

“ News Reminder ”

창의적 사고와 코딩(SW)
프로젝트 과제

첫째. News Reminder

1. “뉴스리마인더”가 무엇인가요?
2. 기획한 의도가 무엇인가요?

둘째. 개발 방법

1. 어떤 언어와 라이브러리를 사용했나요?
2. 구조는 어떻게 되나요?
3. 핵심 코드는 무엇인가요?

셋째. 프로젝트를 진행하며 ...

1. 보완하거나 추가하고 싶은 기능
2. 느낀점
3. 소스코드 및 참고자료 공유

"News Reminder"

“뉴스리마인더”가 무엇인가요?

네이버 뉴스

+

카테고리 별(정치, 사회, 경제)

+

BEST 3 뉴스 기사

=

메일로 알려주는 서비스

“뉴스리마인더”가 무엇인가요?

[2021-06-11] 네이버 뉴스 리마인더



itkoo2000@dankook.ac.kr

숨은 참조: itkoo2000에게 ▼

정치

- [민주당, 인천·시흥 등 6곳 1만호 공급...집값 6~16%면 입주](#)
- ['임대후 최초賃 분양' 1만785가구 공급](#)
- [興, 집값의 10% 공급 인천 안산 파주 등 6곳 선정](#)

생활

- [슬픈 목요일에 비가 내리면...오후부터 새벽까지 전국에 많은 비](#)
- [내일 밤부터 전국에 비...더위 한풀 꺾인다\[날씨\]](#)
- [오늘 오후부터 가장 더워...내일 밤부터 전국 비](#)

경제

- [정부, K반도체 전략 위한 예타 본격 시동](#)
- [소부장 테스트베드 구축 시동](#)
- ['K-반도체 전략' 후속 사업 내년부터 개시](#)

기획한 의도가 무엇인가요?

- 1) 사회가 어떻게 돌아가는지는 알고 싶지만 기사를 읽기는 귀찮은 사람들
- 2) 베스트 기사들만 모아서 보고 싶은 사람들
- 3) 학교, 직장내 사람들과 자연스러운 대화 OK

기획한 의도가 무엇인가요?

- 1) 사회가 어떻게 돌아가는지는 알고 싶지만
지하철, 버스 통근길에서
뉴스 기사의 핵심만
- 2) 베스트 기사들만 모아서 보고 싶은 사람들
짧고 굵게 Check!
- 3) 학교, 직장내 대화에 자연스럽게 말하고 싶은 사람들

