**수집 및 업무 관련 Tool   
사용 매뉴얼**

컨텐츠본부 컨텐츠기획팀

설준영

2022.01

변경내역

|  |  |  |
| --- | --- | --- |
| 일시 | 내역 | 작성자 |
| 2022-01-14 | 최초 작성 | 설준영 |
| 2022-01-19 | 수집 관련 Python CODE 수정 방법 및 예시 내용 추가 | 설준영 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

목 차

[1. 개요 4](#_Toc93509014)

[2. 지식 공방 수집 툴 4](#_Toc93509015)

[2.1 수집 툴 위치 경로 4](#_Toc93509016)

[2.2 내용 5](#_Toc93509017)

[2.3 유지 보수 관련 5](#_Toc93509018)

[3. 구축1팀 관련 수집 툴 5](#_Toc93509019)

[3.1 포털 수집(폴더명) 5](#_Toc93509020)

[3.1.2 네이버 5](#_Toc93509021)

[3.1.3 다음 6](#_Toc93509022)

[3.1.4 다음TV맛집수집 6](#_Toc93509023)

[3.1.5 다음주차장 7](#_Toc93509024)

[3.2 02\_지식공방\_신규(폴더명) 7](#_Toc93509025)

[3.3 03\_매경 100프랜\_3.x 폴더명) 7](#_Toc93509026)

[3.4 04\_트럭서비스센터 폴더명) 7](#_Toc93509027)

[4. 기획팀 관련 수집 툴 7](#_Toc93509028)

[5. 업무 관련 툴 8](#_Toc93509029)

[5.1 뉴스 크롤링(뉴스기사수집툴.py) 8](#_Toc93509030)

[5.1.1 개발 목적 8](#_Toc93509031)

[5.1.2 사용법 8](#_Toc93509032)

[5.1.3 기타 정보 8](#_Toc93509033)

[5.2 배포 파일 비교툴 (파일비교툴.py) 9](#_Toc93509034)

[5.2.1 개발 목적 9](#_Toc93509035)

[5.2.2 사용법 9](#_Toc93509036)

[5.2.3 기타 정보 9](#_Toc93509037)

[5.3 수집 툴 통합 관리 (수집툴통합관리.py) 9](#_Toc93509038)

[5.3.1 개발 목적 9](#_Toc93509039)

[5.3.2 사용법 9](#_Toc93509040)

[5.3.3 기타 정보 10](#_Toc93509041)

[5.4 CMMS (0\_CMMS\_ADMIN.py, 1\_CMMS\_USER.py, 2\_cmms\_list\_DB\_config.py, 3\_DB\_daily\_update.py) 11](#_Toc93509042)

[5.4.1 개발 목적 11](#_Toc93509043)

[5.4.2 사용법(Admin 버전 기준) 11](#_Toc93509044)

[5.4.3 Admin 버전 추가 기능 11](#_Toc93509045)

[5.4.4 개발 환경 및 접속 정보 11](#_Toc93509046)

[5.4.5 3\_CMMS\_list\_DB\_config.py 12](#_Toc93509047)

[5.4.6 2\_DB\_daily\_update.py 12](#_Toc93509048)

[5.4.7 기타 정보 12](#_Toc93509049)

[6. 수집 관련 Python CODE 수정 방법 및 예시 12](#_Toc93509050)

