

**Sprint 2**  
**PROJECT\_DEVELOPMENT\_PHASE**

Date	05 November 2022
Team ID	PNT2022TMID49592
Project Name	Project - Personal expense tracker application

**CODE:**

**app.py(updated):**

```
from flask import Flask, render_template, request, redirect, url_for, session
import ibm_db
import re
from flask import *
from flask import Flask
from random import *
from flask_mail import Mail, Message
app = Flask(__name__)
mail = Mail(app) # instantiate the mail class

# configuration of mail
app.config['MAIL_SERVER']='smtp.gmail.com'
app.config['MAIL_PORT'] = 465
app.config['MAIL_USERNAME'] = 'shrikumar13102001@gmail.com'
app.config['MAIL_PASSWORD'] = 'mlzuhzieegwenrhc'
app.config['MAIL_USE_TLS'] = False
app.config['MAIL_USE_SSL'] = True
mail = Mail(app)
otp = randint(000000,999999)
app.secret_key = 'a'
conn =
ibm_db.connect("DATABASE=bludb;HOSTNAME=b1bc1829-6f45-4cd4-bef4-10cf081900bf.c1o
gj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=32304;SECURITY=SSL;SSLServerCerti
ficate=DigiCertGlobalRootCA.crt;UID=jpg92341;PWD=5HWT2ImD7POkuLnv","")
@app.route("/")
def home():
    return render_template('index.html')
@app.route('/login', methods =['GET', 'POST'])
def login():
    global userid
    msg = ""
    if request.method == 'POST' and 'email' in request.form and 'password' in request.form:
        email = request.form['email']
```

```

password = request.form['password']
stmt = ibm_db.prepare(conn,'SELECT * FROM register WHERE email = ? AND password
= ?')
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['email'] = account['EMAIL']
    msg = 'Logged in successfully !'
    return render_template('add.html', msg = msg)
else:
    msg = 'Incorrect username / password !'
return render_template('login.html', msg = msg)

```

```

@app.route('/wallet')
def wallet():
    return render_template('design.html')
@app.route('/income', methods=['GET', 'POST'])
def income():
    if request.method == 'POST':
        amount = request.form['amount']
        income = request.form['income']
        expense = request.form['expense']
        category = request.form['category']
        insertSQL = "INSERT INTO addexpense VALUES (?, ?, ?, ?)"
        stmt = ibm_db.prepare(conn, insertSQL)
        ibm_db.bind_param(stmt, 1, amount)
        ibm_db.bind_param(stmt, 2, income)
        ibm_db.bind_param(stmt, 3, expense)
        ibm_db.bind_param(stmt, 4, category)
        ibm_db.execute(stmt)
    return render_template('add.html')

```

```

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

```

```

birthdate = request.form['birthdate']
phonenumber = request.form['phonenumber']
job = request.form['job']
income = request.form['income']
sql = "SELECT * FROM register WHERE email = ? "
stmt = ibm_db.prepare(conn,sql)
ibm_db.bind_param(stmt,1,email)
ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)
print(account)
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]+', firstname):
    msg = 'firstname must contain only characters and numbers !'
elif not email or not firstname or not password:
    msg = 'Please fill out the form !'
else:
    insert_sql = "INSERT INTO register VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?)"
    stmt = ibm_db.prepare(conn,insert_sql)
    ibm_db.bind_param(stmt, 1, firstname)
    ibm_db.bind_param(stmt, 2, lastname)
    ibm_db.bind_param(stmt, 3, email)
    ibm_db.bind_param(stmt, 4, password)
    ibm_db.bind_param(stmt, 5, cpassword)
    ibm_db.bind_param(stmt, 6, birthdate)
    ibm_db.bind_param(stmt, 7, phonenumber)
    ibm_db.bind_param(stmt, 8, job)
    ibm_db.bind_param(stmt, 9, income)
    ibm_db.execute(stmt)
    msg = 'You have successfully registered !'
    return render_template('email.html')
elif request.method == 'POST':
    msg = 'Please fill out the form !'

return render_template('register.html', msg = msg)
@app.route('/verify', methods=['GET', 'POST'])
def verify():
    if request.method == 'POST':
        email1 = request.form['email1']
        sql = "SELECT * FROM validate WHERE email1 = ? "
        stmt = ibm_db.prepare(conn,sql)
        ibm_db.bind_param(stmt,1,email1)