개발방법

어떤 언어와 라이브러리를 사용했나요?



```
import requests
```

HTTP Method
GET, POST ...

```
from bs4 import BeautifulSoup
```

Html, XML 파일에서 원하는 데이터를
Parsing 할 수 있도록 도움

```
from time import sleep
```

일정시간동안 프로세스를
정지시킬 수 있도록 도움

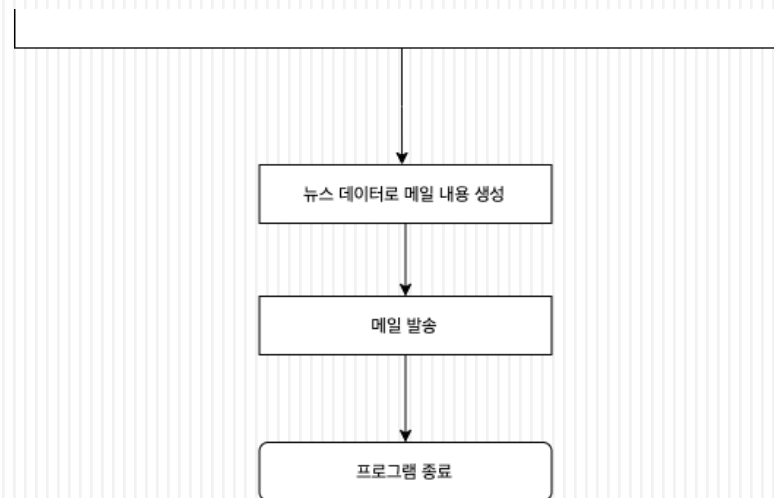
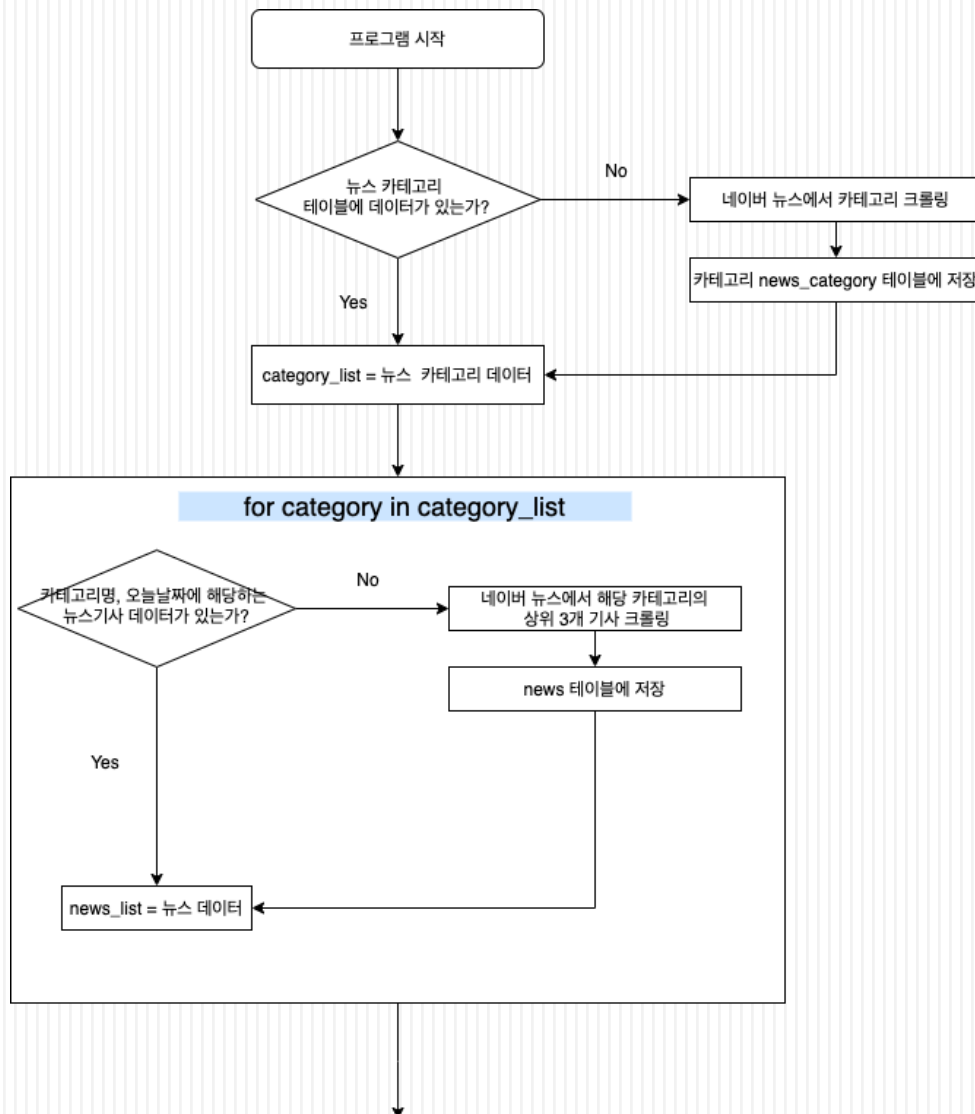
```
import pymysql
```

Mysql(DB)와 연동 할 수 있도록 도움

```
import smtplib  
from email.mime.text import MIMEText
```

이메일을 보내는데 사용하는
SMTP 관련 라이브러리

구조는 어떻게 되나요?



핵심 코드는 무엇인가요?

