

Assignment -2
Flask Application Development

Assignment Date	10 October 2022
Student Name	DINESH KUMAR S R
Register Number	822719104015
Maximum Marks	2 Marks

Question-1:

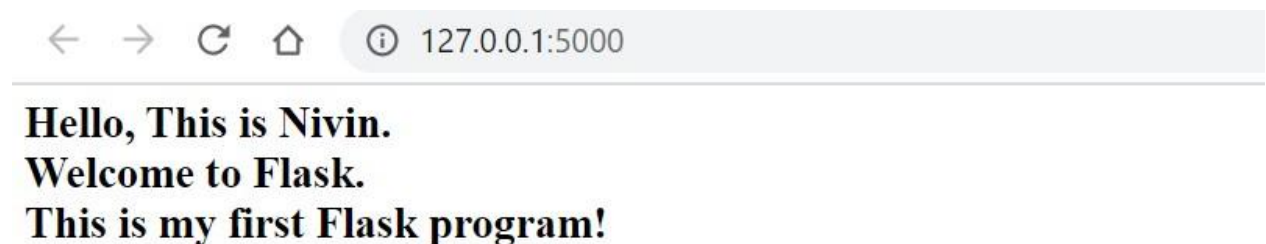
Create a Flask App

Solution:

app.py:

```
from flask import Flask
app = Flask(__name__)
@app.route("/")
def hello_world():
    return "<h1>Hello, This is Nivin. <br> Welcome to Flask.
    <br>This is my first Flask program!</h1>"
```

Output:



Question-2:

Add the Home page, About Page

Solution:

app.py:

```
from flask import Flask
from flask import render_template
```

```

app = Flask(__name__)

@app.route("/")
def hello_world():
    return "<h1>Hello, This is Nivin. <br> Welcome to Flask.
<br>This is a Flask program!</h1>"

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

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

@app.route("/profiles/<username>")
def profile(username):
    return " <h1>Hello,  "+ username  +".  Welcome  to  Flask
Programming!</h1>"

```

home.html:

```

<!DOCTYPE html>
<head>
    <title>
        Home
    </title>
    <link rel="stylesheet" href="static\css\home.css">
</head>
<body>
    <div class="titlediv">
        <h1>Home Page</h1>
    </div>
    <div class="bodydiv">
        <h3>
            This is home page for create home page assignment
        </h3>
    </div>
    
</body>
</html>

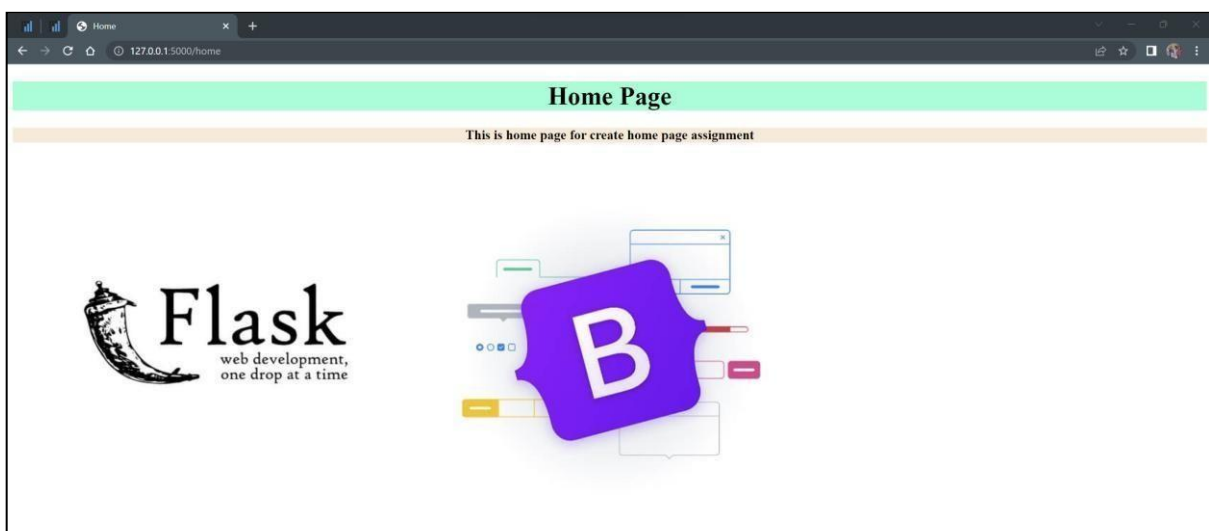
```