```

```

ibm_db.execute(stmt)
account = ibm_db.fetch_assoc(stmt)
print(account)
if account:
    msg = 'Account already exists !'
else:
    insert_sql = "INSERT INTO validate VALUES (?)"
    stmt = ibm_db.prepare(conn,insert_sql)
    ibm_db.bind_param(stmt, 1, email1)
    ibm_db.execute(stmt)
    msg = Message('Hello',sender ='shrikumar13102001@gmail.com',recipients = [email1])
    msg.body = 'Hello Flask message sent from Flask-Mail'
    msg.body = str(otp)
    mail.send(msg)
    return render_template('verify.html')
else:
    return render_template('email.html')
@app.route('/validate',methods=['GET', 'POST'])
def validate():
    user_otp = request.form['otp']
    if otp == int(user_otp):
        return render_template('index.html')
if __name__ == '__main__':
    app.run(debug = True)

```

## USN 5:

### WALLET UPDATE:

#### wallet.html

```

<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
    google.charts.load('current', {'packages':['corechart']});
    google.charts.setOnLoadCallback(drawChart);

function drawChart() {
    var data = google.visualization.arrayToDataTable([
        ['Items', 'Income', 'Expenses'],

```

```

    ['Education', 10000, 1000 ],
    ['Grocery', 9000, 1500],
    ['Hospital', 7500, 250],
    ['vehicle', 7250, 250 ]
  ]);

  var options = {
    title: 'Expense Tracker',
    curveType: 'function',
    legend: { position: 'bottom' }
  };

  var chart = new google.visualization.LineChart(document.getElementById('curve_chart'));

  chart.draw(data, options);
}
</script>
<style>
  body {
    background-image: "{{ url_for('static', filename='images/bg1.jpg') }}"
  }
.navbar {
  overflow: hidden;
  background-image: linear-gradient(to right, #8e2de2, #4a00e0);
  font-family: Arial, Helvetica, sans-serif;
}
.navbar a {
  float: left;
  font-size: 16px;
  color: #1e0076;
  text-align: center;
  padding: 14px 16px;
  text-decoration: none;
}

.dropdown {
  float: left;
  overflow: hidden;
  background-color: #8e2de2;
  font-family: Arial, Helvetica, sans-serif;
}

.dropdown .dropbtn {
  cursor: pointer;

```

```

font-size: 16px;
border: none;
outline: none;
color:#1e0076;
padding: 14px 16px;
background-color: inherit;
font-family: inherit;
margin: 0;
}

.navbar a:hover, .dropdown:hover .dropbtn, .dropbtn:focus {
    background-color:#daebeef;
}

.dropdown-content {
    display: none;
    position: absolute;
    background-image: linear-gradient(to right, #8e2de2, #4a00e0);
    min-width: 160px;
    box-shadow: #daebeef;
    z-index: 1;
}

.dropdown-content a {
    float: none;
    color:#1e0076;
    padding: 12px 16px;
    text-decoration: none;
    display: block;
    text-align: left;
}

.dropdown-content a:hover {
    background-color:#daebeef;
}

.show {
    display: block;
}

table{
    background-image: linear-gradient(to left, #8e2de2, #4a00e0);
    font-family: Arial, Helvetica, sans-serif;}

td{

