

### Chatbot (project-chat) code and output..

Chatbot code:

```
import random
import json
import datetime

RESPONSES_FILE = "responses.json" # Load responses from a JSON file

def load_responses():
    """ Load responses.json without this file program only reply in default responses which doesn't
    provide info """
    try:
        with open(RESPONSES_FILE, "r") as file:
            return json.load(file)
    except FileNotFoundError:
        print(f"Error: {RESPONSES_FILE} not found.")
        return {"default": ["I'm not sure how to respond to that."]}

def save_chat_log(log, user_name, agent_name):
    """ Save conversation between user and agent in separate .txt file with time """
    log_filename = f"chat_log_{datetime.datetime.now().strftime('%Y_%m_%d_%H_%M_%S')}.txt"
    with open(log_filename, "w") as file:
        for entry in log:
            file.write(f"{user_name} (User): {entry['user']}\n{agent_name} (Agent): {entry['agent']}\n\n")

def get_user_name():
    """ Ask name to the user if name doesn't given then show error """
    while True:
        user_name = input("Welcome to Poppleton University Chat! What's your name? ").strip()
        if user_name:
            return user_name.capitalize() # Capitalize the first letter of the user_name if user type
    appeal it convert to Appeal
    print("Please enter a name to continue.")

def toggle_listening_mode(current_mode):
    """ For listening mode user don't need reply from the program """
    new_mode = not current_mode
    mode_status = "ON" if new_mode else "OFF"
    print(f"Listening mode is now {mode_status}.")
    return new_mode

def handle_blank_input(blank_count):
    """ auto disconnect if user give more than 3 empty reply """
    blank_count += 1
    if blank_count >= 3:
        print("Oops! The agent has disconnected. Please try again later.")
```

```

    return blank_count, True
    return blank_count, False

def get_agent_response(user_input, responses, user_name, agent_name):
    """ Reply from agent and in responses add user_name and agent_name in reply """
    lower_input = user_input.lower()
    for keyword, replies in responses.items():
        if keyword in lower_input:
            return random.choice(replies).replace("{name}", user_name).replace("{agent_name}",
agent_name) # name in responses
    return random.choice(responses["default"]).replace("{name}", user_name) # if keywords
doesn't found than reply from default

def chatbot():
    """ This part of code for main code """
    responses = load_responses()
    agent_names = ["Steve Smith", "Rohit Sharma", "Travis Head", "Virat Kohli", "MS Dhoni"]
    random_disconnect_chance = 0.1

    user_name = get_user_name()
    agent_name = random.choice(agent_names)
    print(f"Hello, {user_name}! My name is {agent_name}. How can I assist you today?")
    print("You can type 'listening mode' to enter a mode where I will not respond to your inputs.")

    chat_log = []
    blank_input_count = 0
    listening_mode = False

    while True:
        user_input = input(f"{user_name}: ").strip()

        if user_input.lower() in ["bye", "quit", "exit"]:
            print("Thank you for chatting with us! Have a great day!")
            break

        if user_input.lower() == "listening mode":
            listening_mode = toggle_listening_mode(listening_mode)
            continue

        if listening_mode:
            print("Agent is listening...")
            chat_log.append({"user": user_input, "agent": "Agent is listening..."})
            continue

        if not user_input:
            blank_input_count, should_disconnect = handle_blank_input(blank_input_count)
            if should_disconnect:
                break

```

```

else:
    blank_input_count = 0

if random.random() < random_disconnect_chance:
    print("Oops! The agent has disconnected. Please try again later.")
    break

response = get_agent_response(user_input, responses, user_name, agent_name)
print(f"Agent ({agent_name}): {response}")
chat_log.append({"user": user_input, "agent": response})

save_chat_log(chat_log, user_name, agent_name)

if __name__ == "__main__":
    chatbot()

```

*responses.json:*

```

{
  "hello": [
    "Hi there, {name}! How can I assist you today?",
    "Hello, {name}! How's it going?",
    "Greetings, {name}! What can I do for you today?"
  ],
  "course": [
    "We offer many courses, {name}. Can you specify your area of interest?",
    "{name}, are you looking for undergraduate or postgraduate courses?",
    "Our university has courses ranging from arts to sciences, {name}. Need help deciding?"
  ],
  "undergraduate": [
    "Poppleton University offers a variety of undergraduate programs, including BSc Computing, Business Management, and Engineering.",
    "Explore all undergraduate courses at Poppleton University on our official website."
  ],
  "postgraduate": [
    "At Poppleton University, we offer postgraduate programs like MSc Computing, MBA, and Data Science.",
    "Looking to advance your career, {name}? Our postgraduate programs are designed for future leaders.",
    "Visit Poppleton University's website to explore all our postgraduate courses."
  ],
  "admissions": [
    "Admissions office hours are 10 AM to 4 PM.",
    "You can contact admissions at admissions@poppleton.ac.uk, {name}.",
    "Admissions are located in the main administration building."
  ],
}