[about.html:](#)

```
<!DOCTYPE html>
<head>
  <title>
    About
  </title>
  <link rel="stylesheet" href="static\css\about.css">
</head>
<body>
  <div class="titlediv">
    <h1>About Page</h1>
  </div>
  <div class="bodydiv">
    <h3>
      This is about page for create about page assignment
    <br>
      This page is used to tell about the assignment that is
given to the flask
    <br>
    </h3>
  </div>
</body>
</html>
```

Output:

Home Page:



About Page:



Question-3:

Add the Bootstrap

Solution:

app.py:

```
from flask import Flask
from flask import render_template

app = Flask(__name__)

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

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

@app.route("/withoutbootstrap")
def withoutbootstrap():
    return render_template('withoutbootstrap.html')
```

base.html:

```
<!DOCTYPE html>
<head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap
.min.css" rel="stylesheet">
```

```

    <link
href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">
    <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.b
undle.min.js"></script>
    {% block head %} {% endblock %}
</head>
<body>
    <body background="static/images/signup.png">
    <div class="shadow p-3 mb-5 rounded">
        <nav class="navbar navbar-expand-md navbar-dark bg-dark">
            <div class="container-fluid">
                <div class="navbar-collapse collapse w-100 order-1
order-md-0 dual-collapse2">
                    <ul class="navbar-nav me-auto">
                        <li class="nav-item active">
                            <a class="nav-link" href="home">Home</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link"
href="about">About</a>
                        </li>
                    </ul>
                </div>
                <div class="mx-auto order-0">
                    <a class="navbar-brand mx-auto"
href="home">Personal Expense Tracker Application</a>
                    <button class="navbar-toggler" type="button"
data-bs-toggle="collapse" data-bs-target=".dual-collapse2">
                        <span class="navbar-toggler-icon"></span>
                    </button>
                </div>
                <div class="navbar-collapse collapse w-100 order-3
dual-collapse2">
                    <ul class="navbar-nav ms-auto">
                        <li class="nav-item">
                            <a class="nav-link"
href="signin">Login</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link"
href="signup">Register</a>
                        </li>
                    </ul>
                </div>
            </div>
        </nav>
    </div>

```

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

home.html:

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

{% block head %}
    <title>Home</title>
{% endblock %}

{% block body %}
    <div style="text-align:center;" >
        
    </div>
{% endblock %}
```

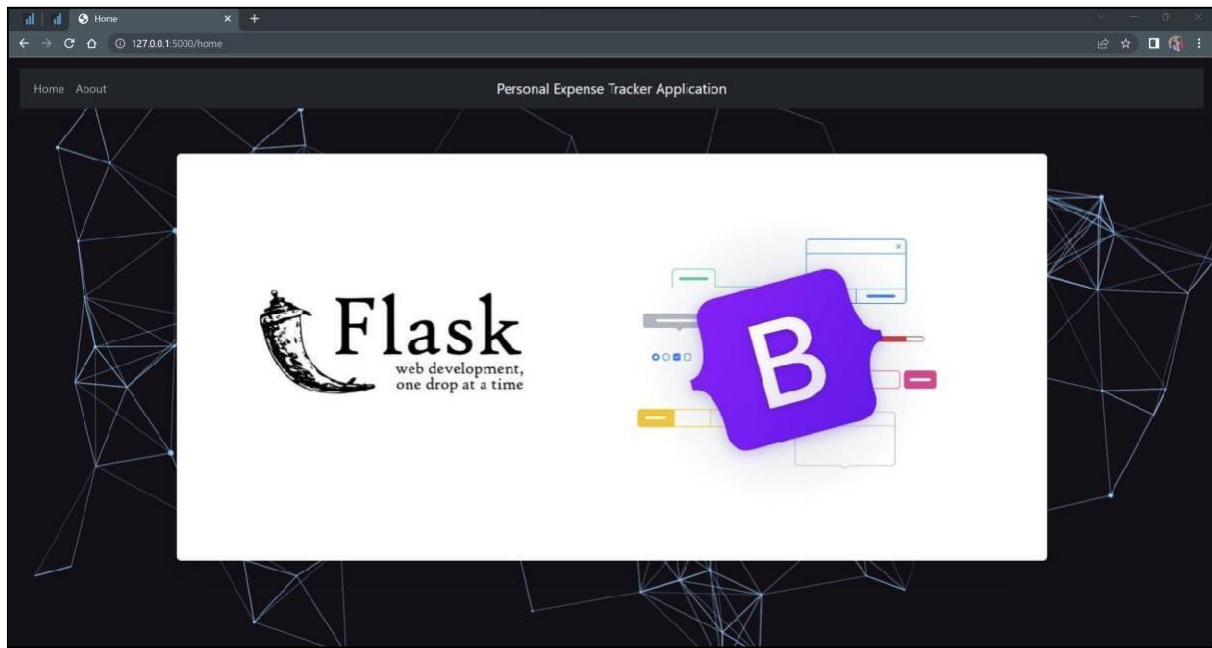
about.html:

```
{% block head %}
    <title>About</title>
{% endblock %}

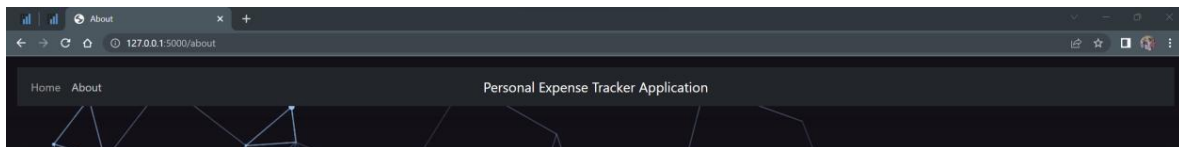
{% block body %}
    <div class="shadow-lg p-3 mb-5 bg-body rounded p-3 mx-auto">
        <h1 class="d-flex justify-content-center">About Page</h1>
        <h3>
            This is about page for create about page assignment
            <br>
            This page is used to tell about the assignment that is
given to the flask
            <br>
        </h3>
    </div>
{% endblock %}
```

Output:

Home Page:

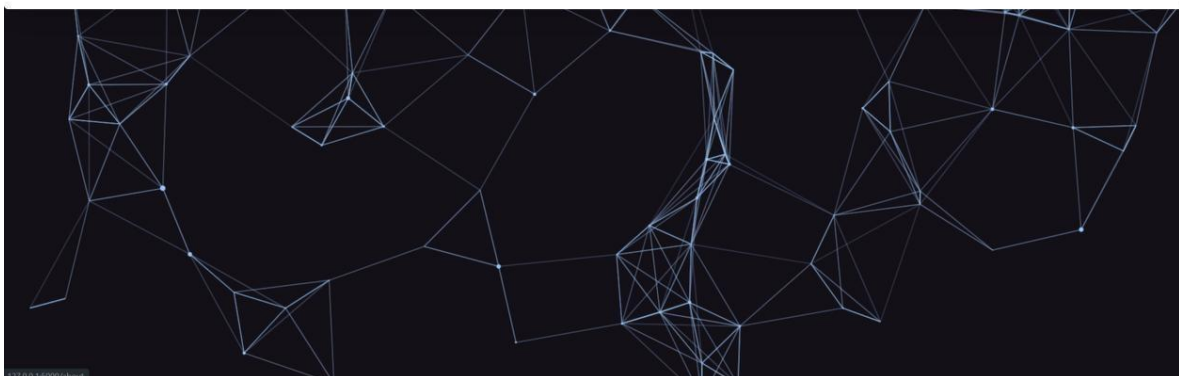


About Page:



About Page

This is about page for create about page using Bootstrap assignment
This page is used to tell about the assignment that is given to the flask



Question-4,5:

4. Add the Sign in page and add the Signup Page with database connectivity

5. Use IBM Db2 as Database.

Solution:

app.py:

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

app = Flask(__name__)
app.debug=True

conn = ibm_db.connect("DATABASE=bludb;HOSTNAME=54a2f15b-5c0f-46df-
8954-
7e38e612c2bd.c1ogj3sd0tgtu0lqde00.databases.appdomain.cloud;PORT=327
33;SECURITY=SSL;SSLServerCertificate=DigiCertGlobalRootCA.crt;UID=cj
177846;PWD=k3WDjLvi6S4zsDZi", '', '')

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

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

@app.route("/signup", methods=('GET', 'POST'))
def signup():
    if request.method == 'POST':
        name = request.form['firstname']
        email = request.form['femail']
        username = request.form['username']
        mobile = request.form['mobile']
        password = request.form['password']

        sql = "SELECT * FROM users WHERE name =? and username=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,name)
        ibm_db.bind_param(stmt,2,username)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)

        if account:
```



```

        return render_template('registered.html')
    else:
        insert_sql = "INSERT INTO users 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, username)
        ibm_db.bind_param(prepare_stmt, 4, mobile)
        ibm_db.bind_param(prepare_stmt, 5, password)
        ibm_db.execute(prepare_stmt)
        return render_template('success.html')

    return render_template('signup.html')

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

@app.route("/signin" , methods=('GET', 'POST'))
def signin():
    if request.method == 'POST':
        femail = request.form['femail']
        fpassword = request.form['password']

        sql = "SELECT * FROM users WHERE email =? and password=?"
        stmt = ibm_db.prepare(conn, sql)
        ibm_db.bind_param(stmt,1,femail)
        ibm_db.bind_param(stmt,2,fpassword)
        ibm_db.execute(stmt)
        account = ibm_db.fetch_assoc(stmt)

        if account:
            return render_template('loginsuccess.html')
        else:
            return render_template('loginfailure.html')
    return render_template('signin.html')

if __name__ == '__main__':
    app.run()