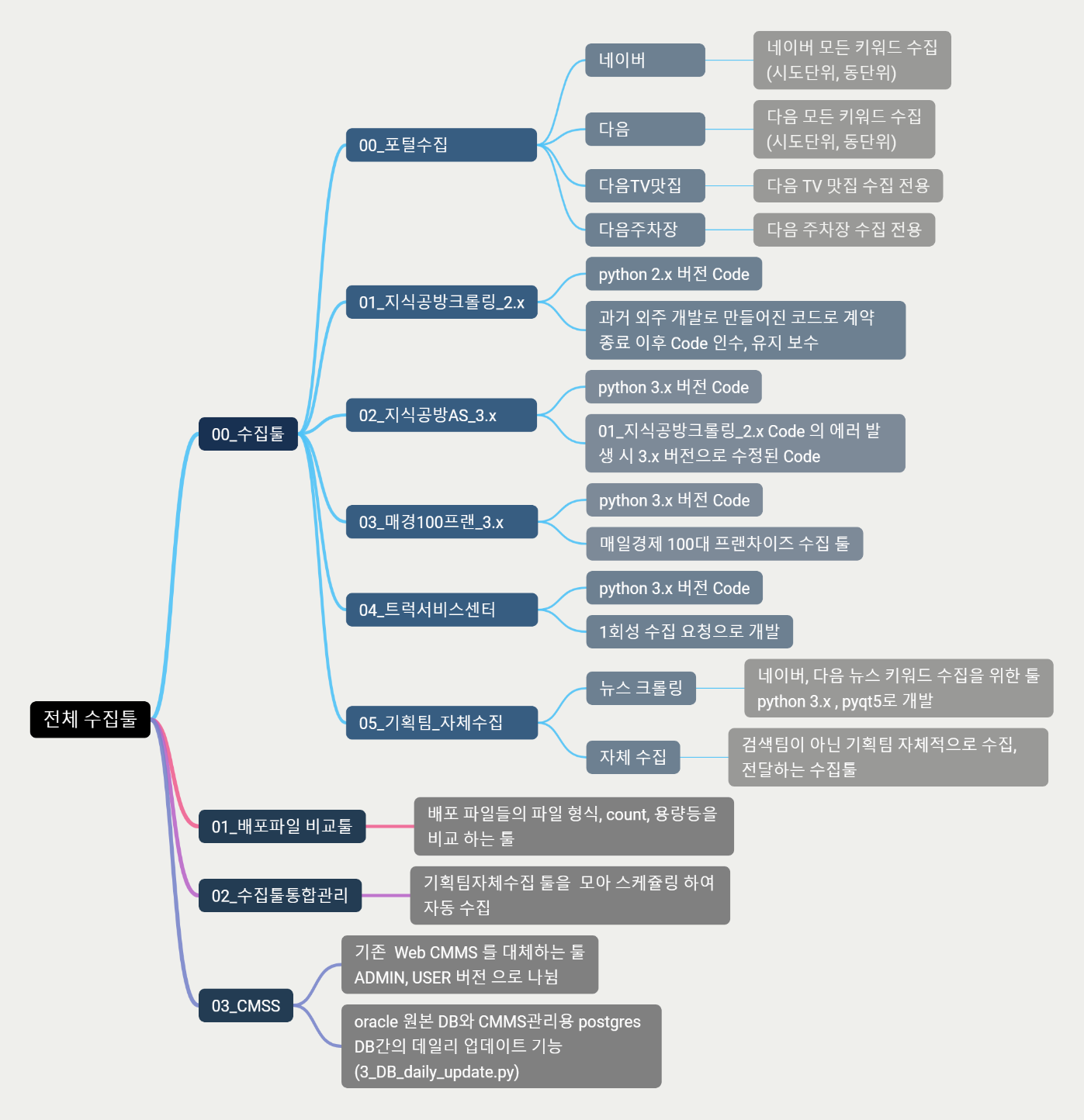
[6.1 일반적인 수집 Code 구조 12](#_Toc93509051)

[6.1.1 지식 공방 수집 형태 예시 12](#_Toc93509052)

[6.1.2 통합관리버전 형태 12](#_Toc93509053)

[6.2 대표적인 오류 발생 케이스와 해결 방법 13](#_Toc93509054)

[6.3 태그나 속성 값이 변한 경우 수정 방법. 14](#_Toc93509055)

1. 개요
   * + - * 지식 공방 수집 툴 + 기획 팀 내부 수집 툴 + 기타 업무 관련 툴의 사용 방법과 주의 사항 등에 관한 안내  
           를 위해 만들어진 문서
2. 지식 공방 수집 툴
   1. 수집 툴 위치 경로
      * + - [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\01\_지식공방크롤링\_2.x\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\01_지식공방크롤링_2.x\)
          - 전체 폴더에 대한 개요  
            
   2. 내용
      * + - 과거 외주(지식공방) 를 통해 개발된 수집 툴로 Python 2.7버전 이상의 환경에서만 작동 가능하다.
          - 현재 수집 1팀의 서버 192.168.11.61 에 Anaconda 2.7 버전환경 구성이 되어 있음.