```

```
"fees": [  
  "The tuition fee details are available on the university website.",  
  "Fee payments can be made online or at the finance office, {name}.",  
  "Late fee submissions will incur a penalty of 5%."  
],  
"fee structure": [  
  "For graduate courses, the fee is around 1 lakh 70 thousands per semester, {name}.",  
  "You can find a detailed breakdown of the fee structure on the university website, {name}.",  
],  
"scholarships": [  
  "Scholarship applications are open from January to March.",  
  "You can apply for need-based and merit-based scholarships, {name}.",  
  "Visit the scholarships office for assistance in applying."  
],  
"library": [  
  "The library opens at 9 AM every day.",  
  "Feel free to visit the library for quiet study spaces, {name}.",  
  "You can borrow books from the library."  
],  
"sports": [  
  "The sports complex is open from 6 AM to 10 PM.",  
  "You can join the football team by registering at the ECA, {name}.",  
  "Basketball courts are available."  
],  
"events": [  
  "Check the university website for upcoming events.",  
  "There's a music concert this Friday at the auditorium, {name}.",  
  "The science fair is happening next month."  
],  
"housing": [  
  "On-campus housing applications open in March.",  
  "Visit the housing office for more details, {name}.",  
  "Housing queries can be emailed to poppleton2025.ac.uk."  
],  
"exams": [  
  "Exam schedules are posted on the student portal.",  
  "Contact your department for exam-related queries, {name}.",  
  "Make sure to bring your student ID to exams."  
],  
"transport": [  
  "The campus shuttle runs every 20 minutes.",  
  "You can find the bus schedule on the university website, {name}.",  
  "Parking permits are available from the transport office."  
],  
"clubs": [  
  "The debate club meets every Wednesday at 5 PM.",  
  "You can join the chess club by signing up online, {name}.",  
  "The photography club is hosting a workshop this weekend."
```

```

],
"financial_aid": [
  "Financial aid applications can be submitted online.",
  "Visit the financial aid office for consultation, {name}.",
  "Financial aid decisions are communicated within 4 weeks."
],
"campus": [
  "The campus map is available on the university website.",
  "Feel free to explore the botanical garden near the library, {name}.",
  "Campus tours are conducted every Friday at 2 PM."
],
"support": [
  "The IT support desk is available 24/7.",
  "Counseling services are located in the health center, {name}.",
  "Academic advisors are available by appointment."
],
"joke": [
  "Why don't scientists trust atoms? Because they make up everything!",
  "Why was the math book sad? It had too many problems.",
  "What do you call fake spaghetti? An impasta!"
],
"cricket": [
  "Fun fact: {agent_name} once scored a century in T20 format!",
  "Did you know? Cricket is played in over 100 countries worldwide.",
  "{agent_name} thinks cricket is the most exciting sport ever!"
],
"coffee": [
  "The coffee bar is open from 8 AM to 8 PM.",
  "You can grab coffee at the main building cafeteria, {name}.",
  "Coffee is available near the library entrance."
],
"default": [
  "I'm sorry, {name}. I didn't quite understand that. Can you rephrase?",
  "{name}, could you provide more details? I'd be happy to help.",
  "Apologies, {name}. I didn't catch that. Can you clarify?",
  "Let me check on that for you, {name}."
]
}

```

## Output:

```
Welcome to Poppleton University Chat! What's your name? Apeal
Hello, Apeal! My name is Virat Kohli. How can I assist you today?
You can type 'listening mode' to enter a mode where I will not respond to your inputs.
Apeal: listening mode
Listening mode is now ON.
Apeal: Now user can write anything but this program doesnot going to reply
Agent is listening...
Apeal: listening mode
Listening mode is now OFF.
Apeal: Now this mode is off
Agent (Virat Kohli): I'm sorry, Apeal. I didn't quite understand that. Can you rephrase?
Apeal: ■
```

```
Welcome to Poppleton University Chat! What's your name? apeal
Hello, Apeal! My name is Virat Kohli. How can I assist you today?
You can type 'listening mode' to enter a mode where I will not respond to your inputs.
Apeal: About fees
Agent (Virat Kohli): Fee payments can be made online or at the finance office, Apeal.
Apeal: Can we get to drink coffee there?
Agent (Virat Kohli): Coffee is available near the library entrance.
Apeal: Admissions
Agent (Virat Kohli): Admissions are located in the main administration building.
Apeal: exit
Thank you for chatting with us! Have a great day!
■
```