```

```

font-family:Georgia, 'Times New Roman', Times, serif;
color:whitesmoke;
font-size: xx-large;
}

income{
  width: 200px;
  background-image: radial-gradient( circle farthest-corner at 22.4% 21.7%, rgba(4,189,228,1)
0%, rgba(2,83,185,1) 100.2% );
  font-family: Arial, Helvetica, sans-serif;
  align-items:right;
  color:#1e0076;
  float: right;
  font-size: 16px;
  text-align: center;
  padding: 14px 10px;
  text-decoration: none;

}

wallet{

  width: 200px;
  background-image: radial-gradient( circle farthest-corner at 22.4% 21.7%, rgba(4,189,228,1)
0%, rgba(2,83,185,1) 100.2% );
  font-family: Arial, Helvetica, sans-serif;
  align-items: right;
  color:#1e0076;
  float: right;
  font-size: 16px;
  text-align: center;
  padding: 14px 10px;
  text-decoration: none;
}
balance {

  width: 200px;
  background-image: radial-gradient( circle farthest-corner at 22.4% 21.7%, rgba(4,189,228,1)
0%, rgba(2,83,185,1) 100.2% );
  font-family: Arial, Helvetica, sans-serif;
  align-items:right;
  color:#1e0076;
  float: right;
  font-size: 16px;

```

```

text-align: center;
padding: 14px 10px;
text-decoration: none;
}

</style>
</head>
<body>

<div class="dropdown">
  <button class="dropbtn" onclick="myFunction()">Menu
    <i class="fa fa-caret-down"></i>
  </button>
  <form action="{{ url_for('wallet') }}" method="post"></form>
  <div class="dropdown-content" id="myDropdown">
    <hr><br>
    <a href="#back"><center><i class="fa fa-hand-o-left" style="font-size:24px"></i>
Back</center></a><hr><br>
    <a href="#history"><center><i class="fa fa-table" style="font-size:24px"></i> History
Table</center></a><hr><br>
    <a href="/income"><center><i class="fa fa-rupee" style="font-size:24px"></i> Add
Expense</center></a><hr><br>
    <a href="#limit"><center><i class="fa fa-line-chart" style="font-size:24px"></i> Set
Limit</center></a> </hr></br>
  </div>
</div>

<div class="navbar">
  <a href="#home"><i class="fa fa-home"></i> Home</a>
  <a href="#notify"><i class="fa fa-envelope"></i> Notify</a>

</form>
</div>
<script type="text/javascript">
function myFunction() {
  document.getElementById("myDropdown").classList.toggle("show");
}
window.onclick = function(e) {
  if (!e.target.matches('.dropbtn')) {
    var myDropdown = document.getElementById("myDropdown");
    if (myDropdown.classList.contains('show')) {
      myDropdown.classList.remove('show');
    }
  }
}

```



```

}
}
</script>
</br><br>
<hr><br>
<image>
<hr><br>
</image>
</br>
<table width="85%" height="50%" align="right">
  <tr><td align="center">PERSONAL EXPENSE TRACKER</td></tr>
</table>
</br><br><br><br><br><br><br><br>
<div id="curve_chart" style="width: 900px; height: 300px"></div>
<income> CURRENT INCOME </BR>RS 10,000</income>
<wallet>WALLET LIMIT</BR>RS 1,500</wallet>
<balance>BALANCE</BR>RS1,000</balance>
</body>
</html>

```

## TESTCASES:

PERSONAL EXPENSE TRACKER

ADD EXPENSE

Back History Table Set Limit

INCOMEvsEXPENSE

₹ 2000

income

Expense

Expense Details  
(like Grocery, Education, Hospital, etc.....)

type

Add Expense

IBM Db2 on Cloud

Load Data Load History **Tables** Views Indexes Aliases MQTs Sequences Application objects

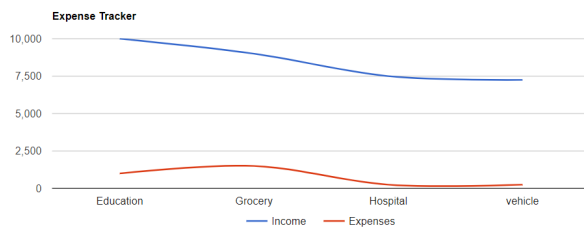
JPG92341.ADEXPENSE Back

Export to CSV

AMOUNT	INCOME	EXPENSE	CATEGORY
2000	10000	2000	Education
2000	12400	2300	hospital



## PERSONAL EXPENSE TRACKER



BALANCE RS1,000	WALLET LIMIT RS 1,500	CURRENT INCOME RS 10,000
--------------------	--------------------------	-----------------------------

**USN 6:**  
**EMAIL ALERT:**  
**email.html**

<!DOCTYPE html>

```
<html>
<head>
  <title>index</title>
</head>
<body>
  <style>
    @import
url("https://fonts.googleapis.com/css2?family=Sansita+Swashed:wght@600&display=swap");
body {
  margin: 0;
  padding: 0;
  box-sizing: border-box;
  display: flex;
  justify-content: center;
  align-items: center;
  height: 100vh;
  background: linear-gradient(45deg, greenyellow, dodgerblue);
  font-family: "Sansita Swashed", cursive;
}
.center {
  position: relative;
  padding: 50px 50px;
  background: #fff;
  border-radius: 10px;
}
.center h1 {
  font-size: 2em;
  border-left: 5px solid dodgerblue;
  padding: 10px;
  color: #000;
  letter-spacing: 5px;
  margin-bottom: 60px;
  font-weight: bold;
  padding-left: 10px;
}
.center .inputbox {
  position: relative;
  width: 300px;
  height: 50px;
  margin-bottom: 50px;
}
.center .inputbox input {
  position: absolute;
  top: 0;
```

```

left: 0;
width: 100%;
border: 2px solid #000;
outline: none;
background: none;
padding: 10px;
border-radius: 10px;
font-size: 1.2em;
}
.center .inputbox:last-child {
margin-bottom: 0;
}
.center .inputbox span {
position: absolute;
top: 14px;
left: 20px;
font-size: 1em;
transition: 0.6s;
font-family: sans-serif;
}
.center .inputbox input:focus ~ span,
.center .inputbox input:valid ~ span {
transform: translateX(-13px) translateY(-35px);
font-size: 1em;
}
.center .inputbox [type="button"] {
width: 50%;
background: dodgerblue;
color: #fff;
border: #fff;
}
.center .inputbox:hover [type="button"] {
background: linear-gradient(45deg, greenyellow, dodgerblue);
}

```

</style>

<form action = "{{ url\_for('verify') }}" method = "post">

<div class="center">

<h1>Please verify your mailid here!!</h1>

<form>

<div class="inputbox">

<input type="text" required="required" name="email1">

<span>Email</span>

</div>

<div class="inputbox">

```
        <input type="submit" value="submit">
    </div>
</form>
</div>
</form>
</body>
</html>
```

## Verify.html

```
<!DOCTYPE html>
<html>
<head>
    <title>OTP Verification</title>
</head>
<body>
    <html>
<head>
    <meta charset="utf-8">
    <title>Stackfindover: Sign in</title>
    <link rel="stylesheet" type="text/css" href="style.css">
</head>

<body>
    <style>
        * {
            padding: 0;
            margin: 0;
            color: #1a1f36;
            box-sizing: border-box;
            word-wrap: break-word;
            font-family: -apple-system,BlinkMacSystemFont,Segoe UI,Roboto,Helvetica
            Neue,Ubuntu,sans-serif;
        }
        body {
            min-height: 100%;
            background-color: #ffffff;
        }
        h1 {
            letter-spacing: -1px;
        }
        a {
            color: #5469d4;
            text-decoration: unset;
```

```

}
.login-root {
  background: #fff;
  display: flex;
  width: 100%;
  min-height: 100vh;
  overflow: hidden;
}
.loginbackground {
  min-height: 692px;
  position: fixed;
  bottom: 0;
  left: 0;
  right: 0;
  top: 0;
  z-index: 0;
  overflow: hidden;
}
.flex-flex {
  display: flex;
}
.align-center {
  align-items: center;
}
.center-center {
  align-items: center;
  justify-content: center;
}
.box-root {
  box-sizing: border-box;
}
.flex-direction--column {
  -ms-flex-direction: column;
  flex-direction: column;
}
.loginbackground-gridContainer {
  display: -ms-grid;
  display: grid;
  -ms-grid-columns: [start] 1fr [left-gutter] (86.6px)[16] [left-gutter] 1fr [end];
  grid-template-columns: [start] 1fr [left-gutter] repeat(16,86.6px) [left-gutter] 1fr [end];
  -ms-grid-rows: [top] 1fr [top-gutter] (64px)[8] [bottom-gutter] 1fr [bottom];
  grid-template-rows: [top] 1fr [top-gutter] repeat(8,64px) [bottom-gutter] 1fr [bottom];
  justify-content: center;
  margin: 0 -2%;
}