원격 접속 정보 : search / 검색 (C:\Anaconda27)

* + - * + Python Code 부분을 제외한 수집, 운용 에 관한 부분은 구축 1팀에서 관리 중(방현아M)
  1. 유지 보수 관련
     + - * Code 오류 내용이 단순 태그 변경정도 일 경우 2.x버전을 직접 수정 하여 재사용   
           이외의 경우 3.7버전 이상의 Code 로 새로 개발하여 사용한다.  
           ([\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\02\_지식공방AS\_3.x\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\02_지식공방AS_3.x\))

1. 구축1팀 관련 수집 툴
   * + - * 지식 공방 계약 종료 이후 새로운 수집 소스에 대한 개발이 필요할 때 개발된 수집 툴
         * 수집 툴의 사용은 구축1팀
         * 각 폴더 별로 수집 툴에 대해 설명
         * Python Version = 3.7
   1. 포털 수집(폴더명)
      * + - Naver 와 Daum 포털을 대상으로 하는 수집 툴
      1. 네이버
         1. 00\_NAVER\_CRAWLING\_DONG.py
            * 인풋 파일의 검색어 리스트를 동단위 리스트 만큼 반복하여 수집
            * Naver검색 리턴 카운트 가 300개가 최대이기에 동단위로 쪼개서 수집하기 위함

Input = ‘inputlist.txt’ / ANSI

Output = ‘Crawl\_result\_dong.txt’ / UTF-8

Reference = ‘sidolist\_dong.txt’

배포 = exe

* + - 1. 00\_NAVER\_CRAWLING\_GU.py
         * 인풋 파일의 검색어 리스트를 구단위 리스트 만큼 반복하여 수집
         * Naver검색 리턴 카운트 가 300개가 최대이기에 동단위로 쪼개서 수집하기 위함

Input = ‘inputlist.txt’ / ANSI

Output = ‘Crawl\_result\_gu.txt’ / UTF-8

Reference = ‘sidolist\_gu.txt’

배포 = exe

* + - 1. 01\_네이버폐업체크.py
         * 인풋 파일의 주소+검색어 리스트를 네이버로 검색하여 검색결과의 존재 유무를 반환
         * 검색 결과가 다수일 경우 첫번째 결과만 리턴

Input = ‘폐업check리스트.txt’ / ANSI

Output = ‘네이버폐업확인결과.txt’ / UTF-8

배포 = exe

* + 1. 다음
       1. 00\_daum\_Crawling.py
          - 인풋 파일의 검색어 리스트를 동단위 리스트 만큼 반복하여 수집

Input = ‘inputlist.txt’ / ANSI

Output = ‘daum\_crawling\_result.txt’ / UTF-8

배포 = exe

* + - 1. 01\_DAUM\_상세정보\_수집전용.py
         * 인풋 파일의 검색 리스트의 다음 상세 정보 페이지를 수집
         * 수집 과정 : 인풋 파일 검색 리스트의 ID수집 -> ID의 상세정보 페이지 호출 -> txt 저장.

Input = ‘daum\_search\_list.txt’ / ANSI / ‘ 주소 , 명칭 ‘ 형식

Output = ‘daum\_crawling\_result.txt’ / UTF-8

배포 = exe

* + - 1. 02\_다음폐업체크API.py
         * 인풋 파일의 검색 리스트의 다음 상세 정보 페이지를 수집
         * 수집 과정 : 인풋 파일 검색 리스트의 ID수집 -> ID의 상세정보 페이지 호출 -> txt 저장.