네이버 뉴스에서 상위 3개의 기사를 가져오는 부분

```
"""
카테고리 별 url 마다 best 3위 기사를 수집해
DB에 저장하는 함수
"""

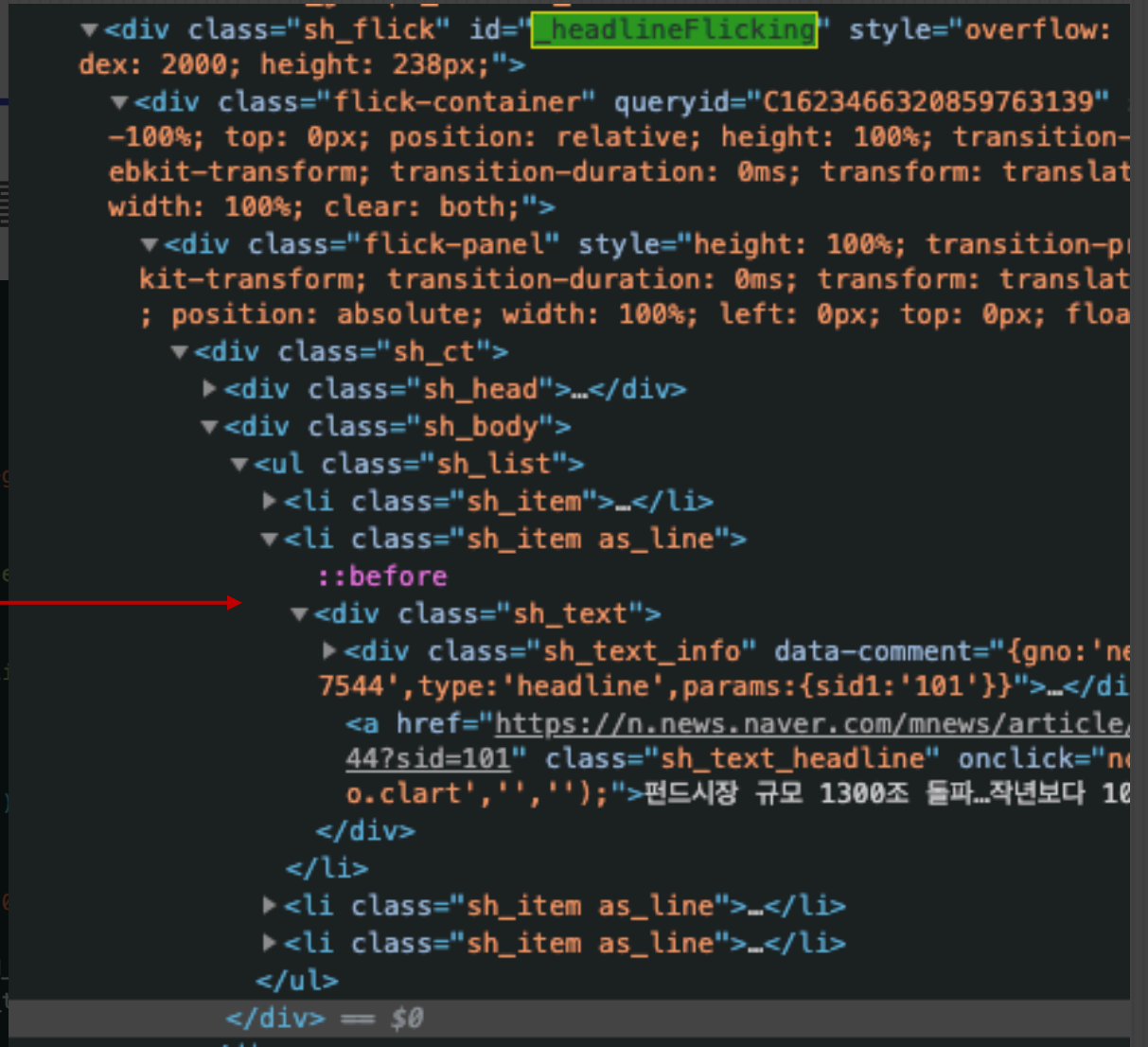
def get_news_content(self, category_name, category_url):
    response = requests.get(category_url)
    html = response.text
    parsed_html = BeautifulSoup(html, 'html.parser')
    news_list = []

    for news_data in parsed_html.select('#_headlineFlicking div div:nth-child(1) div .sh_body ul li'):
        if(len(news_data['class']) == 1):
            continue

        news_title = news_data.select('.sh_text a')[0].text
        news_title = news_title.replace('\n', '')
        news_title = news_title.replace('\\"', '')
        news_url = news_data.select('.sh_text a')[0].get('href')

        news = self.save_news(self.config.PLATFORM_NAVER,
                               category_name, news_title, news_url)
        news_list.append(news)

    return news_list
```



#_healineFlicking > div > div:nth-child(1) > div > .sh_body > ul > li

핵심 코드는 무엇인가요?

네이버 뉴스에서 상위 3개의 기사를 가져오는 부분

```
"""
카테고리 별 url 마다 best 3위 기사를 수집해
DB에 저장하는 함수
"""

def get_news_content(self, category_name, category_url):
    response = requests.get(category_url)
    html = response.text
    parsed_html = BeautifulSoup(html, 'html.parser')
    news_list = []

    for news_data in parsed_html.select('#_headlineFlicking div div:nth-child(1) div .sh_body ul li'):
        if(len(news_data['class']) == 1):
            continue

        news_title = news_data.select('.sh_text a')[0].text
        news_title = news_title.replace('\n', '')
        news_title = news_title.replace('"', '')
        news_url = news_data.select('.sh_text a')[0].get('href')

        news = self.save_news(self.config.PLATFORM_NAVER,
                               category_name, news_title, news_url)
        news_list.append(news)

    return news_list
```

핵심 코드는 무엇인가요?