```

```
    transform: rotate(-12deg) skew(-12deg);
}
.box-divider--light-all-2 {
    box-shadow: inset 0 0 0 2px #e3e8ee;
}
.box-background--blue {
    background-color: #5469d4;
}
.box-background--white {
    background-color: #ffffff;
}
.box-background--blue800 {
    background-color: #212d63;
}
.box-background--gray100 {
    background-color: #e3e8ee;
}
.box-background--cyan200 {
    background-color: #7fd3ed;
}
.padding-top--64 {
    padding-top: 64px;
}
.padding-top--24 {
    padding-top: 24px;
}
.padding-top--48 {
    padding-top: 48px;
}
.padding-bottom--24 {
    padding-bottom: 24px;
}
.padding-horizontal--48 {
    padding: 48px;
}
.padding-bottom--15 {
    padding-bottom: 15px;
}

.flex-justifyContent--center {
    -ms-flex-pack: center;
    justify-content: center;
}
```

```
.formbg {
  margin: 0px auto;
  width: 100%;
  max-width: 448px;
  background: white;
  border-radius: 4px;
  box-shadow: rgba(60, 66, 87, 0.12) 0px 7px 14px 0px, rgba(0, 0, 0, 0.12) 0px 3px 6px 0px;
}
span {
  display: block;
  font-size: 20px;
  line-height: 28px;
  color: #1a1f36;
}
label {
  margin-bottom: 10px;
}
.reset-pass a,label {
  font-size: 14px;
  font-weight: 600;
  display: block;
}
.reset-pass > a {
  text-align: right;
  margin-bottom: 10px;
}
.grid--50-50 {
  display: grid;
  grid-template-columns: 50% 50%;
  align-items: center;
}

.field input {
  font-size: 16px;
  line-height: 28px;
  padding: 8px 16px;
  width: 100%;
  min-height: 44px;
  border: unset;
  border-radius: 4px;
  outline-color: rgb(84 105 212 / 0.5);
  background-color: rgb(255, 255, 255);
  box-shadow: rgba(0, 0, 0, 0) 0px 0px 0px 0px,
```



```

        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(60, 66, 87, 0.16) 0px 0px 0px 1px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px;
    }

```

```

input[type="submit"] {
    background-color: rgb(84, 105, 212);
    box-shadow: rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(0, 0, 0, 0.12) 0px 1px 1px 0px,
        rgb(84, 105, 212) 0px 0px 0px 1px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(0, 0, 0, 0) 0px 0px 0px 0px,
        rgba(60, 66, 87, 0.08) 0px 2px 5px 0px;
    color: #fff;
    font-weight: 600;
    cursor: pointer;
}

```

```

.field-checkbox input {
    width: 20px;
    height: 15px;
    margin-right: 5px;
    box-shadow: unset;
    min-height: unset;
}

```

```

.field-checkbox label {
    display: flex;
    align-items: center;
    margin: 0;
}

```

```

a.ssolink {
    display: block;
    text-align: center;
    font-weight: 600;
}

```

```

.footer-link span {
    font-size: 14px;
    text-align: center;
}

```

```

.listing a {
    color: #697386;
}

```

```

    font-weight: 600;
    margin: 0 10px;
}

.animationRightLeft {
    animation: animationRightLeft 2s ease-in-out infinite;
}
.animationLeftRight {
    animation: animationLeftRight 2s ease-in-out infinite;
}
.tans3s {
    animation: animationLeftRight 3s ease-in-out infinite;
}
.tans4s {
    animation: animationLeftRight 4s ease-in-out infinite;
}

