

# Project 92: Chain-of-Thought Reasoning Demo

---

## Description:

This demo shows how large language models can solve complex problems more accurately when encouraged to “think step by step.” You’ll input a question and see how the model reasons before answering.

---

## chain\_of\_thought\_demo.py

```
import os
import openai
import gradio as gr

# Load OpenAI API key
openai.api_key = os.getenv("OPENAI_API_KEY")

# Function to generate chain-of-thought response
def chain_of_thought_reasoning(question):
    try:
        # Chain-of-thought style prompt
        prompt = (
            "You are a helpful reasoning assistant. For each question, explain\n"
            f"Question: {question}\n"
            "Let's think step by step:"
        )

        # Request from OpenAI
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo", # Works even better with GPT-4
            messages=[{"role": "user", "content": prompt}]
        )

        return response["choices"][0]["message"]["content"].strip()

    except Exception as e:
        return f"Error: {str(e)}"

# Gradio UI
```

```
iface = gr.Interface(
    fn=chain_of_thought_reasoning,
    inputs=gr.Textbox(label="Enter a Complex Question (e.g., math, logic, deci
    outputs="text",
    title="🧠 Chain-of-Thought Reasoning Demo",
    description="Ask a challenging question and the model will 'think step by
)

# Launch the demo
iface.launch()
```