

A **Student Management System** is one of the best beginner Flask projects because it covers all CRUD operations:

- Create (Add Student)
- Read (View Students)
- Update (Edit Student)
- Delete (Remove Student)
- Uses Flask
- Uses SQLite Database
- Uses HTML + CSS
- Easy to explain in interviews and lab exams

Project Title

Student Management System using Python Flask and SQLite

Project Structure

StudentManagementSystem/

```
|
|
|— app.py
|— database.py
|— students.db
|
|— templates/
|   layout.html
|   index.html
|   add_student.html
|   edit_student.html
|
|— static/
|   style.css
|
|— requirements.txt
```

Step 1: Install Flask

Open Command Prompt

pip install flask

Check installation

python -m flask --version

Step 2 : database.py

import sqlite3

def create_table():

 conn = sqlite3.connect("students.db")

 cur = conn.cursor()

 cur.execute("""

CREATE TABLE IF NOT EXISTS students(

 id INTEGER PRIMARY KEY AUTOINCREMENT,

 name TEXT NOT NULL,

 roll TEXT NOT NULL,

 department TEXT,

 email TEXT

)

""")

 conn.commit()

 conn.close()

create_table()

Run once

python database.py

A database named

students.db

will be created.

Step 3 : app.py

from flask import Flask, render_template, request, redirect

```
import sqlite3
import database

app = Flask(__name__)

# Home Page
@app.route('/')
def index():
    conn = sqlite3.connect("students.db")
    cur = conn.cursor()

    cur.execute("SELECT * FROM students")
    students = cur.fetchall()

    conn.close()

    return render_template("index.html", students=students)

# Add Student
@app.route('/add', methods=['GET', 'POST'])
def add():

    if request.method == 'POST':

        name = request.form['name']
        roll = request.form['roll']
        department = request.form['department']
        email = request.form['email']

        conn = sqlite3.connect("students.db")
        cur = conn.cursor()

        cur.execute("""
INSERT INTO students(name,roll,department,email)
```

```
VALUES(?,?,?,?)
```

```
""",(name,roll,department,email))
```

```
conn.commit()
```

```
conn.close()
```

```
return redirect('/')
```

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

```
# Edit Student
```

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

```
def edit(id):
```

```
    conn = sqlite3.connect("students.db")
```

```
    cur = conn.cursor()
```

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

```
        name = request.form['name']
```

```
        roll = request.form['roll']
```

```
        department = request.form['department']
```

```
        email = request.form['email']
```

```
        cur.execute("""
```

```
        UPDATE students
```

```
        SET name=?,roll=?,department=?,email=?
```

```
        WHERE id=?
```

```
        """,(name,roll,department,email,id))
```

```
        conn.commit()
```

```
        conn.close()
```

```
    return redirect('/')
```

```

cur.execute("SELECT * FROM students WHERE id=?", (id,))
student = cur.fetchone()

conn.close()

return render_template("edit_student.html", student=student)

# Delete Student
@app.route('/delete/<int:id>')
def delete(id):

    conn = sqlite3.connect("students.db")
    cur = conn.cursor()

    cur.execute("DELETE FROM students WHERE id=?", (id,))

    conn.commit()
    conn.close()

    return redirect('/')

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

```

Step 4 : templates/layout.html

```

<!DOCTYPE html>

<html>

<head>

```

```
<title>Student Management System</title>

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

</head>

<body>

<div class="container">

<h1>Student Management System</h1>

{% block content %}

{% endblock %}

</div>

</body>

</html>
```

Step 5 : templates/index.html

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

{% block content %}

<a href="/add" class="btn">Add Student</a>
```

<table>

<tr>

<th>ID</th>

<th>Name</th>

<th>Roll</th>

<th>Department</th>

<th>Email</th>

<th>Action</th>

</tr>

{% for s in students %}

<tr>

<td>{{s[0]}}</td>

<td>{{s[1]}}</td>

<td>{{s[2]}}</td>

<td>{{s[3]}}</td>

<td>{{s[4]}}</td>

<td>

Edit

|

Delete

</td>

</tr>

```
{% endfor %}
```

```
</table>
```

```
{% endblock %}
```

Step 6 : templates/add_student.html

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

```
{% block content %}
```

```
<h2>Add Student</h2>
```

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

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

```
<input type="text" name="roll" placeholder="Roll Number" required>
```

```
<input type="text" name="department" placeholder="Department">
```

```
<input type="email" name="email" placeholder="Email">
```

```
<input type="submit" value="Save">
```

```
</form>
```

```
{% endblock %}
```

Step 7 : templates/edit_student.html

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

```
{% block content %}
```

```
<h2>Edit Student</h2>
```

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

```
<input type="text"
```

```
name="name"
```

```
value="{{student[1]}}" required>
```

```
<input type="text"
```

```
name="roll"
```

```
value="{{student[2]}}" required>
```

```
<input type="text"
```

```
name="department"
```

```
value="{{student[3]}}">
```

```
<input type="email"
```

```
name="email"
```

```
value="{{student[4]}}">
```

```
<input type="submit" value="Update">
```

```
</form>
```

```
{% endblock %}
```

Step 8 : static/style.css

```
body{
```

```
font-family:Arial;
```

```
background:#f2f2f2;
```

```
}
```

```
.container{
```

```
width:80%;
```

```
margin:auto;
```

```
background:white;
padding:20px;
margin-top:40px;
box-shadow:0px 0px 10px gray;
}
```

```
table{
width:100%;
border-collapse:collapse;
margin-top:20px;
}
```

```
table,th,td{
border:1px solid gray;
}
```

```
th{
background:#007BFF;
color:white;
padding:10px;
}
```

```
td{
padding:10px;
text-align:center;
}
```

```
input{
width:100%;
padding:10px;
margin:10px 0;
}
```

```
.btn{
background:green;
```

```
padding:10px;
color:white;
text-decoration:none;
}
```

```
a{
text-decoration:none;
color:blue;
}
```

Step 9 : requirements.txt

Flask==3.1.0

Step 10 : Run Project

Create database

python database.py

Run Flask application

python app.py

Output

* Running on http://127.0.0.1:5000

Open your browser:

http://127.0.0.1:5000