Import openai

Import os

# Set up your OpenAI API key from environment variables

Openai.api\_key = os.getenv(“OPENAI\_API\_KEY”)

Def is\_coding\_question(query):

“””A basic check to see if the query is related to coding.”””

Coding\_keywords = [‘python’, ‘code’, ‘programming’, ‘debug’, ‘error’, ‘algorithm’, ‘function’, ‘syntax’]

Query\_lower = query.lower()

Return any(keyword in query\_lower for keyword in coding\_keywords)

Def get\_chatgpt\_response(query):

“””Gets a response from ChatGPT for coding-related questions.”””

If is\_coding\_question(query):

Try:

# OpenAI API request to get ChatGPT response

Response = openai.Completion.create(

Engine=”gpt-3.5-turbo”, # You can use GPT-4 if you have access

Prompt=query,

Max\_tokens=150,

N=1,

Stop=None,

Temperature=0.7,

)

# Extract the response text from OpenAI API

Answer = response.choices[0].text.strip()

Return answer

Except openai.error.AuthenticationError:

Return “Authentication failed. Please check your API key.”

Except openai.error.RateLimitError:

Return “Rate limit exceeded. Please try again later.”

Except openai.error.OpenAIError as e:

Return f”OpenAI API error: {str€}”

Except Exception as e:

Return f”Unexpected error: {str€}”

Else:

Return “I can only help with coding-related questions. Please ask a programming question.”

Def chat():

“””Main chat loop.”””

Print(“Welcome to Coding Assistance with ChatGPT!”)

Print(“Ask me anything related to coding (type ‘exit’ to end).”)

While True:

User\_input = input(“You: “)

If user\_input.lower() == ‘exit’:

Print(“Goodbye!”)

Break

Response = get\_chatgpt\_response(user\_input)

Print(f”ChatGPT: {response}”)

If \_\_name\_\_ == “\_\_main\_\_”:

# Ensure the API key is set as an environment variable before running the script.

If not openai.api\_key:

Print(“Error: OPENAI\_API\_KEY is not set. Please set your OpenAI API key.”)

Else:

Chat()