네이버 뉴스 가져온 기사 데이터를 DB에 저장하는 부분

```
"""
news 객체에 데이터를 담아
DB에 insert 하는 함수
"""

def save_news(self, platform_name, category_name, news_title, news_url):
    news = News()
    news.published_date = self.util.get_current_date()
    news.platform_name = platform_name
    news.category_name = category_name
    news.title = news_title
    news.url = news_url
    news.created_at = self.util.get_current_time()
    insert: Any
    self.news_dao.insert(news)

    return news
```

핵심 코드는 무엇인가요?

News 테이블에 데이터를 저장하고, 조회하는 기능을 가진 Class

```
from db import MysqlController

class NewsDAO(MysqlController):

    def insert(self, news):
        sql = 'INSERT INTO news (published_date, platform_name, category_name, title, url, created_at) VALUES (' + \
            news.to_sql_str() + ')'

        print(sql)
        self.cursor.execute(sql)
        self.connection.commit()

    def select_by_category_name_and_published_date(self, category_name, published_date):
        sql = 'SELECT * FROM news WHERE category_name LIKE \'' + \
            category_name + '\' AND published_date like \'' + published_date + '\''

        self.cursor.execute(sql)
        result = self.cursor.fetchall()
        return result
```

핵심 코드는 무엇인가요?

News 데이터에 데이터를 저장하기 위한 클래스

	id	published_date	platform_name	category_name	title
1	43	2021-06-11	NAVER	정치	민주당, 인천·시흥 등 6곳 1만호 공급...집값
2	44	2021-06-11	NAVER	정치	'임대후 최초賃 분양' 1만785가구 공급
3	45	2021-06-11	NAVER	정치	異, 집값의 10% 공급 인천 안산 파주 등 6
4	46	2021-06-11	NAVER	생활	슬픈 목요일에 비가 내리면...오후부터 새벽까지
5	47	2021-06-11	NAVER	생활	내일 밤부터 전국에 비...더위 한풀 꺾인다[날씨]

```
print(sql)
self.cursor.execute(sql)
```

title	url	created_at
민주당, 인천·시흥 등 6곳 1만호 공급...집값 6~16%면 입주	https://n.news.naver.com/mnew...	1623346181
'임대후 최초賃 분양' 1만785가구 공급	https://n.news.naver.com/mnew...	1623346181
異, 집값의 10% 공급 인천 안산 파주 등 6곳 선정	https://n.news.naver.com/mnew...	1623346181
슬픈 목요일에 비가 내리면...오후부터 새벽까지 전국에 많은 비	https://n.news.naver.com/mnew...	1623346183
내일 밤부터 전국에 비...더위 한풀 꺾인다[날씨]	https://n.news.naver.com/mnew...	1623346183

핵심 코드는 무엇인가요?

News 테이블에 데이터를 저장하고, 조회하는 기능을 가진 Class

```
from db import MysqlController

class NewsDAO(MysqlController):

    def insert(self, news):
        sql = 'INSERT INTO news (published_date, platform_name, category_name, title, url, created_at) VALUES (' + \
            news.to_sql_str() + ')'

        print(sql)
        self.cursor.execute(sql)
        self.connection.commit()

    def select_by_category_name_and_published_date(self, category_name, published_date):
        sql = 'SELECT * FROM news WHERE category_name LIKE \'' + \
            category_name + '\' AND published_date like \'' + published_date + '\''

        self.cursor.execute(sql)
        result = self.cursor.fetchall()
        return result
```

핵심 코드는 무엇인가요?

NewsDAO 클래스가 상속받은 MysqlController Class

```
from db import MysqlController
```

생성자 > DB와의 연결

```
class NewsDAO(MysqlController):
```

```
    import pymysql
```

```
class MysqlController:
```

```
    def __init__(self, host, user, password, db_name):
```

```
        self.connection = pymysql.connect(
```

```
            host=host, port=3306, user=user, password=password, db=db_name,
```

```
            charset='utf8', autocommit=True, cursorclass=pymysql.cursors.DictCursor)
```

```
        self.cursor = self.connection.cursor()
```

```
        self.cursor.execute(sql)
```

```
        result = self.cursor.fetchall()
```

```
        return result
```

핵심 코드는 무엇인가요?

메일서버와 연결 및 메일 발송을 처리하는 기능을 하는 Class

```
import smtplib
from config import Config
from email.mime.text import MIMEText

class MailSender:
    def __init__(self, account, password):
        self.config = Config()
        self.smtp = smtplib.SMTP('smtp.gmail.com', 587)
        self.smtp.starttls()
        self.smtp.login(account, password)

    def send_message(self, to, title, content):
        message = MIMEText(content, 'html')
        message['Subject'] = title
        self.smtp.sendmail(self.config.MAIL_SENDER, to, message.as_string())
        # 세션 종료
        self.smtp.quit()
```

IMAP 허용 무엇인가요?

