
        number= request.form['number']
        stmt = ibm_db.prepare(conn,'INSERT INTO limits VALUES (NULL, % s, % s)
',(session['id'], number))
        ibm_db.execute(stmt)
        return redirect('/limitn')

@app.route("/limitn")
def limitn():
    stmt = ibm_db.prepare(conn,'SELECT limitss FROM `limits` ORDER BY `limits`.`id` DESC
LIMIT 1')
    x= ibm_db.fetch_assoc()
    s = x[0]

    return render_template("limit.html" , y= s)

#REPORT

#log-out

@app.route('/logout')

def logout():
    session.pop('loggedin', None)
    session.pop('id', None)
    session.pop('username', None)
    return render_template('home.html')
```

```
if __name__ == "__main__":  
    app.run(debug=True)
```

base.html

```
<!DOCTYPE html>  
<html lang="en">  
<head>  
    <link rel="stylesheet"  
href="https://cdn.jsdelivr.net/npm/bootstrap@4.5.3/dist/css/bootstrap.min.css"  
integrity="sha384-TX8t27EcRE3e/ihU7zmQxVncDAy5uIKz4rEkgIXeMed4M0jflDPvg6uqKI2  
xXr2" crossorigin="anonymous">  
    <script src="https://cdn.jsdelivr.net/npm/chart.js@3.2.1/dist/chart.min.js"></script>  
    <script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"  
integrity="sha384-DfXdz2htPH0lsSSs5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+Or  
CXaRkfj" crossorigin="anonymous"></script>  
  
    <script src="https://cdn.jsdelivr.net/npm/popper.js@1.16.1/dist/umd/popper.min.js"  
integrity="sha384-9/reFTGAW83EW2RDu2S0VVKalzap3H66lZH81PoYlFhbGU+6BZp6G7niu  
735Sk7lN" crossorigin="anonymous"></script>  
  
    <script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"  
integrity="sha384-B4gt1jrGC7Jh4AgTPSdUtOBvfO8shuf57BaghqFfPIYxofvL8/KUEfYiJOMM  
V+rV" crossorigin="anonymous"></script>  
  
    <link rel="stylesheet" href="https://pro.fontawesome.com/releases/v5.10.0/css/all.css"  
integrity="sha384-AYmEC3Yw5cVb3ZcuHtOA93w35dYTsvhLPVnYs9eStHfGJvOvKxVfELGr  
oGkvsg+p" crossorigin="anonymous"/>  
  
    <title>Document</title>  
</head>  
<body>  
    <nav class="navbar sticky-top navbar-expand-lg navbar-light" style="background-color:  
#e3f2fd;">  
        <a class="navbar-brand" href="#">PET</a>  
        <button class="navbar-toggler" type="button" data-toggle="collapse"  
data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent"  
aria-expanded="false" aria-label="Toggle navigation">  
            <span class="navbar-toggler-icon"></span>  
        </button>  
  
    <div class="collapse navbar-collapse" id="navbarSupportedContent">  
        <ul class="navbar-nav mr-auto">  
            <li class="nav-item active">
```

```

        <a class="nav-link" href="/home">Home <span class="sr-only">(current)</span></a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/add">Add</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/display">History</a>
    </li>
    <li class="nav-item">
        <a class="nav-link" href="/limit">LIMIT</a>
    </li>
    <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" id="navbarDropdown" role="button"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">
            Report
        </a>
        <div class="dropdown-menu" aria-labelledby="navbarDropdown">
            <a class="dropdown-item" href="/today">TODAY</a>
            <a class="dropdown-item" href="/month">Month</a>

            <a class="dropdown-item" href="/year">Year</a>
        </div>
    </li>
</ul>
<div class="form-inline my-5 my-lg-1" style="position: absolute; left: 1250px;">
<ul class="navbar-nav mr-auto " > <li class="nav-item dropdown btn-group open">
    <a class="btn btn-primary" href="#"><i class="fa fa-user fa-fw"></i>User </a>
    <a class="btn btn-primary dropdown-toggle "href="#" id="navbarDropdown"
data-toggle="dropdown" aria-haspopup="true" aria-expanded="false">

</a>

    <div class="dropdown-menu" aria-labelledby="navbarDropdown">
        <a class="dropdown-item" href="#">Profile</a>
        <a class="dropdown-item" href="#">Settings</a>
        <a class="dropdown-item" href="#">Contact-Us</a>
        <div class="dropdown-divider"></div>
        <a class="dropdown-item" style="color: darkred;" href="/logout">Log-Out <i class="fa
fa-sign-out" aria-hidden="true"></i></a>
    </div>
</li>

<div ">
</ul>
</div>

</div>

```

```

</nav>
{% block body %}
{% endblock %}
</body>
</html>

```

add.html

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

```
{% block body %}
```

```

<div class="container">
  <div class="row">
    <div class="col-md-6">
      <h3>Add Expense</h3>
      <form action="/addexpense" method="POST">

        <div class="form-group">
          <label for="">Date</label>
          <input class="form-control" type="datetime-local" name="date"
id="date"></div>
          <div class="form-group"> <label for="">Expense name</label>
          <input class="form-control" type="text" name="expensename"
id="expensename">
          </div>
          <div class="form-group">
            <label for="">Expense Amount</label>
            <input class="form-control" type="number" min="0" name="amount"
id="amount">
            </div>

          <div class="form-group">
            <label for=""></label>
            <select class="form-control" name="paymode" id="paymode">
              <option selected hidden>Pay-Mode</option>
              <option name="cash" value="cash">cash</option>
              <option name="debitcard" value="debitcard">debitcard</option>
              <option name="creditcard" value="creditcard">creditcard</option>
              <option name="epayment" value="epayment">epayment</option>
              <option name="onlinebanking"
value="onlinebanking">onlinebanking</option>

            </select>

          <div class="form-group">

```



```

<label for=""></label>
<select class="form-control" name="category" id="category">

    <option selected hidden>Category</option>
    <option name = "food" value="food">food</option>
    <option name = "entertainment"
value="entertainment">Entertainment</option>
    <option name = "business" value="business">Business</option>
    <option name = "rent" value="rent">Rent</option>
    <option name = "EMI" value="EMI">EMI</option>
    <option name = "other" value="other">other</option>

</select>
</div>

<input class="btn btn-danger" type="submit" value="Add" id="">

</form>

<div style="position: relative; left: 590px; top: -460px;" class="image">
    

</div>

</div>
</div>

</div>

{% endblock %}

```

homepage.html

```

{% extends 'base.html' %}

{% block body %}
<style>
    H1 {
        position: relative;
        right: -790PX;
        top: -400PX;
        color: RED;
    }

```

```

p {
    position: relative;
    right: -800px;
    top: -350px;
    font-family: monospace;
}

span {
    position: relative;
    right: -800px;
    top: -360px;
}

.ccc {
    position: relative;
    top: 80px;
    left: -100px;
}
</style>
<div id=aa class="container">
    <div class="ccc">
        
        <h1>LET START JOURNEY</h1>
        <P>MyBudget web application helps<br> you to maintain budget<br>
            and analyse the expense</P>
        
    </div>
    <span class="btn btn-outline-dark">Let's Begin</span>
</div>

{% endblock %}

```

OUTPUT:



