def main():  
 print("Expertise Level [Beginner/Intermediate/Expert]:")  
 details = []  
 while True:  
 data = input()  
 if str(data) != "Beginner, Intermediate, Expert":  
 break  
 details.append(data)  
 for detail in details:  
 print("User info", detail)  
  
 print("Experience years:")  
 years = []  
 while True:  
 data\_num = input()  
 if data\_num != str(input()):  
 break  
 years.append(data\_num)  
 for year in years:  
 print("Exp in years", year)  
  
 parser = argparse.ArgumentParser()  
 parser.add\_argument("--input", "-i", type=str, required=True)  
 arguments = parser.parse\_args()  
 user\_prompt = arguments.input  
  
 print("user\_prompt :", user\_prompt)  
 if valid\_len(user\_prompt):  
 create\_tagline(user\_prompt)  
 create\_keyword(user\_prompt)  
 else:  
 raise ValueError(  
 f"Prompt exceed the character limit. It must be under {max\_input\_length}.submitted input is{user\_prompt}")  
  
  
  
  
# function for deciding maximum valid input character length  
def valid\_len(snippet: str) -> bool:  
 return len(snippet) <= max\_input\_length  
  
  
# Load your API key from an environment variable or secret management service  
  
  
def create\_tagline(snippet, a, b):  
 openai.api\_key = os.getenv("OPENAI\_API\_KEY")  
 prompt = f" I am a creative writing{a}. I have been a professional editor for {b} years. I am going to " \  
 f"provide you following short story and you will give " \  
 f"me a critical feedback to improve my prose. You will hold me to the highest literary standards, " \  
 f"and your feedback will be open-ended and include examples or suggestions. You will also commend me " \  
 f"where I did well. Story Starts {snippet}: Story ends. \n I will now give you one or two paragraphs of " \  
 f"critical feedback to improve your prose and style "  
 response = openai.Completion.create(model="text-curie-001",  
 prompt=prompt,  
 temperature=0, max\_tokens=250)  
  
 tagline\_text: str = response["choices"][0]["text"] # extract only output text  
 tagline\_text = tagline\_text.strip() # removing white space within output text  
 end\_char = tagline\_text[-1]  
 if end\_char not in {".", "!", "?"}: # ending of the output text: Added "..." for truncate the text  
 tagline\_text += "...!"  
  
 print("Result:", tagline\_text)  
 return tagline\_text  
  
  
def create\_keyword(words: str) -> List[str]:  
 openai.api\_key = os.getenv("OPENAI\_API\_KEY")  
 prompt = f"Creative keywords network for the product {words}: "  
 print(prompt)  
  
 response = openai.Completion.create(model="text-curie-001",  
 prompt=prompt,  
 temperature=0, max\_tokens=32)  
 # print(response)  
  
 keywords: str = response["choices"][0]["text"] # extract only output text  
 keywords = keywords.strip()  
 word\_array = re.split(",| \n |;|- ", keywords)  
 word\_array = [a.lower().split() for a in word\_array]  
 word\_array = [a for a in word\_array if len(a) > 0]  
  
 print("Keywords:", word\_array)  
 return word\_array  
  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()