3. 이메일 클라이언트 구성 (예: Outlook, Eudora, Netscape Mail)

설정 방법

IMAP 액세스:

(IMAP를 사용하여 다른 클라이언트에서 Dankook University 메일에 액세스)

[자세히 알아보기](#)

상태: **IMAP를 사용할 수 있습니다.**

- ☒ IMAP 사용
☐ IMAP 사용 안함

앱비밀번호 생성

! 제3자 액세스

지금도 이 앱 비밀번호를 사용하시나요?

앱 비밀번호를 사용하면 2단계 인증을 지원하지 않는 앱 및 기기에 로그인할 수 있습니다. 이 앱 또는 기기를 더 이상 사용하지 않는 경우 Google 계정에 액세스할 수 없도록 앱 비밀번호를 삭제하세요. [자세히 알아보기](#)



Mac의 메일

6월 11일에 마지막으로 사용됨

6월 11일에 생성됨

제거

핵심 코드는 무엇인가요?

메일서버와 연결 및 메일 발송을 처리하는 기능을 하는 Class

```
import smtplib
from config import Config
from email.mime.text import MIMEText

class MailSender:
    def __init__(self, account, password):
        self.config = Config()
        self.smtp = smtplib.SMTP('smtp.gmail.com', 587)
        self.smtp.starttls()
        self.smtp.login(account, password)

    def send_message(self, to, title, content):
        message = MIMEText(content, 'html')
        message['Subject'] = title
        self.smtp.sendmail(self.config.MAIL_SENDER, to, message.as_string())
        # 세션 종료
        self.smtp.quit()
```

핵심 코드는 무엇인가요?

main.py

```
crawling = Crawling()
config = Config()
mail_sender = MailSender(config.MAIL_SENDER, config.MAIL_APP_KEY)
util = Util()
```

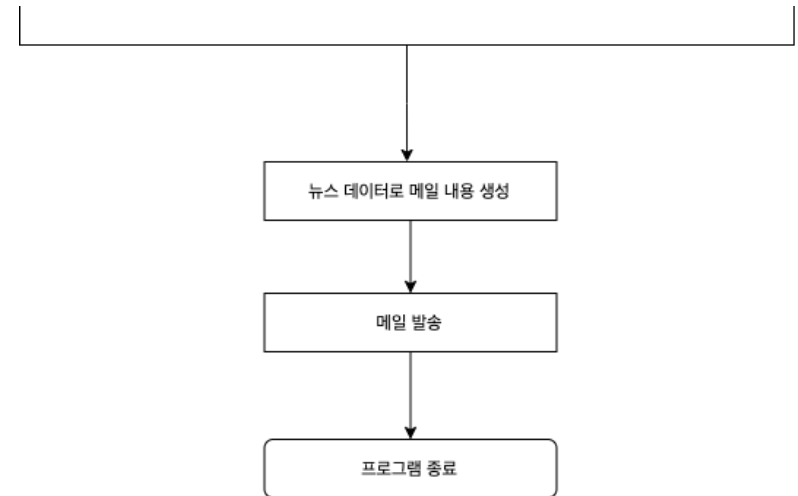
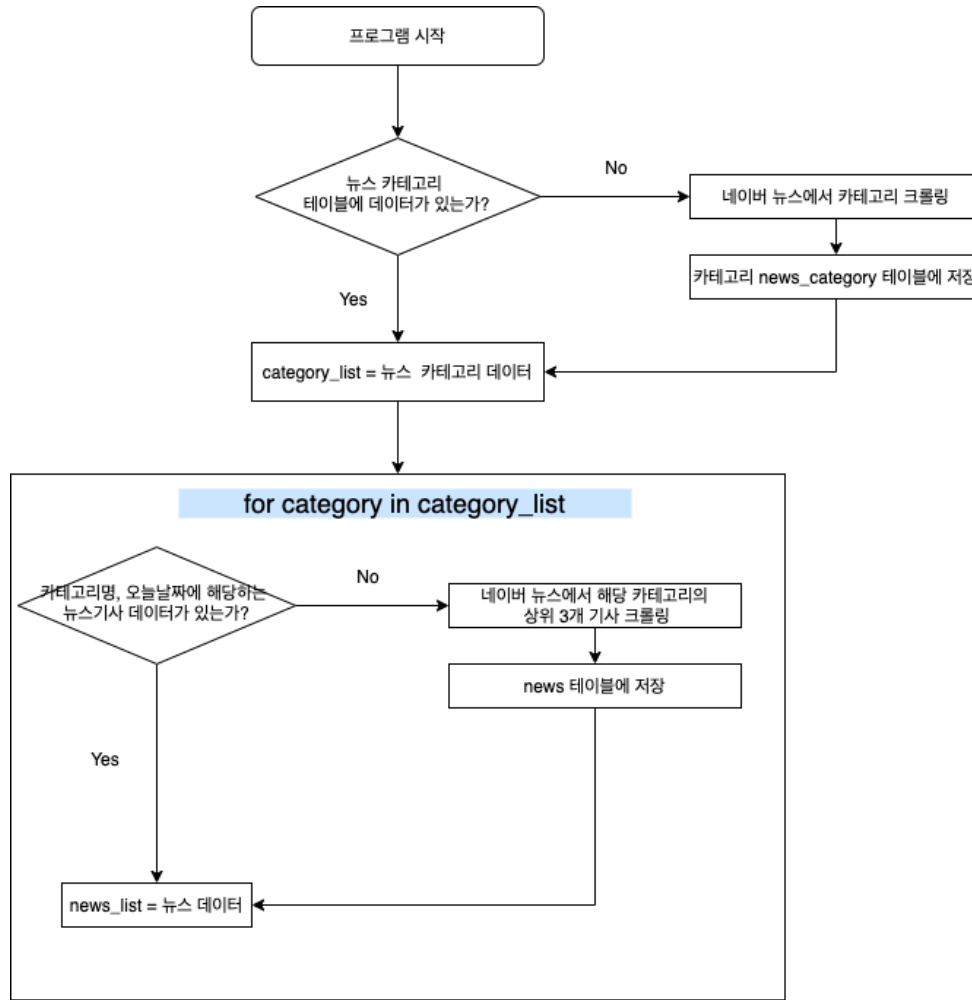
메일 제목 및 내용 선언

```
current_date = util.get_current_date()
to = 'itkoo2000@gmail.com' # 메일을 보낼 사람
mail_title = '[' + current_date + '] 네이버 뉴스 리마인더'
mail_content = ''
```

