

# Project 34: Tweet Generator

---

## Description:

This tool helps generate catchy, concise, and engaging tweets based on your topic, audience, and style. Whether you're aiming to be witty, insightful, or informative—it crafts content that fits the 280-character Twitter limit.

---

## tweet\_generator.py

```
import openai
import os
import gradio as gr

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

# Function to generate a tweet
def generate_tweet(topic, tone):
    # System message sets assistant behavior
    messages = [
        {
            "role": "system",
            "content": (
                f"You are a social media expert. Write a tweet on the topic pr\n"
                "Make sure it is short, punchy, engaging, and within Twitter's
            )
        },
        {
            "role": "user",
            "content": f"Topic: {topic}"
        }
    ]

    try:
        # OpenAI API call
        response = openai.ChatCompletion.create(
            model="gpt-3.5-turbo",
            messages=messages
```

```

    )

    # Return tweet suggestion
    return response["choices"][0]["message"]["content"].strip()

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

# Gradio interface
iface = gr.Interface(
    fn=generate_tweet,                                # Core tweet generat
    inputs=[
        gr.Textbox(label="Tweet Topic (e.g. AI for Beginners)"),
        gr.Radio(["funny", "educational", "inspiring", "controversial"], label=
    ],
    outputs="text",                                    # Display tweet
    title="🐦 Tweet Generator",
    description="Enter a topic and tone, and I'll create a scroll-stopping twe
)

# Launch the Twitter content creator
iface.launch()

```