Input = ‘daum\_search\_list.txt’ / ANSI / ‘ 주소 , 명칭 ‘ 형식

Output = ‘daum\_crawling\_result.txt’ / UTF-8

배포 = exe

* + 1. 다음TV맛집수집
       1. 01\_다음TV맛집\_ALL.py
          - ‘ TV 맛집 ‘ 명칭을 동별로 다음에서 검색하여 TV방영 결과가 있는 경우 상세 정보 페이지 호출, 해당 페이지의 정보를 수집.

Output = ‘다음TV맛집\_전체\_수집결과.txt’ / UTF-8

Reference 1= ‘daum\_tv\_sidolist.txt’ / ANSI

Reference 2= ‘daum\_tv맛집\_list\_중복제거.txt’ / ANSI / 중간 생성 파일

배포 = exe

* + - 1. 02\_다음TV맛집\_DETAIL\_ONLY.py
         * 01\_다음TV맛집\_ALL.py 에서 수집된 중간 파일 ‘daum\_tv맛집\_list.txt’ 을 사용하여 상세 정보 페이지만을 수집

Output = ‘다음TV맛집\_전체\_수집결과.txt’ / UTF-8

Reference 1= ‘daum\_tv\_sidolist.txt’ / ANSI

Reference 2= ‘daum\_tv맛집\_list\_중복제거.txt’ / ANSI

배포 = exe

* + 1. 다음주차장
       1. 다음주차장수집.py
          - 다음의 ‘주차장’ 검색어를 동별로 수집, 해당 ID의 주차장 상세정보를 수집

Output = ‘parkingLot.txt’ / UTF-8

Reference 1= ‘sidolist\_dong.txt’ / ANSI

Reference 2= ‘ID\_list\_중복제거.txt’ / ANSI / 중간 생성 파일

배포 = exe

* 1. 02\_지식공방\_신규(폴더명)
     + - * 지식 공방 계약 종류 이후 구축1팀의 요청에 의해 신규 개발
         * 배포 = py
         * Python Version = 3.7
  2. 03\_매경 100프랜\_3.x 폴더명)
     + - * 매경 100대 프랜차이즈 리스트에 따라 신규 개발
         * 배포 = py
         * Python Version = 3.7
  3. 04\_트럭서비스센터 폴더명)
     + - * 트럭 관련 서비스센터 수집
         * 배포 = py
         * Python Version = 3.7

1. 기획팀 관련 수집 툴
   * + - * 기획팀 에서 수집하여 전달하는 리스트에 관련된 수집툴([\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\05\_기획팀\_자체수집\자체수집\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\05_기획팀_자체수집\자체수집\))
         * 이후 설명 될 업무 관련 툴 중 ‘ 02\_수집툴통합관리 ‘ 항목에서 추가 설명.
         * 배포 = exe
         * Python Version = 3.7
2. 업무 관련 툴
   * + - * 내부 업무의 편리, 효율 및 기존 기능 대체를 위해 개발된 일련의 업무 관련 프로그램.
   1. 뉴스 크롤링(뉴스기사수집툴.py)
      1. 개발 목적
         * + 특정 키워드의 네이버, 다음 뉴스 검색 결과를 수집.
           + [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\05\_기획팀\_자체수집\뉴스크롤링\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\05_기획팀_자체수집\뉴스크롤링\)
      2. 사용법
         1. 시작 날짜와 종료 날짜를 형식에 맞게 입력
         2. 호출 페이지 설정 칸에 하고자 하는 페이지 개수 입력(숫자만)
         3. VIEW페이지까지 수집이 필요한 경우 체크박스 란 체크 -> VIEW 수집 Sheet 가 추가되어 수집
         4. 수집 버튼 클릭.

텍스트이(가) 표시된 사진

자동 생성된 설명

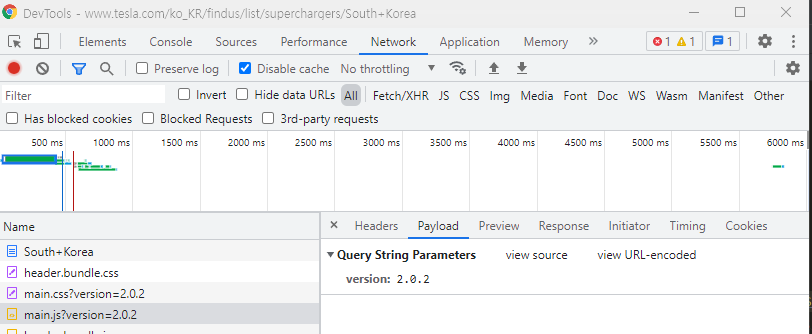
* + 1. 기타 정보
       - * Input = ‘ Crawling\_keywords.txt ‘ / ANSI
         * Output = ‘ 뉴스기사수집결과\_중복제거.xlsx ‘
         * Reference = ‘ 뉴스기사수집결과.xlsx ‘ / 중간 결과
         * Version = Python 3.7
         * Main Library = Pyqt5
         * 배포 = exe
  1. 배포 파일 비교툴 (파일비교툴.py)
     1. 개발 목적
        + - 배포 파일 폴더 간의 변화 값들을 비교하여 Excel 파일로 저장.
          - [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\01\_배포파일비교툴\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\01_배포파일비교툴\)
     2. 사용법
        1. 기준 경로에 변화 기준 폴더 경로 입력.
        2. 비교하고자 하는 폴더를 비교 경로에 입력.
        3. 시작 버튼 클릭
           + <그림. 1>

텍스트이(가) 표시된 사진

자동 생성된 설명

* + 1. 기타 정보
       - * Input = 설정 경로
         * Output = 배포데이터검수결과\_{‘실행 날짜’}.xlsx
         * Reference = ‘ 기준\_파일리스트.txt ‘ , ‘ 비교\_파일리스트.txt ‘ / 중간 결과
         * Version = Python 3.7
         * Main Library = Pyqt5
         * 배포 = exe
  1. 수집 툴 통합 관리 (수집툴통합관리.py)
     1. 개발 목적
        + - 기획팀 내부 수집 전달을 목적으로 만들어진 수집 툴을 일괄로 수집 혹은 스케쥴링 하여 수집하기 위함.
          - [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\02\_수집툴통합관리\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\02_수집툴통합관리\)|
     2. 사용법
        1. .수집 툴 불러오기 버튼을 통해 수집하고자 하는 exe 수집툴(여러 개 가능) 선택 참조<그림. 2>
           + <그림. 2>
           + 텍스트이(가) 표시된 사진

             자동 생성된 설명
        2. 불러온 리스트의 수집 주기 수정이 필요할 경우 해당 리스트 클릭, 상위 탭에서 원하는 정보대로 수정한 후 설정 저장 버튼 클릭. 참조<그림. 2>
        3. 수집 주기 설정은 툴이 위치한 경로의 ‘ crawl\_schedule\_config.txt ‘ 파일을 열어서 직접 수정 가능.
        4. 텍스트에 예약된 정보대로 수집이 필요한 경우 **‘ 예약 실행 ‘** 버튼 클릭
        5. 예약된 정보가 아닌 현재 시점에서 일괄 수집이 필요한 경우 **‘ 시 작 ‘** 버튼 클릭
        6. 불러온 리스트를 초기화 하려면 **‘ 리스트 초기화 ‘** 버튼 클릭.
     3. 기타 정보
        + - Input = ‘ crawl\_schedule\_config.txt ‘ / UTF-8
          - Output = //수집결과//{각 수집툴폴더명칭}//{각 수집결과 파일명칭}
          - Reference = ‘ exe\_list.txt ‘ / 수집 툴 불러오기 버튼을 통해 불러온 리스트가 저장되어 있는 파일.  
            추후 이 파일의 리스트를 가지고 실행 파일의 경로를 읽어 옴.
          - Version = Python 3.7
          - Main Library = Pyqt5
          - 배포 = exe
          - 주의 사항 = 설정 파일에 저장된 주기 설정 값의 범주를 넘어갈 경우 해당 주기에 맞게 스케쥴링 되지 않음. 해당 주기에 맞는 로직 추가 필요.  
            
  2. CMMS   
     (0\_CMMS\_ADMIN.py, 1\_CMMS\_USER.py, 2\_cmms\_list\_DB\_config.py, 3\_DB\_daily\_update.py)
     1. 개발 목적
        + - 기존 웹 형태의 CMMS의 속도 결함, 추가 유지보수가 어려움에 따라 로컬 DB 형식으로 새로 개발.
          - [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\03\_CMMS\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\03_CMMS\)
     2. 사용법(Admin 버전 기준)
        + - 
     3. Admin 버전 추가 기능
        1. Data\_check, service\_check 칼럼 추가 및 편집 가능.(체크 박스 칸만 가능)
     4. 개발 환경 및 접속 정보
        1. Postgres DataBase 정보
           + host = "192.168.11.61"
           + port = "5432"
           + dbname = "postgres"
           + user = "postgres"
           + pw = "rjator"
           + Schema = “cmms”
           + Dbname = “cmms\_list”
        2. Oracle 원본 Database정보
           + host = " 192.168.10.30 "
           + port = "1521"
           + dbname = "imbd"
           + user = "contentsteam"
           + pw = " zjsxpscm"
           + Schema = “imduser”
           + Dbname = “tbw\_dt\_infocol”
     5. 3\_CMMS\_list\_DB\_config.py
        + - 최초 Database Table구성 관련 code 정보
     6. 2\_DB\_daily\_update.py
        1. Oracle DB와 Postgresql DB(CMMS 서버) 간의 DB 데일리 업데이트를 위해 개발
        2. 구축1팀 서버에서 상시 작동중 ( C:\CMMS\_LIST )
        3. 시작프로그램에 등록하여 재부팅 되더라도 자동으로 실행.
        4. 오류 발생 시 실행폴더 내에 log.txt 파일 생성되어 확인가능.
     7. 기타 정보
        + - Version = Python 3.7
          - Main Library = Pyqt5
          - 배포 = exe

1. 수집 관련 Python CODE 수정 방법 및 예시
   * + - * Python Code로 작성된 수집 툴이 작동하지 않고 에러가 날 때 수정 방법에 대한 설명을 목적으로 함.
         * 아래 설명된 내용 이외에도 오류가 발생하는 원인은 무수히 많으며 모두 열거 및 설명 할 수 없음
   1. 일반적인 수집 Code 구조
      * + - 현재 수집 코드는 수집 라이브러리에 따라 차이는 있지만 크게 지식공방 수집 형태와 통합관리버전 형태 두가지로 나뉜다.
          - 통합관리버전 형태는 통합수집관리 툴 에서 실행하기 위한 대상들로 배포 또한 exe로 이루어지며 기존 코드에서 일부 코드만 추가된 형태이며 기본적인 골자는 지식공방 수집 코드 형태와 유사하다.
      1. 지식 공방 수집 형태 예시
         1. 가장 상위 에 전역 변수, Parameter 값 선언
         2. Main 함수 : 수집 함수 반복 실행 및 텍스트 저장
         3. 페이지의 상세정보 수집을 위한 인덱스 수집( 필요 없는 경우 패스) 함수
         4. 페이지의 상세정보 수집을 위한 함수
         5. [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\01\_지식공방크롤링\_2.x\금융은행\_완료!\보험\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\01_지식공방크롤링_2.x\금융은행_완료!\보험\)
         6. E. 경로의 ABL생명.py 참조
      2. 통합관리버전 형태
         1. 가장 상위 에 전역 변수, Parameter 값 선언
         2. Today 와 아웃풋 경로 및 파일 이름 선언
         3. Main 함수 : 수집 함수 실행 ,실패시 에러메시지 텍스트로 출력
         4. 수집 코드의 반복 실행 및 텍스트 저장 함수
         5. 페이지의 상세정보 수집을 위한 인덱스 수집( 필요 없는 경우 패스) 함수
         6. 페이지의 상세정보 수집을 위한 함수
         7. [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\05\_기획팀\_자체수집\자체수집\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\05_기획팀_자체수집\자체수집\)
         8. E. 경로의 01\_1\_오피넷.py 참조
   2. 대표적인 오류 발생 케이스와 해결 방법
      * + - 오류 발생 원인은 매우 다양한데 그 근본 원인은 웹페이지 내부적 변동으로 발생한다.  
            대표적인 케이스는 아래와 같다.
        1. 수집을 위해 설정된 태그나 속성 값이 변경된 경우  
           - 6.3 항목에서 자세히 설명
        2. 호출 API parameter 값이나 API URL 자체가 변경된 경우  
           -개발자 도구를 열어 변경된 Payload 페이지를 기존과 변경된 parameter 속성이 있는지 확인 한 후 해당 값을 수정한다.
           + <그림 3.>
           + 
        3. API KEY 권한이 만료된 경우  
           - 해당 API 키를 새로 발급 받아 수집 코드에서 Key값을 수정한다.
           + <그림 4.>
           + 텍스트이(가) 표시된 사진