1. 뉴스 카테고리 url이 있는지 체크, 있다면 조회 없다면 크롤링해서 저장

```
is_use = 1
news_category_list = crawling.news_category_dao.select_by_is_use(is_use)
if(len(news_category_list) == 0):
    print('뉴스 카테고리 데이터가 존재하지않습니다 ... 새로 데이터를 수집합니다 ...')
    news_category_list = crawling.get_category_url()
else:
    print('뉴스 카테고리 데이터가 존재합니다 ... ')
```

핵심 코드는 무엇인가요?








print('뉴스 카테고리 데이터가 존재합니다 ... ')

핵심 코드는 무엇인가요?

main.py

```
crawling = Crawling()
```

```
config = Config()
```

	 id	 name	 url	 is_use	 created_at
1	91	정치	https://m.news.naver.com/main.nhn?mode=LSD&sid1=100	1	1623343256
2	92	생활	https://m.news.naver.com/main.nhn?mode=LSD&sid1=103	1	1623343256
3	94	경제	https://m.news.naver.com/main.nhn?mode=LSD&sid1=101	1	1623343256
4	95	IT	https://m.news.naver.com/main.nhn?mode=LSD&sid1=105	1	1623343256
5	96	사회	https://m.news.naver.com/main.nhn?mode=LSD&sid1=102	1	1623343256
6	97	세계	https://m.news.naver.com/main.nhn?mode=LSD&sid1=104	1	1623343256
7	90	뉴스홈	https://m.news.naver.com/home.nhn	0	1623343256
8	93	많이본	https://m.news.naver.com/rankingList.nhn	0	1623343256

```
# 1. 뉴스 카테고리 url이 있는지 체크, 있다면 조회 없다면 크롤링해서 저장
```

```
is_use = 1
```

```
news_category_list = crawling.news_category_dao.select_by_is_use(is_use)
```

```
if(len(news_category_list) == 0):
```

```
    print('뉴스 카테고리 데이터가 존재하지않습니다 ... 새로 데이터를 수집합니다 ...')
```

```
    news_category_list = crawling.get_category_url()
```

```
else:
```

```
    print('뉴스 카테고리 데이터가 존재합니다 ... ')
```


핵심 코드는 무엇인가요?