```

base.html:

```

<!DOCTYPE html>
<head>
    <meta charset="utf-8">

```

```

        <meta name="viewport" content="width=device-width, initial-
scale=1">
        <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/css/bootstrap
.min.css" rel="stylesheet">
        <link
href="https://getbootstrap.com/docs/5.2/assets/css/docs.css"
rel="stylesheet">
        <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.1/dist/js/bootstrap.b
undle.min.js"></script>
        {% block head %} {% endblock %}
</head>
<body>
    <body background="static/images/signup.png">
    <div class="shadow p-3 mb-5 rounded">
        <nav class="navbar navbar-expand-md navbar-dark bg-dark">
            <div class="container-fluid">
                <div class="navbar-collapse collapse w-100 order-1
order-md-0 dual-collapse2">
                    <ul class="navbar-nav me-auto">
                        <li class="nav-item active">
                            <a class="nav-link" href="home">Home</a>
                        </li>
                        <li class="nav-item">
                            <a class="nav-link"
href="about">About</a>
                        </li>
                    </ul>
                </div>
                <div class="mx-auto order-0">
                    <a class="navbar-brand mx-auto"
href="home">Personal Expense Tracker Application</a>
                    <button class="navbar-toggler" type="button"
data-bs-toggle="collapse" data-bs-target=".dual-collapse2">
                        <span class="navbar-toggler-icon"></span>
                    </button>
                </div>
                <div class="navbar-collapse collapse w-100 order-3
dual-collapse2">
                    <ul class="navbar-nav ms-auto">
                        <li class="nav-item">
                            <a class="nav-link"
href="signin">Login</a>
                        </li>
                        <li class="nav-item">

```

```

                                <a class="nav-link"
href="signup">Register</a>
                                </li>
                                </ul>
                                </div>
                                </div>
                                </nav>
                                </div>
                                {% block body %} {% endblock %}
</body>
</html>

```

signin.html:

```

{% extends "base.html"%}

{% block head %}
    <title>Sign In</title>
    <link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.13.0/css/all.min.css">
{% endblock %}

{% block body %}
    <form class="row g-3 needs-validation" novalidate novalidate
action = "{{ url_for('signin') }}" method = "POST">
        <div class="w-25 p-3 mx-auto shadow-lg p-3 mb-5 bg-body
rounded">
            <div class="mb-3" name="formpad">

                <label for="femail" class="form-label">Email:
</label>

                <input type="email" class="form-control"
name="femail" placeholder="Your Email" required>
                <div class="invalid-feedback">
                    Please enter valid email
                </div><br>

                <label for="password" class="form-label">Password :
</label>

                <input type="password" class="form-control"
name="password" placeholder="Your Password" required>
                <div class="invalid-feedback">
                    Please enter your password
                </div><br>

```

```

        <div class="col-12">
            <button class="btn btn-primary"
type="submit">Login</button>
        </div>
    </div>
</div>
</form>

<script>
    (function()
    {
        'use strict';
        window.addEventListener('load', function()
        {
            var forms = document.getElementsByClassName('needs-
validation');
            var validation = Array.prototype.filter.call(forms,
function(form)
            {
                form.addEventListener('submit', function(event)
                {
                    if (form.checkValidity() === false)
                    {
                        event.preventDefault();
                        event.stopPropagation();
                    }
                    form.classList.add('was-validated');
                }, false);
            });
        }, false);
    })();
</script>
{% endblock %}

```

signup.html:

```

{% extends "base.html"%}

{% block head %}
    <title>Signup</title>
{% endblock %}

{% block body %}
    <form class="row g-3 needs-validation" novalidate action =
"{{url_for('signup')}}" method = "POST">

```

```

<div class="shadow-lg p-3 mb-5 bg-body rounded w-25 p-3
mx-auto">
    <div class="mb-3" name="formpad">
        <label for="firstname" class="form-label">Name :
    </label>

        <input type="text" class="form-control"
name="firstname" placeholder="Your Name" required>
        <div class="invalid-feedback">
            Please enter your name
        </div><br>

        <label for="femail" class="form-label">Email:
    </label>

        <input type="email" class="form-control"
name="femail" placeholder="Your Email" required>
        <div class="invalid-feedback">
            Please enter valid email
        </div><br>

        <label for="username" class="form-label">Username
: </label>

        <input type="text" class="form-control"
name="username" placeholder="Your User Name" required>
        <div class="invalid-feedback">
            Please enter your username
        </div><br>

        <label for="mobile" class="form-label">Mobile :
    </label>

        <input type="text" class="form-control"
name="mobile" placeholder="Your mobile number" required>
        <div class="invalid-feedback">
            Please enter your mobile
        </div><br>

        <label for="password" class="form-label">Password
: </label>

        <input type="password" class="form-control"
name="password" placeholder="Your Password" onkeyup='check();'
required>
        <div class="invalid-feedback">
            Please enter your password
        </div><br>

        <label for="repassword" class="form-label">Re-
type password : </label>