             자동 생성된 설명
        4. 수집을 위해 설정된 input파일 등에 문제가 있는 경우( 인코딩, 수집 조건 설정 값등 )  
           - 수집 실행을 위한 조건 중 인풋파일이 있는 경우가 있는데 이 파일의 인코딩이나 구분자 등을 확인한다.
   3. 태그나 속성 값이 변한 경우 수정 방법.
      * + - [\\ddss\_nas05\MAPPLAN\MAPPLAN\_1\01.DATA\98.기획팀수집툴배포\96\_SourceCode\00\_수집툴\05\_기획팀\_자체수집\자체수집\](file:///\\ddss_nas05\MAPPLAN\MAPPLAN_1\01.DATA\98.기획팀수집툴배포\96_SourceCode\00_수집툴\05_기획팀_자체수집\자체수집\)
          - 해당 페이지의 02\_2\_테슬라전기차.py 파일을 예시로 사용하였으며 실제 기존 코드에 에러가 발생 했을 때 수정한 과정을 아래에 기록하였음.
        1. 기존 코드를 실행 하여 에러 메시지, 현상에 대한 확인.  
           - 에러 메시지는 발생하지 않으며 수집 결과 빈 텍스트가 반환  
           - 상세 정보 수집 함수에서 아래 그림과 같이 try ~ except : pass 로 설정되어 아무 값 없이 리턴  
           <그림 4.>  
           텍스트, 모니터, 스크린샷, 화면이(가) 표시된 사진

           자동 생성된 설명
        2. Try ~ except 문 삭제 후 실행 : panel-panel 클래스를 찾을 수 없다는 에러 메시지 출력
        3. 수집 사이트에 접속하여 수집 하고자 했던 위치의 태그, 속성들을 확인  
           <그림 5.>  
           텍스트이(가) 표시된 사진

           자동 생성된 설명
        4. div = bsObj.find('div',{"class":"panel-panel"}) 코드에 해당하는 클래스 값을 찾을 수가 없음
        5. <그림 4.>에서 기존 수집 항목을 보면 명칭, 주소, 지점, 충전기정보 로 파악되기에 해당 정보들을 얻기 위한 태그들에 대한 정보를 <그림 5.> 와 같이 개발자 도구에서 찾아 수정해 준다.  
           <그림 6.>  
           텍스트이(가) 표시된 사진

           자동 생성된 설명
        6. <그림 6.> 과 같이 수정이 이루어 졌으며 주요 수정 사항 으로는
           + 기본 정보가 들어있던 클래스가 div = bsObj.find('div',{"class":"panel-panel"}) 에서 div = bsObj.find('address', {"class": "vcard"}) 로 변경.
           + Tell 과 차저 정보는 기존과 같은 태그이기에 수정이 없음
           + div태그의 Lacation-map ID에서 좌표 정보를 얻을수 있어 이를 가져와 정제 하여 x,y 좌표를 분리 하여 저장.

-끝-