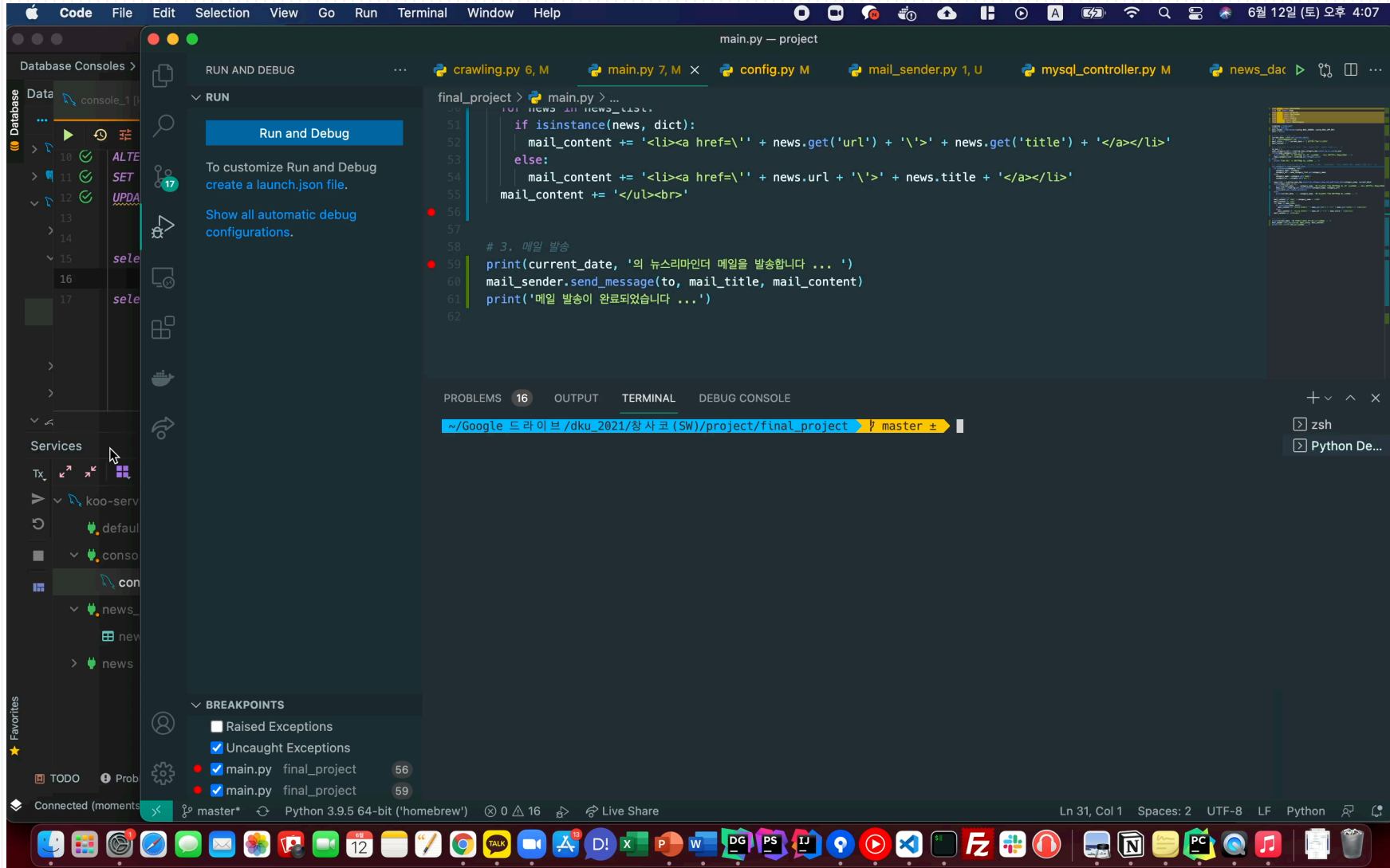
main.py

```
# 2. 뉴스 카테고리 url로 각 카테고리 별 best 3 기사 저장되어있는지 체크, 있다면 조회, 없다면 크롤링해서 저장
for category in news_category_list:
    mail_content += '<h2>' + category.get('name') + '</h2>'
    category_name = category.get('name')

    news_list = crawling.news_dao.select_by_category_name_and_published_date(category_name, current_date)
    mail_content += '<ul>'
    if(len(news_list) == 0):
        print(current_date, ',', category_name, '에 해당하는 뉴스 데이터가 존재하지않습니다 ... 새로 데이터를 수집합니다 ...')
        news_list = crawling.get_news_content(category_name, category.get('url'))
        # 메일 내용 생성
        for news in news_list:
            mail_content += '<li><a href=\"' + news.url + '\">' + news.title + '</a></li>'
            sleep(util.get_random_seconds())
    else:
        print(current_date, ',', category_name, '에 해당하는 뉴스 데이터가 존재합니다 ... ')
        # 메일 내용 생성
        for news in news_list:
            mail_content += '<li><a href=\"' + news.get('url') + '\">' + news.get('title') + '</a></li>'
    mail_content += '</ul><br>'

# 3. 메일 발송
print(current_date, '의 뉴스리마인더 메일을 발송합니다 ... ')
mail_sender.send_message(to, mail_title, mail_content)
print('메일 발송이 완료되었습니다 ...')
```

실행영상



프로젝트를 진행하며 ...

