

해당 코드를 app.py라는 파일을 만들어 입력한 뒤 저장해주시면 됩니다

```
In [ ]: #!pip install streamlit langchain openai

import streamlit as st
import os
from dotenv import load_dotenv
load_dotenv()
OPENAI_API_KEY = os.getenv("OPENAI_API_KEY")

st.set_page_config(page_title="이메일 작성 서비스예요~", page_icon=":robot:")
st.header("이메일 작성기")

def getEmail():
    input_text = st.text_area(label="메일 입력", label_visibility='collapsed',
                             placeholder="당신의 메일은...", key="input_text")
    return input_text

input_text = getEmail()

# 이메일 변환 작업을 위한 템플릿 정의
query_template = """
메일을 작성해주세요.
아래는 이메일입니다:
이메일: {email}
"""

from langchain_classic import PromptTemplate
# PromptTemplate 인스턴스 생성
prompt = PromptTemplate(
    input_variables=["email"],
    template=query_template,
)

from langchain_classic.chat_models import ChatOpenAI
# 언어 모델을 호출합니다
def loadLanguageModel():
```

```
llm = ChatOpenAI(temperature=0.0,model_name='gpt-4.1-mini')
return llm

# 예시 이메일을 표시
st.button("*예제를 보여주세요*", type='secondary', help="봇이 작성한 메일을 확인해보세요.")
st.markdown("### 봇이 작성한 메일은:")

if input_text:
    llm = loadLanguageModel()
    # PromptTemplate 및 언어 모델을 사용하여 이메일 형식을 지정
    prompt_with_email = prompt.format(email=input_text)
    formatted_email = llm.invoke(prompt_with_email).content
    # 서식이 지정된 이메일 표시
    st.write(formatted_email)
```

만든 뒤 터미널에서 해당 폴더의 위치까지 이동한 뒤 streamlit run app.py 명령어로 streamlit을 실행합니다.