

**Let's build your first Flask + SQLite app step by step** — slow, clean, beginner-friendly.

We'll make:

- Database connection
  - A table
  - Insert data
  - Display data on a web page
- 

## Step 1: Install required package

We'll use **Flask-SQLAlchemy** (easy + industry standard).

```
pip install flask flask-sqlalchemy
```

## Project Structure

```
flask_crud/
├── app.py
├── instance/database.db ((auto-created))
└── templates/
    ├── index.html
    └── update.html
```

### Important:

If you already ran the old app, **delete database.db** first so the new fields are created.

---

## app.py (FULL CRUD with id, name, city, phone)

```
from flask import Flask, render_template, request, redirect, url_for
from flask_sqlalchemy import SQLAlchemy
```

```
app = Flask(__name__)
```

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

```

db = SQLAlchemy(app)

# User table
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)

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

# CREATE + READ
@app.route("/", methods=["GET", "POST"])
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()
        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"])
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()
        return redirect(url_for("index"))

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

# DELETE
@app.route("/delete/<int:id>")
def delete(id):
    user = User.query.get_or_404(id)
    db.session.delete(user)
    db.session.commit()
    return redirect(url_for("index"))

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

```

---

## 📄 templates/index.html (CREATE + READ + DELETE)

```
<!DOCTYPE html>
<html>
<head>
  <title>Flask CRUD</title>
</head>
<body>

<h2>Add User</h2>

<form method="POST">
  <input type="text" name="name" placeholder="Name" required>
  <input type="text" name="city" placeholder="City" required>
  <input type="text" name="phone" placeholder="Phone" required>
  <button type="submit">Add</button>
</form>

<hr>

<h2>User List</h2>

<table border="1" cellpadding="10">
  <tr>
    <th>ID</th>
    <th>Name</th>
    <th>City</th>
    <th>Phone</th>
    <th>Actions</th>
  </tr>

  {% for user in users %}
  <tr>
    <td>{{ user.id }}</td>
    <td>{{ user.name }}</td>
    <td>{{ user.city }}</td>
    <td>{{ user.phone }}</td>
    <td>
      <a href="/update/{{ user.id }}">Edit</a> |
      <a href="/delete/{{ user.id }}" onclick="return confirm('Delete
this user?')">Delete</a>
    </td>
  </tr>
  {% endfor %}
</table>

</body>
</html>
```

---

## 📄 templates/update.html (UPDATE)

```
<!DOCTYPE html>
<html>
```

```
<head>
  <title>Update User</title>
</head>
<body>

<h2>Update User</h2>

<form method="POST">
  <input type="text" name="name" value="{{ user.name }}" required>
  <input type="text" name="city" value="{{ user.city }}" required>
  <input type="text" name="phone" value="{{ user.phone }}" required>
  <button type="submit">Update</button>
</form>

<br>
<a href="/">Back</a>

</body>
</html>
```

---

## ▶ Run the App

```
python app.py
```

Open browser   
**http://127.0.0.1:5000/**

- You now have:
- ✓ Full CRUD
- ✓ Multiple fields
- ✓ Real database app

Output:-

### Add User

### User List

ID	Name	City	Phone	Actions
2	raj sir	santacruz	9322454545	<a href="#">Edit</a>   <a href="#">Delete</a>

Activate Windows  
Go to Settings to activate Windows.