In [9]:
```
pip install nltk
```

```
Requirement already satisfied: nltk in ./.venv/lib/python3.13/site-package
s (3.9.1)
Requirement already satisfied: click in ./.venv/lib/python3.13/site-packag
es (from nltk) (8.2.1)
Requirement already satisfied: joblib in ./.venv/lib/python3.13/site-packa
ges (from nltk) (1.5.2)
Requirement already satisfied: regex>=2021.8.3 in ./.venv/lib/python3.13/s
ite-packages (from nltk) (2025.7.34)
Requirement already satisfied: tqdm in ./.venv/lib/python3.13/site-package
s (from nltk) (4.67.1)
Note: you may need to restart the kernel to use updated packages.
```

In [10]:
```
import nltk
from nltk.chat.util import Chat, reflections
```

In [11]:
```
pairs = [
    [
        r"(hi|hello|hey)",
        ["Hello! I am your healthcare assistant. How can I help you today
    ],
    [
        r"(.*) (fever|temperature|high temperature)",
        ["It sounds like you might have a fever. Please stay hydrated and
    ],
    [
        r"(.*) (headache|migraine)",
        ["Headaches can have many causes, including stress and dehydratio
    ],
    [
        r"(.*) (cough|cold)",
        ["A cough could be due to allergies, infection, or irritation. If
    ],
    [
        r"(.*) chest pain",
        ["Chest pain can be serious. Please seek immediate medical attent
    ],
    [
        r"i need an appointment with (.*)",
        ["Sure, I can help you schedule an appointment with %1. Could you
    ],
    [
        r"(.*) healthy diet",
        ["A healthy diet includes fruits, vegetables, whole grains, lean
    ],
    [
        r"(.*) exercise",
        ["Regular exercise (30 minutes a day) improves cardiovascular hea
    ],
    [
        r"(.*) emergency",
        ["If this is an emergency, please call your local emergency numbe
    ],
    [
        r"quit",
        ["Take care of your health. Goodbye!"]
```

```
    ]
]
```

In [12]:
```
import nltk
from nltk.chat.util import Chat, reflections
import pickle
from datetime import datetime

appointments = []

def healthcare_chatbot():
    conversation_log = []
    print("Healthcare Assistant: Hello! I am here to provide general heal
    chat = Chat(pairs, reflections)

    while True:
        user_input = input("You: ")
        conversation_log.append(("User", user_input))

        if user_input.lower() == 'quit':
            break

        response = chat.respond(user_input)
        conversation_log.append(("Assistant", response))
        print("Healthcare Assistant:", response)

        # Save appointment details if detected
        if "appointment" in user_input.lower():
            appointments.append({
                'datetime': datetime.now().strftime("%Y-%m-%d %H:%M"),
                'request': user_input,
                'response': response
            })

    # Save to binary file
    timestamp = datetime.now().strftime("%Y%m%d_%H%M%S")
    filename = f"appointments_{timestamp}.bin"
    with open(filename, 'wb') as f:
        pickle.dump({
            'conversation': conversation_log,
            'appointments': appointments
        }, f)
    print(f"\nConversation and appointments saved to {filename}")

if __name__ == "__main__":
    healthcare_chatbot()
```

```
Healthcare Assistant: Hello! I am here to provide general health guidance.
Type 'quit' to exit.

Conversation and appointments saved to appointments_20250829_153337.bin
```

In [14]:
```
def read_appointments(filename):
    with open(filename, 'rb') as f:
        data = pickle.load(f)

    print("\nSaved Appointments:")
    for apt in data['appointments']:
        print(f"Date/Time: {apt['datetime']}")
        print(f"Request: {apt['request']}")
```

```
        print(f"Response: {apt['response']}")
        print("-" * 50)
```