보완하고 싶은 점

1) 파이썬의 Class 등 배우지 않은 부분을 사용한 부분 > 매끄럽지 못함

2) DB 생성

```
create table news_category
(
    id          int auto_increment
        primary key,
    name        varchar(50)      null,
    is_use      tinyint(1) default 1 null,
    url         text              null,
    created_at  int              null
);
```

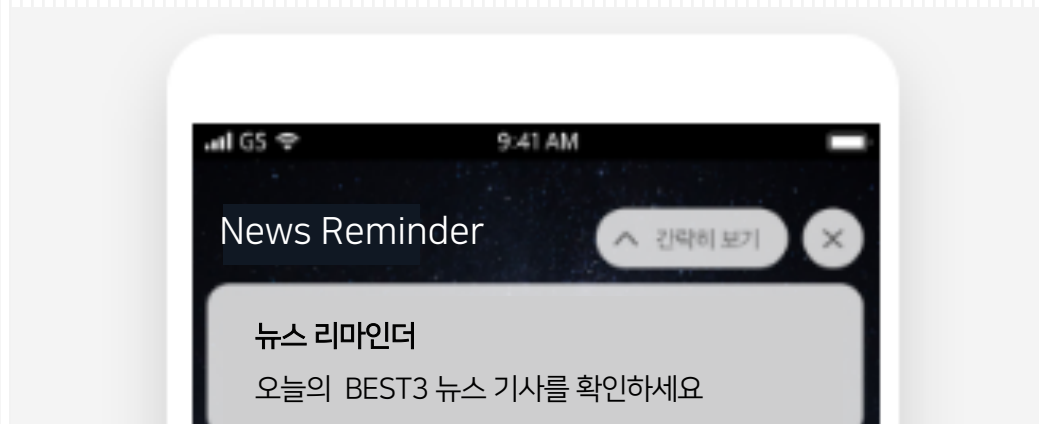
```
create table news
(
    id          int auto_increment
        primary key,
    published_date date          not null,
    platform_name varchar(50)    null,
    category_name varchar(50)    not null,
    title       text             null,
    url         text             null,
    created_at  int              null
);
```

추가하고 싶은 기능

1) 언론사별 기사 추가 크롤링 (다양한 데이터)



2) 메일 형식이 아닌 앱 형식의 푸시알람



파이썬 > 간결 & 지원 함수 다양

Class 및 상속 공부 하고싶다 ...

자유 프로젝트이다 보니 욕심이 생긴다 ...

처음에 크롤링 후 콘솔에 보여주는 기획

-> 데이터 저장 위해 DB 연동

-> 콘솔 말고 메일 발송

소스코드 공유

Github

https://github.com/ITKOO/news_reminder

ITKOO model 폴더 db 폴더 밑으로 경로 변경

d385971 13 hours ago 3 commits

__pycache__	네이버 뉴스 크롤링 기능 개발	22 hours ago
config	model 폴더 db 폴더 밑으로 경로 변경	13 hours ago
db	model 폴더 db 폴더 밑으로 경로 변경	13 hours ago
module	model 폴더 db 폴더 밑으로 경로 변경	13 hours ago
README.md	Create README.md	22 hours ago
main.py	네이버 뉴스 크롤링 기능 개발	22 hours ago

README.md

News Reminder

단국대학교 1학년 1학기
창의적사고와 코딩(SW) 수업 개인 프로젝트

- 실행방법

1. config 폴더안에 있는 config.py 파일에서 아래의 내용을 자신의 환경에 맞게 수정합니다.

```
self.DB_HOST = 'host'
```

참고자료 공유

크롤링 관련 > BeautifulSoup 공식문서

<https://www.crummy.com/software/BeautifulSoup/bs4/doc/>

크롤링 관련 > BeautifulSoup 예제

<https://wikidocs.net/85739>

SMTP 메일 발송 관련 예제 블로그

<https://yeolco.tistory.com/93>

감사합니다.

itkoo2000@dankook.ac.kr