```

```

        <input type="password" class="form-control"
name="repassword" placeholder="Re-type your Password"
onkeyup='check();'required>
        <div class="invalid-feedback">
            Please retype your password
        </div><br>

        <div class="col-12">
            <button class="btn btn-primary" type="submit"
onclick="matchPassword()">Register</button>
        </div>
    </div>
</div>
</form>

<script>
    (function()
    {
        'use strict';
        window.addEventListener('load', function()
        {
            var forms = document.getElementsByClassName('needs-
validation');
            var validation = Array.prototype.filter.call(forms,
function(form)
            {
                form.addEventListener('submit', function(event)
                {
                    if (form.checkValidity() === false)
                    {
                        event.preventDefault();
                        event.stopPropagation();
                    }
                    form.classList.add('was-validated');
                }, false);
            });
        }, false);
    }
    );

</script>
{% endblock %}

```

SQL Code :

```

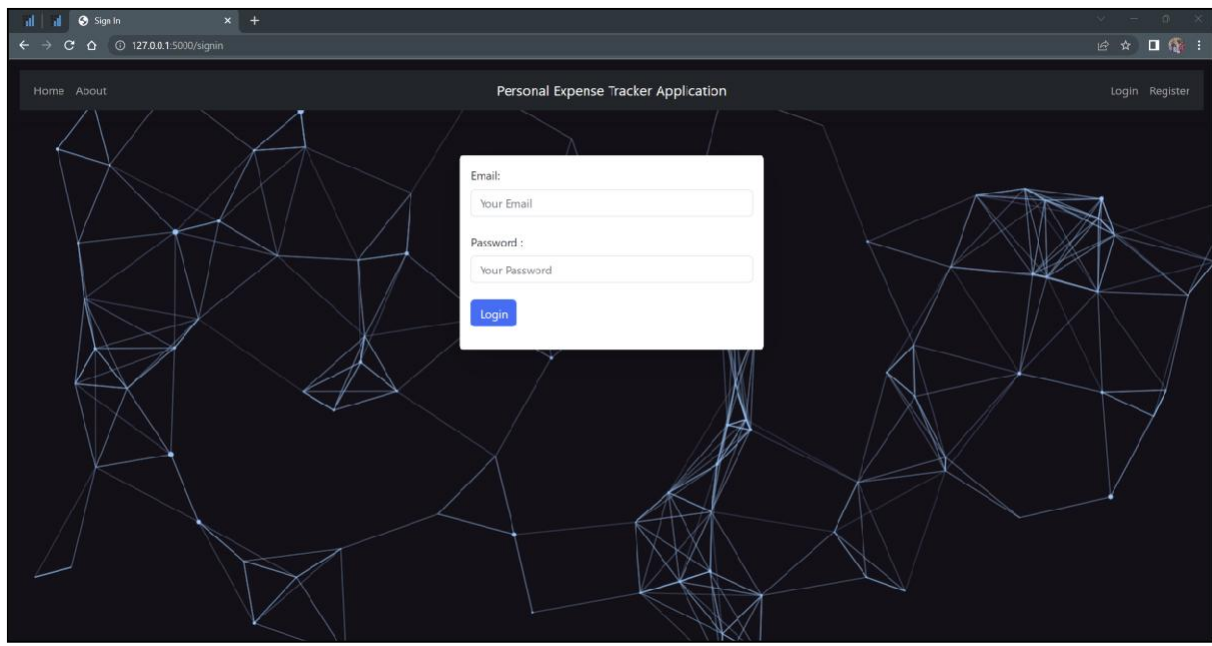
CREATE TABLE users
( NAME VARCHAR(255),

