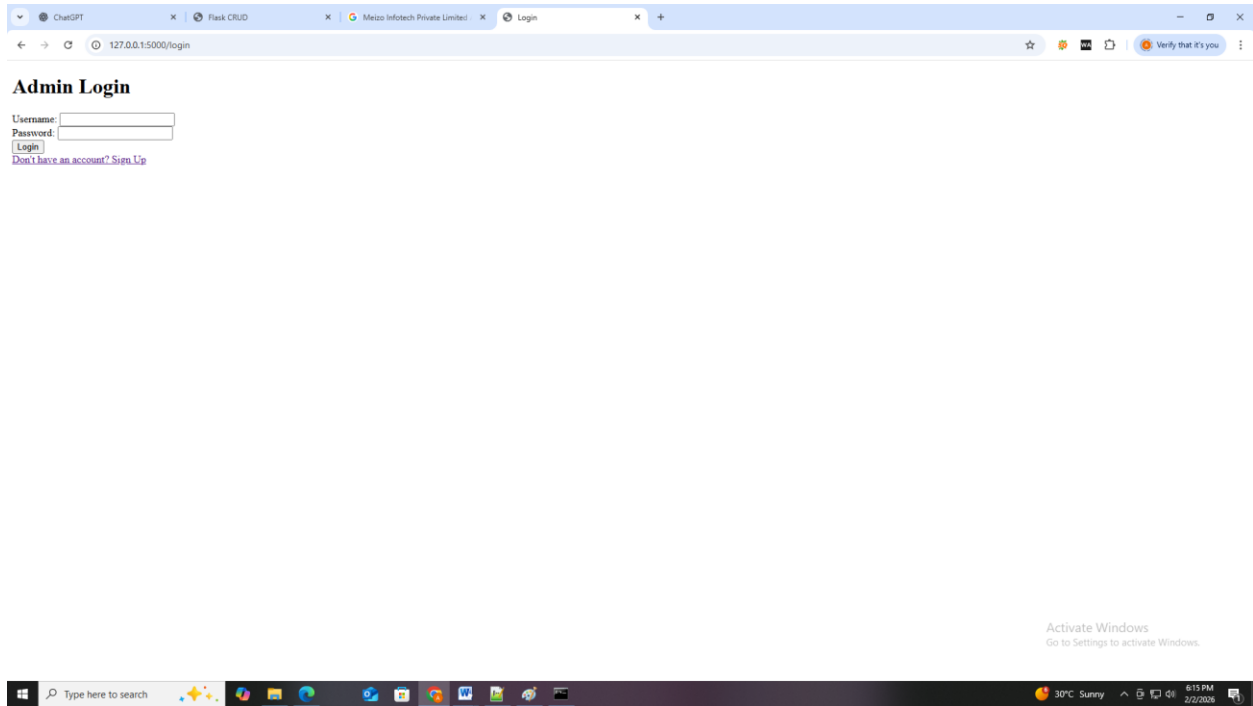
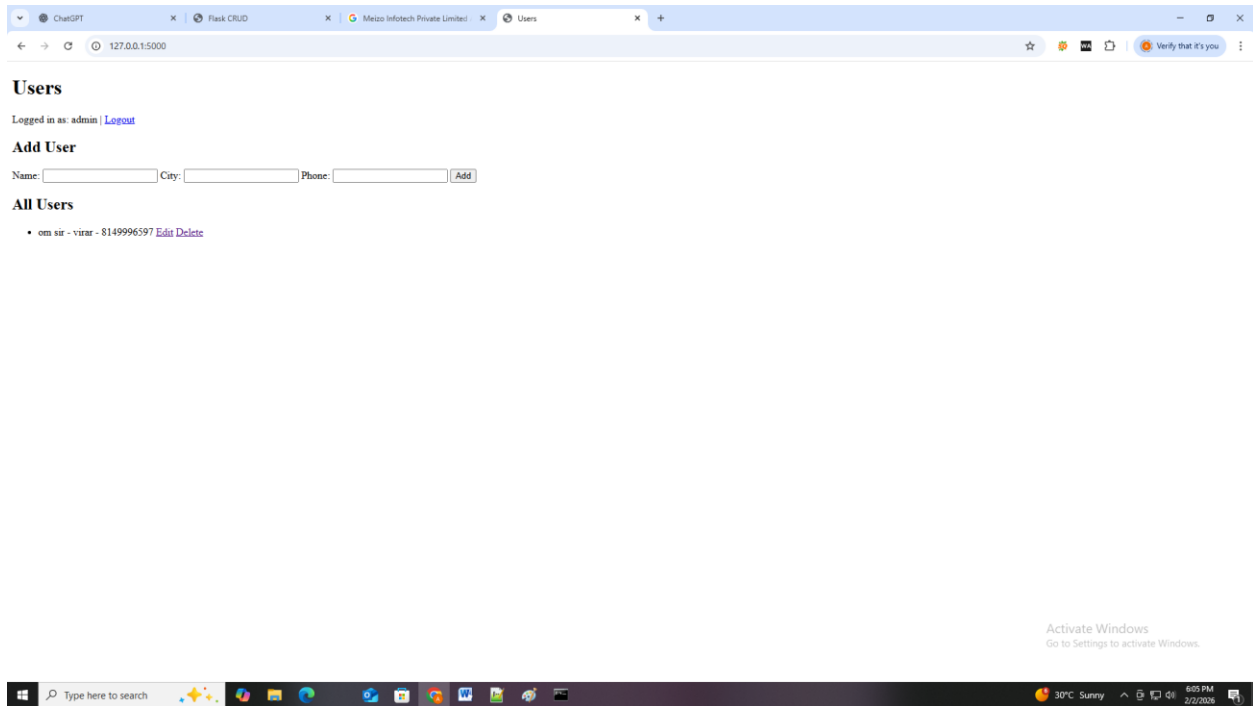