```

```

@keyframes animationLeftRight {
    0% {
        transform: translateX(0px);
    }
    50% {
        transform: translateX(1000px);
    }
    100% {
        transform: translateX(0px);
    }
}

```

```

@keyframes animationRightLeft {
    0% {
        transform: translateX(0px);
    }
    50% {
        transform: translateX(-1000px);
    }
    100% {
        transform: translateX(0px);
    }
}

```

</style>

<form action = "{{ url\_for('validate') }}" method="post">

<div class="login-root">

<h4> OTP has been sent to the email id. Please check the email for the confirmation.</h4>

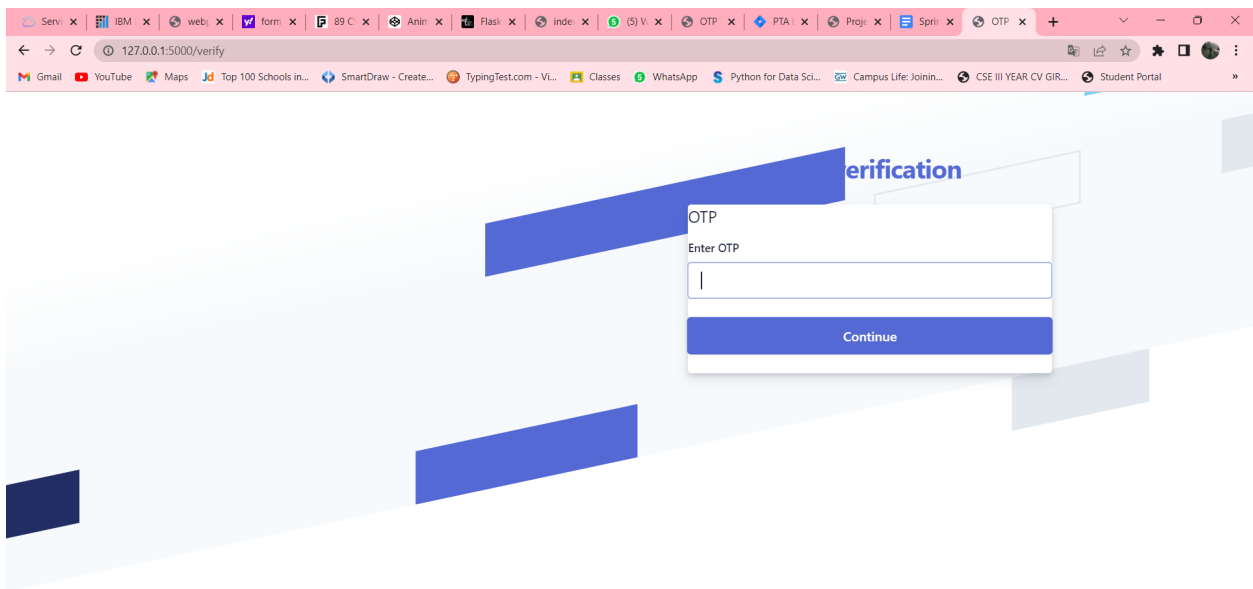
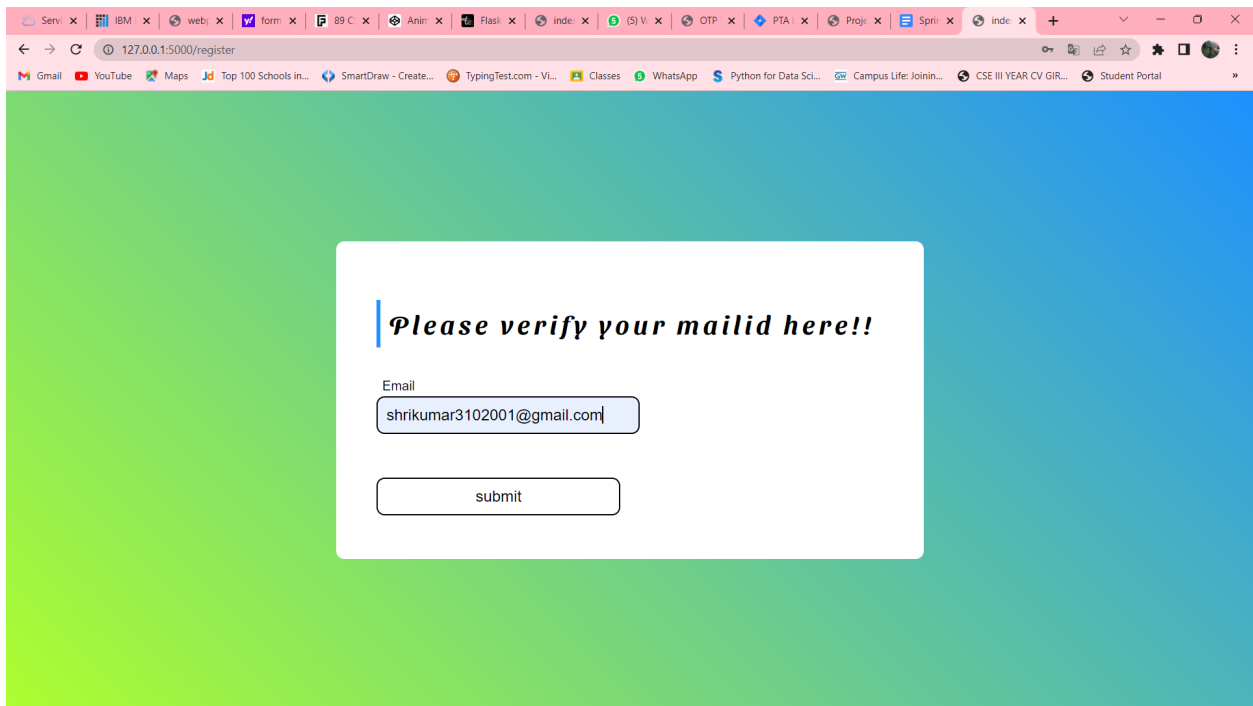
```

<div class="box-root flex-flex flex-direction--column" style="min-height: 100vh;flex-grow: 1;">
  <div class="loginbackground box-background--white padding-top--64">
    <div class="loginbackground-gridContainer">
      <div class="box-root flex-flex" style="grid-area: top / start / 8 / end;">
        <div class="box-root" style="background-image: linear-gradient(white 0%, rgb(247, 250, 252) 33%); flex-grow: 1;">
        </div>
      </div>
      <div class="box-root flex-flex" style="grid-area: 4 / 2 / auto / 5;">
