Python programming & practices

ROAD DESIGN SUPPORTER

Progress Report : 1

Date : 2023. 11. 26

Name : SeongJun, Na

ID : 192002

1. Introduction

1) Background

Modern society has achieved tremendous growth through rapid industrialization and thr birth of the fourth industry. Technologies in various fields such as transportation, IT, medicine, and science are developing day by day. It makes our lives more convenient and safe. In particular, the development of transportation and roads made the movement of resources convenient and led to active exchanges between local residents. This is one of the factors that greatly contributed to the rapid development of the industry. In addition, since human's desire for convenience is endless, the development of autonomous driving through artificial intelligence is currently underway. As time goes by, the amount of data we have increases, and it is almost impossible for us humans to memorize all these contents. That's why we keep information and use it through searches. The road and transportation sectors, like other fields, are making endless progress. Recently, as artificial intelligence and autonomous driving have begun to be introduced, new standards for roads have been proposed. In addition, many factors such as the width of the road, the radius of the curve, and the design reference speed of the road are considered in the process of designing the road. In this situation, it is inefficient for us to memorize all those vast amounts of data Because we have Python. We need programs that can easily manage vast amounts of data used in the process of designing roads. I'm going to make this through Python.

2) Project goal

Data on road design standards are analyzed and made into data. This can be referenced through simple keyword input when necessary and helps designers

understand through image search. In addition, it forms data on frequently sought content through the storage of search records and helps to use easy keywords easily. Lastly, for the easy use of useful information obtained from using the program, I produce a program that can send the contents I organized by e-mail. That is the goal of this project.

3) Differences from existing programs

Searching for documents and finding data through keywords is solved by our search engines. However, what I felt while studying civil engineering was that the specialized contents such as roads, traffic, and structures were only guidelines issued by government agencies. To utilize this, it is necessary to access the homepage, read the pdf, read major books and read standard guidelines directly. I had to find each one myself. I have never experienced a database management program focused on the road field. So I'm going to make it myself.

2. Functional Requirement

1) Design Criteria Search Engine

- Recommend highly relevant content when entering keywords on road design criteria.

2) Image Search Engine

- Improve understanding and convenience through image output such as tables and charts when entering keywords related to road design criteria.

3) Search History Retention Function

- Store search history to make it easy for users to see and use.

4) Data mail transfer function

- Provide the ability to mail user notes from the application

3. Progress

1) Implementation of features

(1) Data mail transfer function

- input & output

part_1 Get the email information from users

```
sender_email = input("Enter your email address: ")
sender_password = getpass.getpass("Enter your email password: ")
receiver_email = input("Enter the recipient's email address: ")
```

part_2 Email Server settings for Gmail

```
smtp_server = "smtp.gmail.com"
smtp_port = 587
```

part_3 write the email's subject and body

```
subject = input("Enter the subject of the email: ")
body = input("Enter the body of the email: ")
```

part_4 Set up email content using MIMEText

```
message = MIMEMultipart()
message['From'] = sender_email
message['To'] = receiver_email
message['Subject'] = subject
message.attach(MIMEText(body, 'plain'))
```

part_5 Connecting and logging in to SMTP servers

```
with smtplib.SMTP(smtp_server, smtp_port) as server:
    server.starttls()
    server.login(sender_email, sender_password)
```

part_6 Sending Email

```
server.sendmail(sender_email, receiver_email, message.as_string())
```

part_7 printing the success message

```
print("이메일이 성공적으로 전송되었습니다.")
```

- Explanation

This function is " Data mail transfer function ". Get the user's email information and get the email information that the user wants to send the message. And then get the input from users, about the mail's subject and body. After that, the mail will be send through the gmail. This program only can use the gmail. If we want to use the

another mail program, we need to change the SMTP servers. If you send a message correctly, the message "이메일이 성공적으로 전송되었습니다" will show up.

- The contents which we have learned.

import smtplib _ module import

: ' smtplib ' module

' smtplib ' module provides a client session for sending emails over a simple Mail Transfer Protocol (SMTP).

from email.mime.text import MIMEText _ using Class

: ' MIMEText ' Class

This Class is something to make the email which type is MIME.