Signup and login of admin :-



After login admin can perform crud operation of add user , update , delete user record.



Run this **in your project folder** (`D:\flask-project>`):

```
pip install flask-login
```

You want to **add admin signup, login, and logout functionality** to your Flask CRUD app

Flask-project/app.py file code:-

```
from flask import Flask, render_template, request, redirect, url_for, flash
from flask_sqlalchemy import SQLAlchemy
from flask_login import LoginManager, UserMixin, login_user, login_required, logout_user, current_user
from werkzeug.security import generate_password_hash, check_password_hash

app = Flask(__name__)
app.secret_key = "supersecretkey" # Needed for sessions and flash messages

# Database config
app.config['SQLALCHEMY_DATABASE_URI'] = 'sqlite:///database.db'
app.config['SQLALCHEMY_TRACK_MODIFICATIONS'] = False

db = SQLAlchemy(app)

# Login manager setup
login_manager = LoginManager()
login_manager.init_app(app)
login_manager.login_view = "login"

# User table for CRUD
class User(db.Model):
    id = db.Column(db.Integer, primary_key=True)
```

```

name = db.Column(db.String(100), nullable=False)
city = db.Column(db.String(100), nullable=False)
phone = db.Column(db.String(15), nullable=False)

# Admin table for authentication
class Admin(UserMixin, db.Model):
    id = db.Column(db.Integer, primary_key=True)
    username = db.Column(db.String(50), unique=True, nullable=False)
    password = db.Column(db.String(200), nullable=False)

# Create database
with app.app_context():
    db.create_all()

@login_manager.user_loader
def load_user(admin_id):
    return Admin.query.get(int(admin_id))

# ----- ADMIN AUTH ROUTES -----

# Admin signup
@app.route("/signup", methods=["GET", "POST"])
def signup():
    if request.method == "POST":
        username = request.form["username"]

```

```
password = request.form["password"]
```

```
if Admin.query.filter_by(username=username).first():
```

```
    flash("Username already exists!")
```

```
    return redirect(url_for("signup"))
```

```
hashed_password = generate_password_hash(password)
```

```
new_admin = Admin(username=username, password=hashed_password)
```

```
db.session.add(new_admin)
```

```
db.session.commit()
```

```
flash("Admin account created! Please log in.")
```

```
return redirect(url_for("login"))
```