```

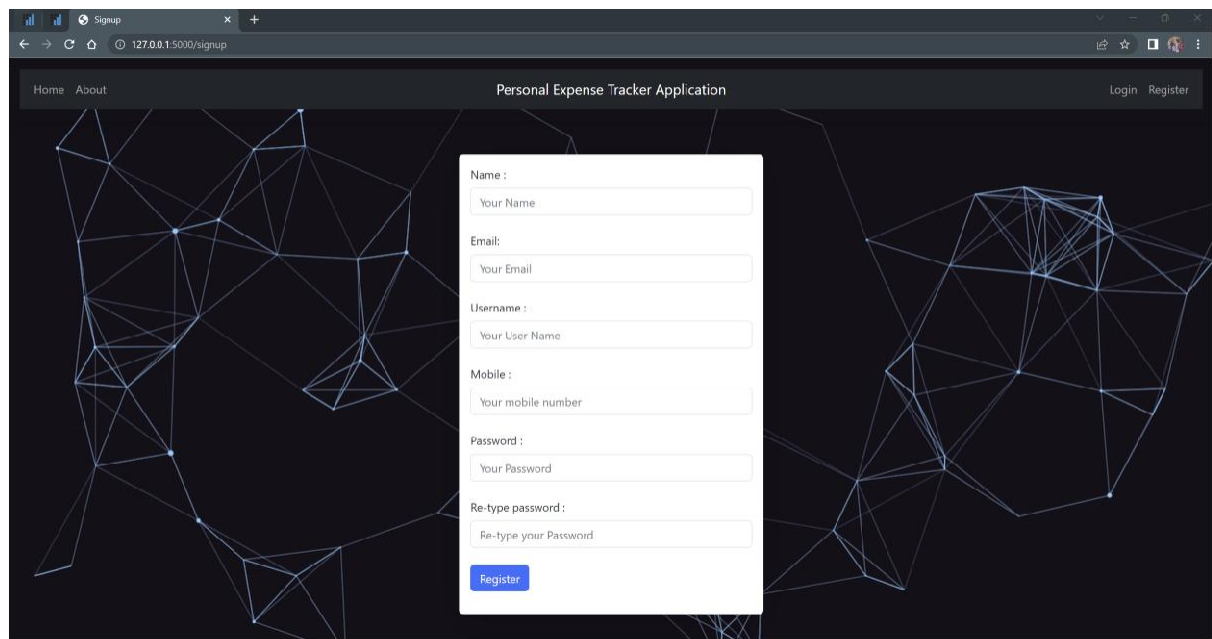
```
email VARCHAR(255),  
username VARCHAR(255),  
mobile VARCHAR(255),  
password VARCHAR(255)  
);
```

Output:

Signin Page:



Signup Page:



Database Details:

The screenshot shows the IBM Db2 on Cloud web interface. The 'Tables' tab is selected, displaying a list of tables. The 'USERS' table is selected, and its definition is shown on the right. The table has five columns: NAME, EMAIL, USERNAME, MOBILE, and PASSWORD, all of type VARCHAR(255) and nullable.

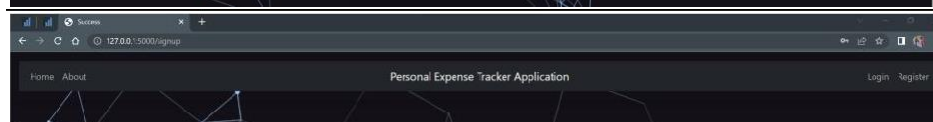
Name	Schema	Properties
USERS	EXY80423	...

Name	Data type	Nullable	Length	Scale
NAME	VARCHAR	Y	255	0
EMAIL	VARCHAR	Y	255	0
USERNAME	VARCHAR	Y	255	0
MOBILE	VARCHAR	Y	255	0
PASSWORD	VARCHAR	Y	255	0

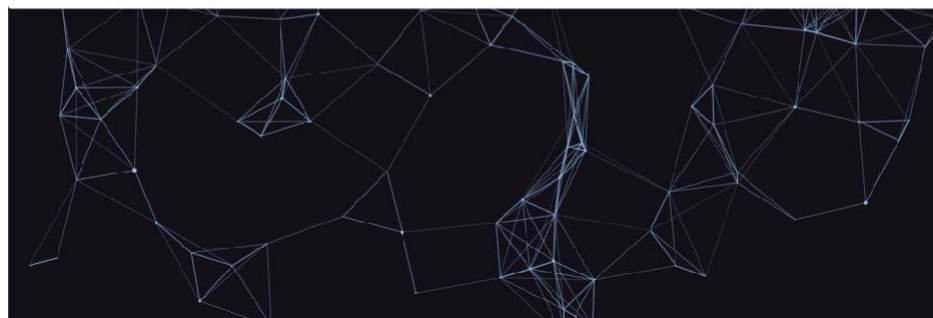
Registration Process:

If New User:

The screenshot shows the 'Personal Expense Tracker Application' registration form. The form is titled 'Personal Expense Tracker Application' and has a 'Login Register' link. The form fields are: Name (Prem Kashyap Chalakamathi), Email (nmani3008@gmail.com), Username (Prem0046), Mobile (96686454), Password (masked with asterisks), and Re-type password (masked with asterisks). A 'Register' button is at the bottom.



