

lessonCode3

clean_chat.py

This Python script creates an **interactive chatbot** using the **DeepSeek AI model** through the **Ollama CLI**. The chatbot allows users to enter prompts and receive AI-generated responses. A key feature of this script is that it **removes unnecessary** `<think>` tags from the AI's output to provide a cleaner response.

Code Explanation

1 Importing Required Module

```
import subprocess
```

- The `subprocess` module allows the script to run system commands, such as calling **Ollama** from the terminal.

2 Function to Run AI Model & Remove `<think>` Tags

```
def run_ollama_prompt(prompt, model="deepseek-r1:1.5b"):
    """Runs a prompt using Ollama CLI and returns the response."""
    command = ["ollama", "run", model, prompt]

    try:
        result = subprocess.run(command, capture_output=True, text=True, c
heck=True, encoding="utf-8")
        response = result.stdout.strip()

        # Remove <think> tags if present
        response = response.replace("<think>", "").replace("</think>", "").stri
p()

        return response
    except subprocess.CalledProcessError as e:
```

```
print("Error running Ollama:", e)
return None
```

Understanding the `<think>` Tag Removal

- Some AI models **internally generate thoughts** before responding. These **internal thoughts** are sometimes **enclosed in `<think>` tags** and appear in the response.
- Example of an AI response **before cleaning**:

```
<think>Analyzing the question...</think> Paris
```

- **Without removing `<think>` tags**, the output would include unnecessary text that makes it **less readable**.
- **After cleaning**, the response is simply:

```
Paris
```

- The line:

removes `<think>` and `</think>` from the response, ensuring a **clean and direct output**.

```
response = response.replace("<think>", "").replace("</think>", "").strip()
()
```

3 Running the Chatbot in a Loop

```
if __name__ == "__main__":
    while True:
        prompt = input("Enter your prompt (type 'exit' to exit): ")

        if prompt.lower() == "exit":
            print("Exiting...")
            break

        response = run_ollama_prompt(prompt)

        if response:
```

```
print("Response:", response)
else:
    print("Failed to get a response.")
```

How This Works

1. **Keeps running** in a loop until the user types `"exit"`.
2. **Takes input** from the user.
3. **Sends the input to the AI model** and receives a response.
4. **Cleans the response** by removing `<think>` tags.
5. **Prints the cleaned response**.
6. **Exits when the user types** `"exit"`.

Example Usage

Enter your prompt (type 'exit' to exit): What is the capital of France?

Response: Paris

Enter your prompt (type 'exit' to exit): Who discovered gravity?

Response: Sir Isaac Newton

Enter your prompt (type 'exit' to exit): exit

Exiting...

How to Run the Script

1. **Ensure Ollama is installed** (`ollama` should work in the terminal).
2. **Run the script:**

```
python clean_chat.py
```

3. **Start chatting** with the AI.

Now this **properly explains** why the `<think>` tag is removed, what happens if it isn't, and how the script ensures a **clean response**.