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

```
# Admin login
```

```
@app.route("/login", methods=["GET", "POST"])
```

```
def login():
```

```
    if request.method == "POST":
```

```
        username = request.form["username"]
```

```
        password = request.form["password"]
```

```

admin = Admin.query.filter_by(username=username).first()

if admin and check_password_hash(admin.password, password):
    login_user(admin)
    flash("Logged in successfully!")
    return redirect(url_for("index"))

else:
    flash("Invalid username or password!")
    return redirect(url_for("login"))

return render_template("login.html")

# Admin logout

@app.route("/logout")

@login_required

def logout():
    logout_user()
    flash("Logged out successfully!")
    return redirect(url_for("login"))

# ----- CRUD ROUTES -----

# CREATE + READ

@app.route("/", methods=["GET", "POST"])

@login_required # Only admin can access

def index():

```

```
if request.method == "POST":
    user = User(
        name=request.form["name"],
        city=request.form["city"],
        phone=request.form["phone"]
    )
    db.session.add(user)
    db.session.commit()
    flash("User added successfully!")
    return redirect(url_for("index"))

users = User.query.all()

return render_template("index.html", users=users)
```

UPDATE

```
@app.route("/update/<int:id>", methods=["GET", "POST"])
```

```
@login_required
```

```
def update(id):
```

```
    user = User.query.get_or_404(id)
```

```
if request.method == "POST":
```

```
    user.name = request.form["name"]
```

```
    user.city = request.form["city"]
```

```
    user.phone = request.form["phone"]
```

```
    db.session.commit()
```

```
flash("User updated successfully!")

return redirect(url_for("index"))

return render_template("update.html", user=user)

# DELETE

@app.route("/delete/<int:id>")
@login_required
def delete(id):

    user = User.query.get_or_404(id)

    db.session.delete(user)

    db.session.commit()

    flash("User deleted successfully!")

    return redirect(url_for("index"))

if __name__ == "__main__":

    app.run(debug=True)
```

templates/index.html file:-

```
<!DOCTYPE html>

<html>

<head><title>Users</title></head>

<body>
```

```
<h1>Users</h1>
```

```
<p>Logged in as: {{ current_user.username }} | <a href="{{ url_for('logout') }}">Logout</a></p>
```

```
<h2>Add User</h2>
```

```
<form method="POST">
```

```
  Name: <input type="text" name="name" required>
```

```
  City: <input type="text" name="city" required>
```

```
  Phone: <input type="text" name="phone" required>
```

```
  <button type="submit">Add</button>
```

```
</form>
```

```
<h2>All Users</h2>
```

```
<ul>
```

```
  {% for user in users %}
```

```
    <li>
```

```
      {{ user.name }} - {{ user.city }} - {{ user.phone }}
```

```
      <a href="{{ url_for('update', id=user.id) }}">Edit</a>
```

```
      <a href="{{ url_for('delete', id=user.id) }}">Delete</a>
```

```
    </li>
```

```
  {% endfor %}
```

```
</ul>
```

```
</body>
```

```
</html>
```

templates/update.html file code:-

```
<!DOCTYPE html>

<html>

<head><title>Update User</title></head>

<body>

  <h1>Update User</h1>

  <form method="POST">

    Name: <input type="text" name="name" value="{{ user.name }}" required><br>

    City: <input type="text" name="city" value="{{ user.city }}" required><br>

    Phone: <input type="text" name="phone" value="{{ user.phone }}" required><br>

    <button type="submit">Update</button>

  </form>

  <a href="{{ url_for('index') }}">Back</a>

</body>

</html>
```

tempaltes/signup.html file:-

```
<!DOCTYPE html>

<html>

<head><title>Signup</title></head>

<body>

  <h1>Admin Signup</h1>

  <form method="POST">

    Username: <input type="text" name="username" required><br>

    Password: <input type="password" name="password" required><br>

    <button type="submit">Sign Up</button>
```

```
</form>
<a href="{{ url_for('login') }}">Already have an account? Login</a>
</body>
</html>
```

templates/login.html file code :-

```
<!DOCTYPE html>
<html>
<head><title>Login</title></head>
<body>
  <h1>Admin Login</h1>
  <form method="POST">
    Username: <input type="text" name="username" required><br>
    Password: <input type="password" name="password" required><br>
    <button type="submit">Login</button>
```

```
</form>
```

```
<a href="{{ url_for('signup') }}">Don't have an account? Sign Up</a>
```

```
</body>
```

```
</html>
```