Registration successful...
Proceed to Login Page to signin



If already registered User:

The first screenshot shows the 'Personal Expense Tracker Application' registration page. The form includes the following fields:

- Name: Prem Kashyap Chalakamarthi
- Email: nmmani3008@gmail.com
- Username: Prem0046
- Mobile: 96686454
- Password: (masked with asterisks)
- Re-type password: (masked with asterisks)

The 'Register' button is visible at the bottom of the form.

The second screenshot shows the same application after a successful registration. A message is displayed: "Already Registered..... Proceed to Login Page to sign in".

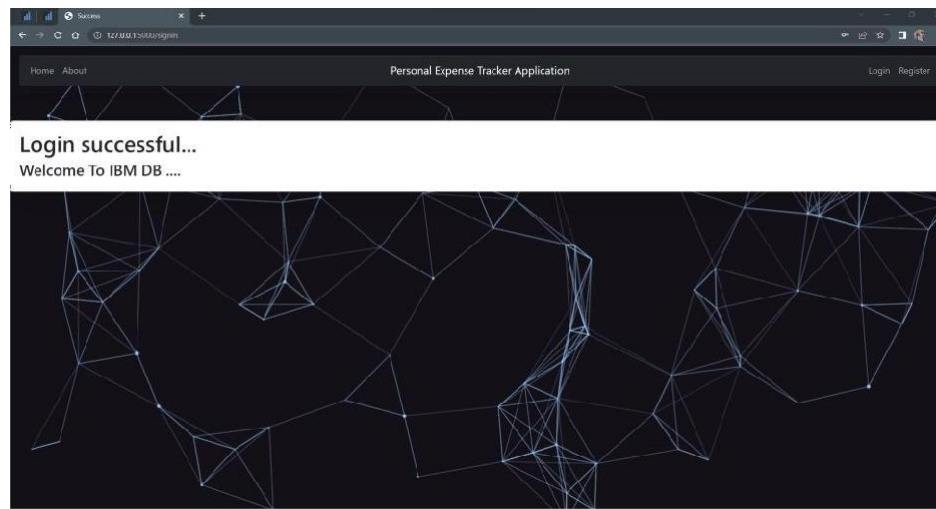
Login Process:

If already registered User:

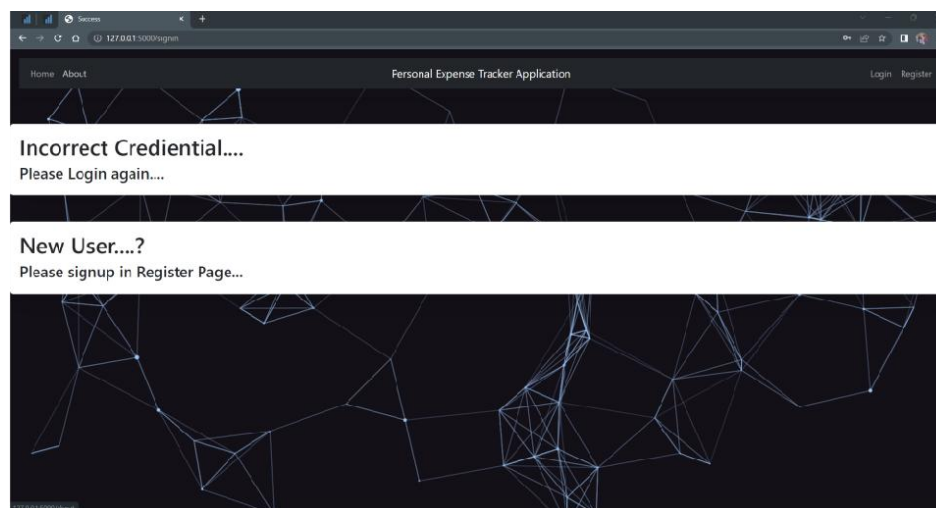
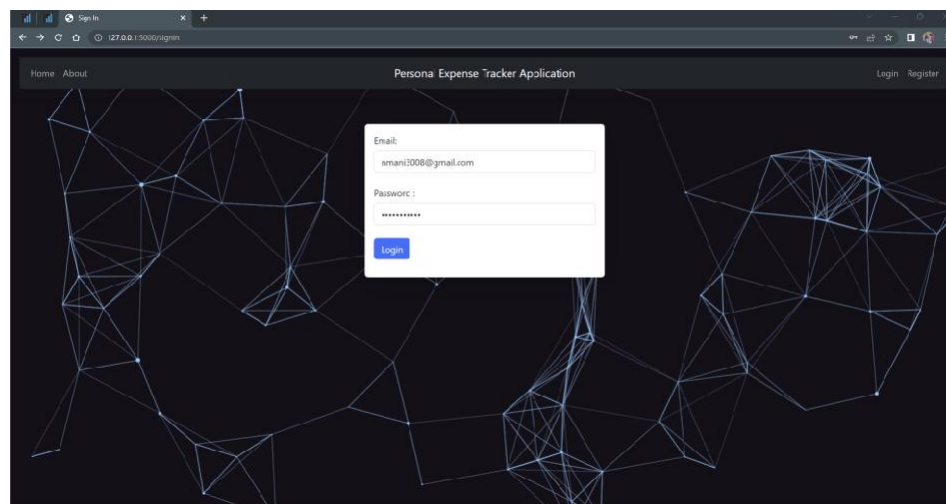
The screenshot shows the 'Personal Expense Tracker Application' login page. The form includes the following fields:

- Email: nmmani3008@gmail.com
- Password: (masked with asterisks)

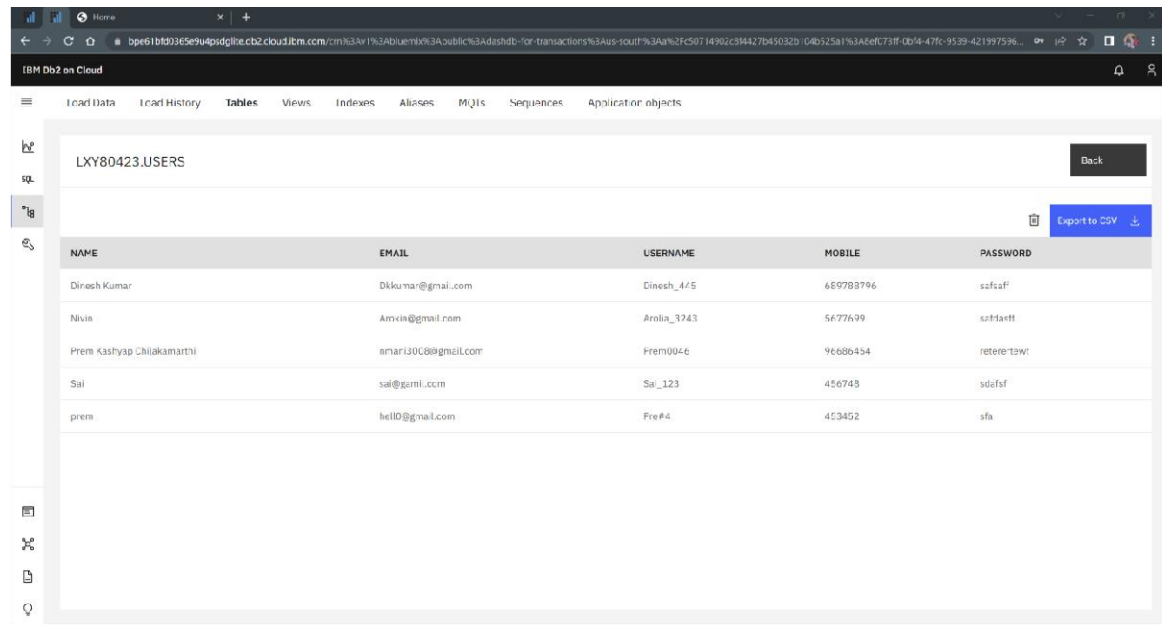
The 'Login' button is visible at the bottom of the form.



If Not Registered or Invalid Credentials:



Some User Deatails in IBM-db2:



The screenshot shows the IBM Db2 on Cloud web interface. The browser address bar displays a long URL. The interface includes a top navigation bar with tabs for 'Load Data', 'Load History', 'Tables', 'Views', 'Indexes', 'Aliases', 'MJIs', 'Sequences', and 'Application objects'. The 'Tables' tab is selected, showing a table named 'LXY80423.USERS'. A 'Back' button is in the top right corner. Below the table name, there is a trash icon and an 'Export to CSV' button. The table itself has five columns: 'NAME', 'EMAIL', 'USERNAME', 'MOBILE', and 'PASSWORD'. It contains five rows of user data.

NAME	EMAIL	USERNAME	MOBILE	PASSWORD
Direch Kumar	Dkkumar@gmail.com	Dinesh_445	669783796	szfraff
Nivis	Amcin@gmail.com	Amilia_3743	5677699	szfraff
Prem Kashyap Chilikamathi	amari3008@gmail.com	Prem0046	96686454	retete1awt
Sai	sai@gmail.com	Sa_123	456743	sdafsf
prem	hell0@gmail.com	Prem#4	453452	sfa