MIME is a standard Internet protocol used to contain and transmit different types of data. You can use this class to set up email bodies, and if necessary, you can add bodies in various formats, such as HTML.

from email.mime.multipart import MIMEMultipart _ using Class

: ' MIMEMultipart ' Class

This Class is for making the multiple partial messages in type of MIME. This Class can process the devided information in two parts, subject and body.

import getpass _ module import

: ' getpass ' module

This module is something for getting the passwords safely from users. When we write the passwords in the input line, the contents which we have written don't show up.

- Code screenshot

```python
import smtplib
from email.mime.text import MIMEText
from email.mime.multipart import MIMEMultipart
import getpass

# 사용자에게 이메일 정보 입력 받기
sender_email = input("Enter your email address: ")
sender_password = getpass.getpass("Enter your email password: ")
receiver_email = input("Enter the recipient's email address: ")

# 이메일 서버 설정 (Gmail의 경우)
smtp_server = "smtp.gmail.com"
smtp_port = 587

# 이메일 내용 입력 받기
subject = input("Enter the subject of the email: ")
body = input("Enter the body of the email: ")

# MIMEText를 사용하여 이메일 내용 설정
message = MIMEMultipart()
message['From'] = sender_email
message['To'] = receiver_email
message['Subject'] = subject
message.attach(MIMEText(body, 'plain'))

# SMTP 서버에 연결 및 로그인
with smtplib.SMTP(smtp_server, smtp_port) as server:
    server.starttls()
    server.login(sender_email, sender_password)

    # 이메일 보내기
    server.sendmail(sender_email, receiver_email, message.as_string())

print("이메일이 성공적으로 전송되었습니다.")
```

2) The result of Test

(1) Data mail transfer function

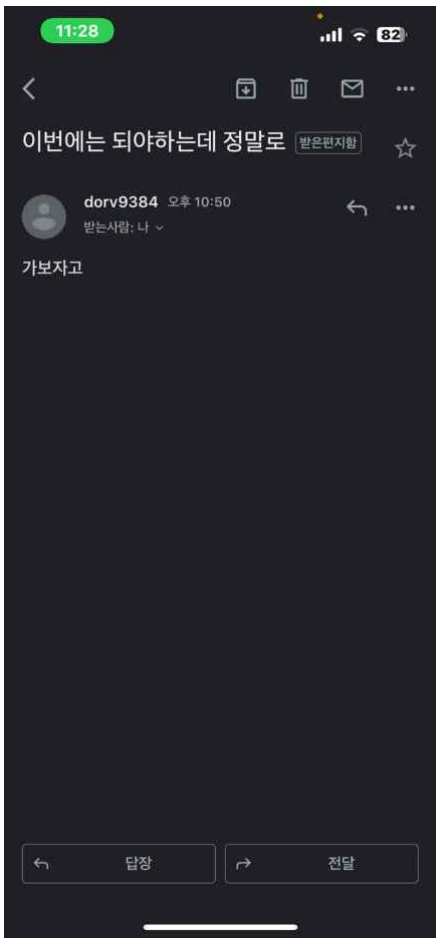
- Explanation

Login on my email and send an email to my friend.

- Screenshot of the test's result

```
(py2309-MS) C:\Users\00das\Desktop\파이썬프로그래밍\프로젝트\코드파일>C:/Users/00das/ana
conda3/envs/py2309-MS/python.exe c:/Users/00das/Desktop/파이썬프로그래밍/프로젝트/코드파
일/sending_email.py
Enter your email address: dorv9384@gmail.com
Enter your email password:
Enter the recipient's email address: deer2150@gmail.com
Enter the subject of the email: 이번에는 되야하는데 정말로
Enter the body of the email: 가보자고
이메일이 성공적으로 전송되었습니다.

(py2309-MS) C:\Users\00das\Desktop\파이썬프로그래밍\프로젝트\코드파일>
```



4. Changes in Comparison to the Plan

- None Changes

5. Schedule

업무	11/3	11/10	12/1	12/15
제안서 작성	---------->			
데이터 파일 생성	---------->			
기능 1	---------->			
기능 2	---------->			
기능 3	---------->			
기능 4	---------->			
기능 향상				---------->