        <div class="box-root box-divider--light-all-2 animationLeftRight tans3s" style="flex-grow: 1;"></div>
        </div>
        <div class="box-root flex-flex" style="grid-area: 6 / start / auto / 2;">
          <div class="box-root box-background--blue800" style="flex-grow: 1;"></div>
        </div>
        <div class="box-root flex-flex" style="grid-area: 7 / start / auto / 4;">
          <div class="box-root box-background--blue animationLeftRight" style="flex-grow: 1;"></div>
          </div>
          <div class="box-root flex-flex" style="grid-area: 8 / 4 / auto / 6;">
            <div class="box-root box-background--gray100 animationLeftRight tans3s" style="flex-grow: 1;"></div>
            </div>
            <div class="box-root flex-flex" style="grid-area: 2 / 15 / auto / end;">
              <div class="box-root box-background--cyan200 animationRightLeft tans4s" style="flex-grow: 1;"></div>
              </div>
              <div class="box-root flex-flex" style="grid-area: 3 / 14 / auto / end;">
                <div class="box-root box-background--blue animationRightLeft" style="flex-grow: 1;"></div>
                </div>
                <div class="box-root flex-flex" style="grid-area: 4 / 17 / auto / 20;">
                  <div class="box-root box-background--gray100 animationRightLeft tans4s" style="flex-grow: 1;"></div>
                  </div>
                  <div class="box-root flex-flex" style="grid-area: 5 / 14 / auto / 17;">
                    <div class="box-root box-divider--light-all-2 animationRightLeft tans3s" style="flex-grow: 1;"></div>
                    </div>
                    </div>
                    </div>
                    <div class="box-root padding-top--24 flex-flex flex-direction--column" style="flex-grow: 1; z-index: 9;">

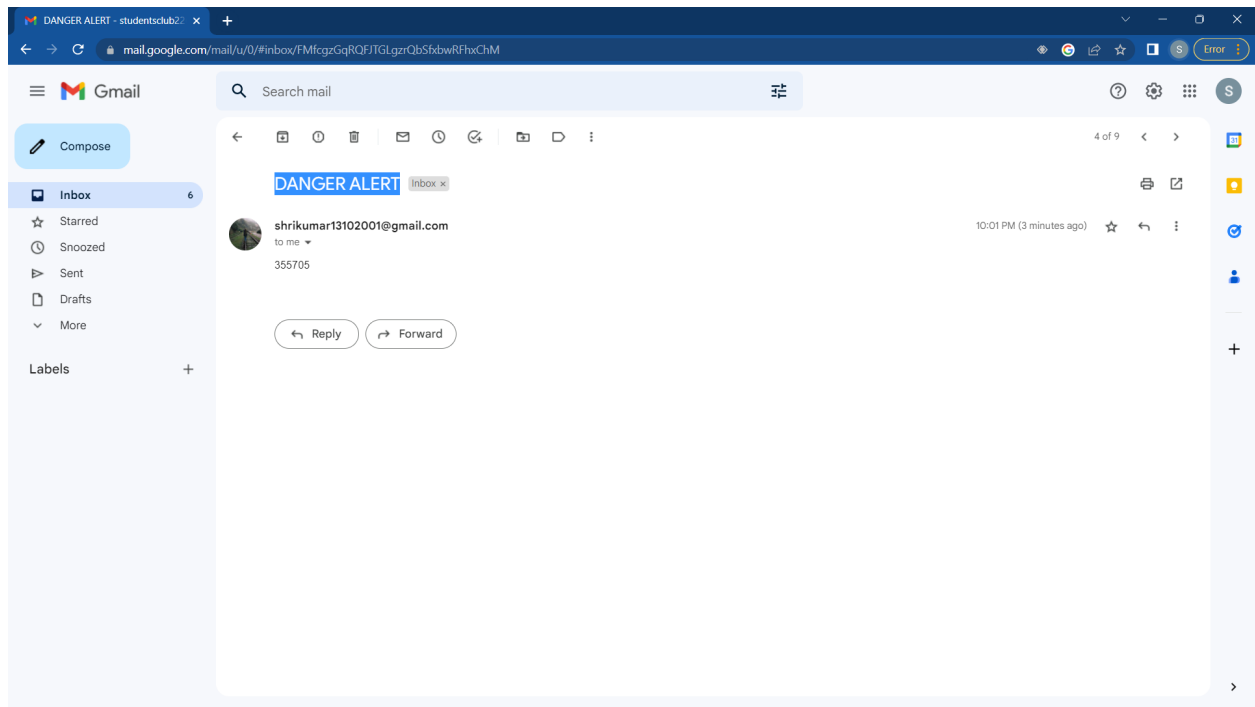
```

[illegible]

## TESTCASES:



Email alert:



History page:

Sen

IBM

wet

fori

89

Ani

ind

(5)

OTI

PTA

Pro

Spr

wet

Nei

IBM

Upi

W

hov

Ret

pyti

sql

x

+

-

□

×

127.0.0.1:5000/table

Gmail

YouTube

Maps

Top 100 Schools in...

SmartDraw - Create...

TypingTest.com - VL...

Classes

WhatsApp

Python for Data Sci...

Campus Life: Joinin...

CSE III YEAR CV GIR...

Student Portal

EXPENSE HISTORY

Amount	Income	Expense	Category
1000	12000	2000	hospital

Sprint-2.pdf

Show all

×