Beginner-Friendly Guide to Building and Running a Flask Chatbot in Python

---

Part 1: Understanding the Raw Python Chatbot Code

We'll build a simple chatbot that responds to a few predefined financial questions.

Step 1: Write Basic Chatbot Logic Using Python

```
# Define possible questions and their answers
responses = {
    "What is the total revenue for Apple in 2024?":
        "Apple's total revenue in 2024 was $391.04 billion.",
    "How did Tesla's net income change from 2023 to 2024?":
        "Tesla's net income dropped from $14.99B in 2023 to $7.09B in 2024 — a
52.7% decrease.",
    "Which company had the highest cash flow in 2024?":
        "Microsoft had the highest operating cash flow in 2024 with $118.45
billion.",
    "How much did Microsoft's revenue grow from 2023 to 2024?":
        "Microsoft's revenue grew from $198.27B in 2023 to $245.07B in 2024 — a
23.6% increase.",
    "What are Apple's total assets in 2024?":
        "Apple's total assets in 2024 were $364.98 billion."
}

# Ask the user for input and reply accordingly
question = input("Ask a question: ")
if question in responses:
    print(responses[question])
else:
    print("Sorry, I don't understand that question.")
```

How It Works:

- You create a `responses` dictionary with predefined Q&A.
- `input()` collects user input from the terminal.
- `if-else` checks if the input matches one of the keys in the dictionary.

Part 2: Converting the Chatbot into a Web App Using Flask

Step 1: Install Flask

Open your terminal and run:

```
pip install flask
```

Step 2: Create a Python File (`app.py`)

```
from flask import Flask, render_template, request

app = Flask(__name__)

responses = {
    "What is the total revenue for Apple in 2024?":
        "Apple's total revenue in 2024 was $391.04 billion.",
    "How did Tesla's net income change from 2023 to 2024?":
        "Tesla's net income dropped from $14.99B in 2023 to $7.09B in 2024 — a
52.7% decrease.",
    "Which company had the highest cash flow in 2024?":
        "Microsoft had the highest operating cash flow in 2024 with $118.45
billion.",
    "How much did Microsoft's revenue grow from 2023 to 2024?":
        "Microsoft's revenue grew from $198.27B in 2023 to $245.07B in 2024 — a
23.6% increase.",
    "What are Apple's total assets in 2024?":
        "Apple's total assets in 2024 were $364.98 billion."
}

@app.route("/", methods=["GET", "POST"])
def index():
    answer = ""
    if request.method == "POST":
        question = request.form["question"]
        answer = responses.get(question,
"Sorry, I don't have an answer for that question.")
    return render_template("index.html", answer=answer)

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

Step 3: Create a Frontend File (`templates/index.html`)

Create a folder named `templates`, and inside it, create `index.html`:

```
<!DOCTYPE html>
<html>
<head>
    <title>Financial Chatbot</title>
</head>
<body>
    <h2>Ask the Financial Chatbot</h2>
    <form method="post">
        <input type="text" name="question" placeholder="Ask your question
here...">
        <input type="submit" value="Ask">
    </form>
    {% if answer %}
    <p><strong>Answer:</strong> {{ answer }}</p>
    {% endif %}
</body>
</html>
```

Step 4: Run the Flask App

In the terminal, navigate to the folder where `app.py` is located:

```
cd path/to/your/project
python app.py
```

Go to your browser and open:

```
http://127.0.0.1:5000
```

You'll see the chatbot interface!

Part 3: What Happens in Flask

- `@app.route("/")`: This maps the homepage URL.
- `render_template()`: Loads the HTML file.
- `request.form["question"]`: Grabs the input from the form.

Part 4: Limitations of the Chatbot

- Only responds to 5 hardcoded questions.
- Doesn't understand natural language or typos.

- No backend database or memory.
- Not scalable yet (for learning/demo purposes only).

Part 5: How to Deploy Later (Optional)

If you want to share it online:

- Use **Replit**, **Render**, or **PythonAnywhere** to host the Flask app.
- You'll need to push your code to **GitHub** if you use Render.

Let me know if you'd like a step-by-step guide for deploying too!