

ADVANCED PYTHON PROGRAMMING – CSI3007 Fall Semester (2024-2025)

Mobile App Development - Flutter

NAME: P SURIYA KUMARI

REG NO: 22MIC0181

Slot :- A2 + L7 + L8 + L27 + L28

Faculty:-Dr. Prof. Sharmila Banu K

Q) Web application using flask:

requirements.txt

Flask==2.3.2

Flask-WTF==1.1.1

Flask-SQLAlchemy==3.1.0

Flask-Migrate==4.0.4

WTForms==3.0.1

config.py

import os

BASE_DIR = os.path.abspath(os.path.dirname(__file__))

class Config:

SECRET_KEY = os.environ.get("SECRET_KEY", "dev-secret-key")

SQLALCHEMY_DATABASE_URI = os.environ.get("DATABASE_URL")
or \

"sqlite:/// " + os.path.join(BASE_DIR, "feedback.db")

SQLALCHEMY_TRACK_MODIFICATIONS = False

models.py

```
from flask_sqlalchemy import SQLAlchemy
```

```
from datetime import datetime
```

```
db = SQLAlchemy()
```

```
class Feedback(db.Model):
```

```
    id = db.Column(db.Integer, primary_key=True)
```

```
    student_name = db.Column(db.String(150), nullable=False)
```

```
    roll_number = db.Column(db.String(50), nullable=False)
```

```
    course = db.Column(db.String(100))
```

```
    rating = db.Column(db.Integer) # 1-5
```

```
    comments = db.Column(db.Text)
```

```
    created_at = db.Column(db.DateTime, default=datetime.utcnow)
```

```
    def __repr__(self):
```

```
        return f"<Feedback {self.student_name} {self.roll_number}>"
```

forms.py

```
from flask_wtf import FlaskForm
```

```
from wtforms import StringField, TextAreaField, SelectField, IntegerField,  
SubmitField
```

```
from wtforms.validators import DataRequired, Length, NumberRange
```

```
class FeedbackForm(FlaskForm):
```

```
student_name = StringField("Student name", validators=[DataRequired(),
Length(max=150)])

roll_number = StringField("Roll number", validators=[DataRequired(),
Length(max=50)])

course = StringField("Course / Subject", validators=[Length(max=100)])

rating = SelectField("Rating", choices=[('5','5'),('4','4'),('3','3'),('2','2'),('1','1')],
default='5')

comments = TextAreaField("Comments (how was the class / teacher / lab?",
validators=[Length(max=1000)])

submit = SubmitField("Submit Feedback")
```

app.py

```
from flask import Flask, render_template, redirect, url_for, flash, request
from config import Config
from models import db, Feedback
from forms import FeedbackForm
from flask_migrate import Migrate

app = Flask(__name__)
app.config.from_object(Config)

db.init_app(app)
migrate = Migrate(app, db)

@app.before_first_request
def create_tables():
    db.create_all()
```

```

@app.route('/', methods=['GET','POST'])
def index():
    form = FeedbackForm()
    if form.validate_on_submit():
        fb = Feedback(
            student_name=form.student_name.data.strip(),
            roll_number=form.roll_number.data.strip(),
            course=form.course.data.strip(),
            rating=int(form.rating.data),
            comments=form.comments.data.strip()
        )
        db.session.add(fb)
        db.session.commit()
        flash("Thank you! Your feedback has been recorded.", "success")
        return redirect(url_for('preview', fid=fb.id))
    return render_template('index.html', form=form)

```

```

@app.route('/preview/<int:fid>')
def preview(fid):
    fb = Feedback.query.get_or_404(fid)
    return render_template('preview.html', fb=fb)

```

```

@app.route('/admin')
def admin():
    # simple admin view (no auth for demo). Sort newest first.

```

```
all_fb = Feedback.query.order_by(Feedback.created_at.desc()).all()

return render_template('admin.html', feedbacks=all_fb)
```

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

templates/index.html

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width,initial-scale=1" />
    <title>Student Feedback</title>
    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">
</head>
<body>
    <div class="page">
        <header class="hero">
            <h1>Student Feedback</h1>
            <p>Share honest feedback — it helps improve classes</p>
        </header>

        <main class="container">
            <form method="POST">
                {{ form.hidden_tag() }}
                <label>Student name</label>
```

```
{{ form.student_name(class="input") }}
```

```
<label>Roll number</label>
```

```
{{ form.roll_number(class="input") }}
```

```
<label>Course / Subject</label>
```

```
{{ form.course(class="input") }}
```

```
<label>Rating</label>
```

```
{{ form.rating(class="input") }}
```

```
<label>Comments</label>
```

```
{{ form.comments(class="textarea") }}
```

```
{{ form.submit(class="btn") }}
```

```
</form>
```

```
</main>
```

```
<footer class="footer">
```

```
<a href="{{ url_for('admin') }}">Admin view</a>
```

```
</footer>
```

```
</div>
```

```
<script src="{{ url_for('static', filename='js/app.js') }}"></script>
```

```
</body>
```

```
</html>
```

templates/preview.html (this is the generated preview for the submitted feedback)

```
<!doctype html>

<html lang="en">

<head>

  <meta charset="utf-8" />

  <meta name="viewport" content="width=device-width,initial-scale=1" />


  <title>Feedback Preview</title>

  <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

</head>

<body>

  <div class="preview-card">

    <h2>Feedback Submitted  </h2>


    <div class="meta">

      <strong>Name:</strong> {{ fb.student_name }}<br>

      <strong>Roll:</strong> {{ fb.roll_number }}<br>

      <strong>Course:</strong> {{ fb.course or '—' }}<br>

      <strong>Rating:</strong> {{ fb.rating }} / 5<br>

      <strong>Submitted at:</strong> {{ fb.created_at.strftime('%Y-%m-%d %H:%M:%S') }}

    </div>


    <div class="comments">

      <h3>Comments</h3>

      <p>{{ fb.comments or 'No additional comments.' }}</p>

    </div>

  </div>

</body>

</html>
```

```
<div class="actions">

    <a class="btn" href="{{ url_for('index') }}">Submit another</a>

    <a class="btn ghost" href="{{ url_for('admin') }}">Go to admin</a>

</div>

</div>

</body>

</html>
```

templates/admin.html

```
<!doctype html>

<html lang="en">

<head>

    <meta charset="utf-8" />

    <meta name="viewport" content="width=device-width,initial-scale=1" />

    <title>Admin - Feedbacks</title>

    <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}">

</head>

<body>

    <div class="container">

        <header><h1>All Student Feedbacks</h1></header>

        <table class="feedback-table">

            <thead>

                <tr><th>#</th><th>Name</th><th>Roll</th><th>Course</th><th>Rating</th>
                <th>Comments</th><th>When</th></tr>

            </thead>
```



```

<tbody>
  {% for f in feedbacks %}
    <tr>
      <td>{{ loop.index }}</td>
      <td>{{ f.student_name }}</td>
      <td>{{ f.roll_number }}</td>
      <td>{{ f.course or '—' }}</td>
      <td>{{ f.rating }}</td>
      <td>{{ f.comments[:80] }} {% if f.comments|length > 80 %}...{% endif
%}</td>
      <td>{{ f.created_at.strftime('%Y-%m-%d %H:%M') }}</td>
    </tr>
  {% else %}
    <tr><td colspan="7" class="muted">No feedback yet.</td></tr>
  {% endfor %}
</tbody>
</table>
</div>
</body>
</html>

```

static/css/style.css

```

:root{
  --bg:#fffafc; --card:#fff; --accent:#ff6b9a; --muted:#6b6b6b;
}

body{font-family:Inter,system-ui,Arial; background:linear-
gradient(120deg,#fffafc,#f3f7ff); margin:0; padding:0;}

```

```
.page .hero {padding:28px; text-align:center;}

.container {max-width:760px;margin:20px auto;padding:20px;}

.input, .textarea, select {width:100%; padding:10px; border-radius:8px;
border:1px solid #eee; box-shadow:0 4px 18px rgba(0,0,0,0.04); margin-
bottom:12px;}

.textarea {min-height:110px; resize:vertical;}

.btn {background:var(--accent); color:white; padding:10px 18px; border-
radius:12px; border:none; cursor:pointer; text-decoration:none;}

.btn.ghost {background:transparent;color:var(--accent); border:1px solid var(--
accent); padding:8px 12px;}

.preview-card {max-width:640px;margin:30px auto;background:var(--
card);padding:22px;border-radius:16px;box-shadow:0 10px 30px
rgba(0,0,0,0.08)}

.feedback-table {width:95%; margin:20px auto;border-collapse:collapse;}

.feedback-table th, .feedback-table td {padding:10px;border-bottom:1px solid
#f1f1f1;font-size:14px;}

.muted {color:var(--muted); text-align:center;}
```

static/js/app.js

```
// minimal; you can add client-side validation or enhancements later

console.log("Student Feedback front-end loaded");
```

OUTPUT:

