

```
from openai import OpenAI
import base64
import cv2

def get_gpt_answer(prompt: str):

    """
    This function sends a request to ChatGPT 3.5 turbo.
    Example of using this function: get_gpt_answer("Put your prompt here")
    """

    client = OpenAI()
    completion = client.chat.completions.create(
        model="gpt-3.5-turbo",
        messages=[
            {"role": "user", "content": prompt}
        ]
    )
    # Return only the response from ChatGPT 3.5 Turbo
    return completion.choices[0].message.content

def get_gpt4_answer(prompt: str):

    """
    This function sends a request to ChatGPT 4.
    Example of using this function: get_gpt4_answer("Put your prompt here")
    """

    client = OpenAI()
    completion = client.chat.completions.create(
        model="gpt-4",
        messages=[
            {"role": "user", "content": prompt}
        ]
    )
    # Return only the response from ChatGPT 4
    return completion.choices[0].message.content

def get_summary_passage(book_passage):

    """
    This function sends a request to ChatGPT 4 to summarize text.
    Example of using this function: get_summary_passage("Put your text here")
    """

    preprompt = "Summarize this into 2 sentences: "
    prompt = preprompt + book_passage
    client = OpenAI()
    completion = client.chat.completions.create(
        model="gpt-4",
        messages=[
            {"role": "user", "content": prompt}
        ]
    )
    # Return only the response from ChatGPT 4
    return(completion.choices[0].message.content)

def get_summary_text(prompt: str):

    """
    This function sends a request to ChatGPT 4.
    Example of using this function: get_summaary_text("Put your prompt here")
    """
```

```

client = OpenAI()
completion = client.chat.completions.create(
    model="gpt-4",
    messages=[
        {"role": "user", "content": prompt}
    ]
)
# Return only the response from ChatGPT 4
return(completion.choices[0].message.content)

def get_generated_image_url(finalprompt: str):
    """
    This function sends a request to DALL-E 3, generates an image, and then returns the url for
    the image.
    Example of using this function: get_gpt4_answer("Put you prompt here")
    """

    client = OpenAI()
    response = client.images.generate(
        model="dall-e-3",
        prompt=finalprompt,
        size="1024x1024",
        quality="standard",
        n=1,
    )

    # Returns the image url
    image_url = response.data[0].url
    return(image_url)

def get_full_text_from_image(image):
    """
    This function extracts text from an image using ChatGPT 4 Vision
    Example of using this function: get_full_text_from_image(put image here)
    """

    client = OpenAI()
    # Convert the image to base64
    is_success, buffer = cv2.imencode(".jpg", image)
    image_bytes = buffer.tobytes()
    base64_image = base64.b64encode(image_bytes).decode('utf-8')

    # Make the API request
    response = client.chat.completions.create(
        model="gpt-4-vision-preview", # Use the correct model name
        messages=[
            {
                "role": "user",
                "content": [
                    {
                        "type": "text",
                        "text": "Transcribe the text in this image. Only return the text
                        transcribed."
                    },
                    {
                        "type": "image_url",
                        "image_url": {
                            "url": f"data:image/jpeg;base64,{base64_image}"
                        }
                    }
                ]
            }
        ]
    )

```

```
        ],
    },
    ],
    max_tokens=2000
)

# Extract the translated text from the response
translated_text = response.choices[0].message.content
return(translated_text)
```