# ContactFlow – Python Contact Book API

## Step 1: Update Ubuntu System

Purpose: Ensure system packages are up to date.
Command:  `sudo apt update && sudo apt upgrade -y`

## Step 2: Install Required Tools

Purpose: Install Python, pip, venv, git.
Command:

```
 sudo apt install -y python3 python3-pip python3-venv git curl
```

## Step 3: Create Project Directory

Purpose: Organize project files.
Command:

```
 mkdir contactflow && cd contactflow
```

## Step 4: Create Virtual Environment

Purpose: Isolate project dependencies.
Command:

```
 python3 -m venv venv
 source venv/bin/activate
```

## Step 5: Install FastAPI & Uvicorn

Purpose: Backend framework & server.
Command:

```
 pip install fastapi uvicorn
```

## Step 6: Create FastAPI App

Purpose: Base API creation.
Command:

```
 nano main.py
```

### Step 7: Run FastAPI Locally

Purpose: Verify API works.
Command:

```
uvicorn main:app --host 0.0.0.0 --port 8000
```

### Step 8: Install PostgreSQL

Purpose: Database for contacts.
Command:

```
sudo apt install -y postgresql postgresql-contrib
```

### Step 9: Create Database & User

Purpose: Prepare DB.
Commands:

```
sudo -i -u postgres
psql
```

### Step 10: Create Contacts Table

Purpose: Store contact data.
SQL:

```
CREATE TABLE contacts (...)
```

### Step 11: Install psycopg2

Purpose: DB connectivity.
Command:

```
pip install psycopg2-binary
```

### Step 12: Database Connection File

Purpose: Connect API to DB.
File:
database.py

### Step 13: Pydantic Models

Purpose: Validate API data.
File:
models.py

## Step 14: CRUD API Implementation

Purpose: Contact operations.

File:

main.py

## Step 15: systemd Service

Purpose: Run API permanently.

File:

/etc/systemd/system/contactflow.service

## Step 16: Start systemd Service

Purpose: Enable auto-start.

Commands:

```
sudo systemctl start contactflow
sudo systemctl enable contactflow
```

## Step 17: Install Nginx

Purpose: Reverse proxy.

Command:

```
sudo apt install -y nginx
```

## Step 18: Configure Nginx

Purpose: Route traffic.

File:

/etc/nginx/sites-available/contactflow

## Step 19: Enable CORS

Purpose: Allow frontend access.

File:

main.py

## Step 20: Secure PostgreSQL Port

Purpose: Database security.

Commands:

```
 sudo ufw enable
 sudo ufw allow ssh
 sudo ufw allow 80
```

## Step 21: Postman Testing

Purpose: Verify API.

Endpoints:

```
 POST /contacts
 GET /contacts
 DELETE /contacts/{id}
```