

UWordMap 데이터베이스 구조 설명

데이터베이스 파일 형식

데이터베이스 파일은 CSV 파일 4개와 윈도우용 조회 프로그램에 내장된 SQLite 데이터베이스 파일(`uwm.db`) 이 있다.

CSV 파일

- 각 테이블은 탭으로 필드가 구분된 CSV 파일로 되어 있다.
- 파일의 첫 번째 줄은 필드 이름이다.
- 필드의 내용에 필드 구분자로 쓰는 **탭 문자** 또는 **큰 따옴표(")**가 들어가 있는 경우, 필드 내용의 앞과 뒤를 **큰 따옴표(")** 한 개로 감싸고, 필드 내용에 있는 큰 따옴표는 두 개를 사용한다.
 - 예문 원문: `서림이가 개연한 어조로 말하자 박유복이가 걱정에게 "형님, 서 장사의 말에도 일리가 있소."`라고 말하였다.
 - 예문 수정 후: `"서림이가 개연한 어조로 말하자 박유복이가 걱정에게 ""형님, 서 장사의 말에도 일리가 있소.""`라고 말하였다."
 - 일반적인 CSV 처리 프로그램은 이러한 처리를 자동으로 해주므로 파일을 직접 다룰 때의 주의 사항이다.

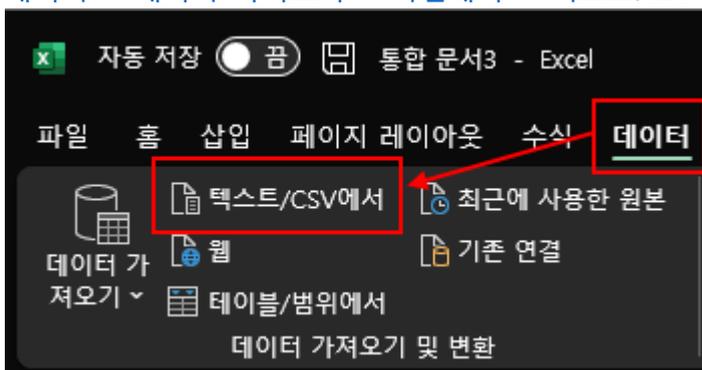
파이썬에서 CSV 파일을 여는 예제 프로그램

아래 파이썬 프로그램은 `uwin` 테이블의 CSV 파일을 여는 방법을 나타낸다.

```
import csv
with open('uwin.csv', 'r', encoding='utf-8') as f:
    reader = csv.reader(f, delimiter='\t')
    for i, row in enumerate(reader):
        if i == 0:
            continue # 첫 줄에 있는 필드 목록은 읽지 않는다.
        keyid, name, sensetag, pos, origin, explain1, bigo, spattern = row
        # 아래에 필요한 내용을 추가한다.
```

엑셀에서 CSV 파일을 여는 방법

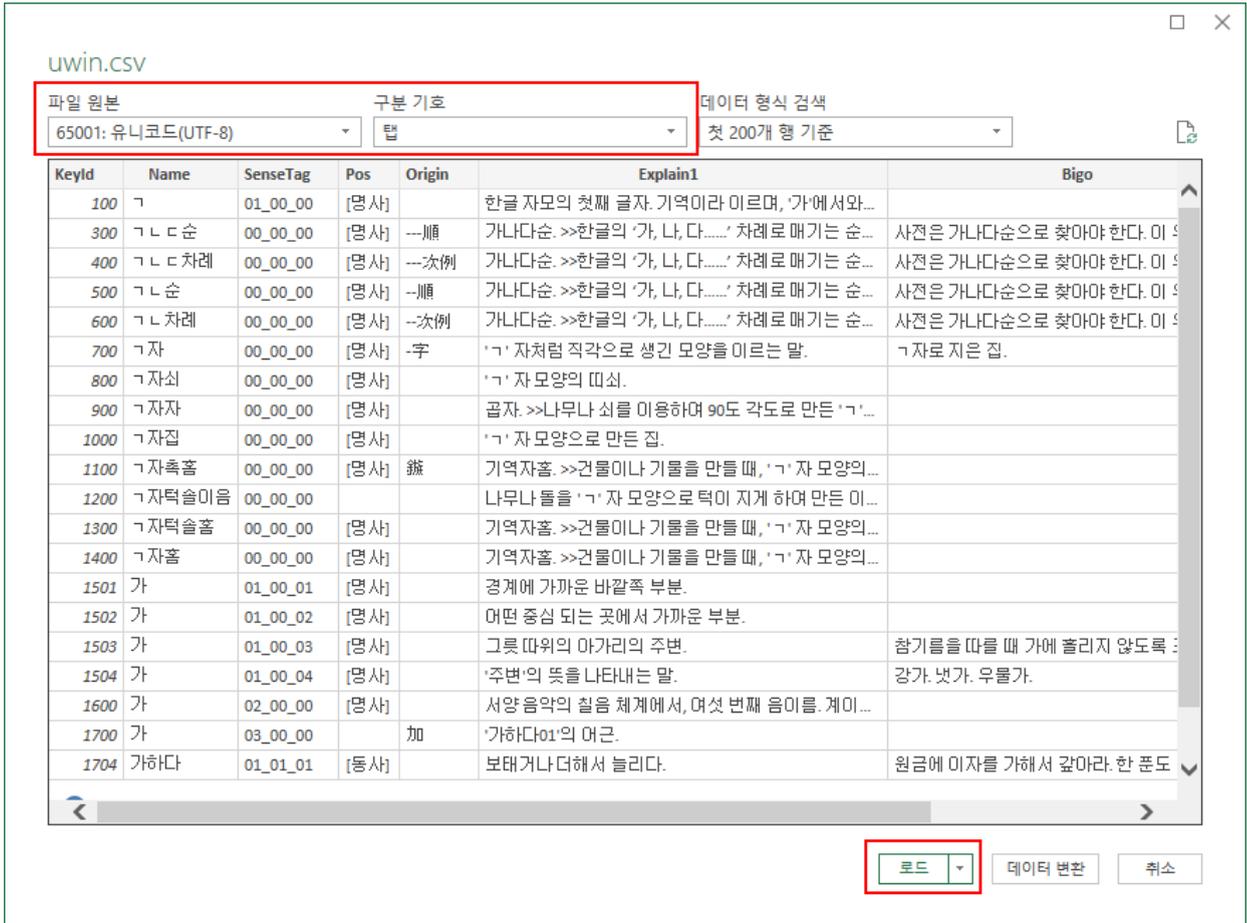
1. 데이터 -> 데이터 가져오기 -> 파일에서 -> 텍스트/CSV에서 선택



2. csv 파일 선택

3. 파일 원본 65001: 유니코드(UTF-8), 구분 기호 탭 확인

4. 로드 버튼을 누른다.



- 주의: uwin 테이블 등은 분량이 많아서 엑셀에서 가져올 때 오류가 발생하거나 일부만 가져올 수 있다. 가급적 대용량 파일을 처리할 수 있는 다른 프로그램을 사용하는 것이 좋다.
- Notepad++, Emeditor 등과 같은 텍스트 편집기를 사용하는 것을 추천한다.

SQLite DB 파일 사용법(전문가용)

윈도우용 조회 프로그램에 내장된 uwm.db 파일은 SQLite 프로그램으로 내용을 확인할 수 있다.(CSV 파일에 있는 것과 동일한 내용)

또한 SQL 문으로 원하는 조건을 직접 입력하여 조회하거나 내용을 추가, 수정할 수 있다.

- SQLite 프로그램 다운로드(윈도우, 리눅스, 맥 등에서 사용 가능)
 - <https://www.sqlite.org/download.html> : (영어)SQLite 공식 프로그램(SQL 명령어 직접 입력)
 - <https://sqlitebrowser.org/> : (영어)SQLite 브라우저 프로그램(일부 기능 버튼으로 조작 가능)

테이블 목록

- uwin : 낱말의 표제어와 뜻풀이, 용례
- relation : 낱말의 상하위어 관계
- rel_etc : 두 낱말 간의 동의어, 반댓말, 높임말 등의 관계
- rel_vn : 용언과 낱말의 의미제약 관계

필드 목록 설명에서 따로 괄호로 데이터 종류를 표시를 하지 않은 필드는 모두 TEXT이다.

- **TEXT**: 문자열
- **INTEGER**: 정수(64비트. C#/자바에서 `long`, C/C++에서 `long long`)

uwin 테이블

파일: `uwin.csv`

필드 목록

- **KeyId(INTEGER)**: 낱말 고유 번호.
 - `relation`, `rel_etc`, `rel_vn` 테이블에서 사용한다.
- **Name**: 표제어
- **SenseTag**: 어깨번호
- **Pos**: 품사
- **Origin**: 원어(외래어, 한자). -로 표시된 것은 순우리말 부분이다.
 - 예: `가나다순` -> `---順`
- **Explain1**: 뜻풀이
- **Bigo**: 용례

쿼리 예제

'의사'로 끝나는 표제어를 가진 단어를 검색하는 예제

```
SELECT keyid, name, sensetag, pos, explain1
FROM uwin
WHERE name LIKE '%의사';
```

relation 테이블

파일: `relation.csv`

상하위어 관계를 기록하는 테이블이다. 상하위어 관계에 적용되는 규칙은 아래와 같다.

- 모든 낱말의 최상위어는 `UWIN(100000000)`이다.
- 하위어는 단 하나의 상위어를 가진다.
- 상위어는 여러 하위어를 가질 수 있다.

필드 목록

- **Child(INTEGER)**: 하위어 고유 번호
- **Parent(INTEGER)**: 상위어 고유 번호
- **IsTerminal(INTEGER)**: 하위어 존재 여부(1이면 하위어의 하위어가 없음)

쿼리 예제

'사과_05_00_00'의 상위어를 찾는 예제

```

SELECT * FROM uwin WHERE keyid IN (
  SELECT parent FROM relation WHERE child IN (
    SELECT b.keyid FROM uwin b WHERE name='사과' AND sensetag='05_00_00'
  )
);

```

rel_etc 테이블

파일: [rel_etc.csv](#)

두 낱말 사이의 동의어, 반의어 등의 관계를 기록하는 테이블이다. 따라서 한 낱말의 대한 동의어를 찾으려면 **RelType**이 **dongui**이고, **ConId1**, **ConId2** 양 쪽 필드에서 찾고자 하는 낱말의 **KeyId**가 일치하는 레코드를 찾아야 한다.

필드 목록

- **RelType**: 관계 종류
- **ConId1**(INTEGER): 낱말1 KeyID
- **ConId2**(INTEGER): 낱말2 KeyID

쿼리 예제

'가속_01_00_00'의 반댓말을 찾는 예제

```

SELECT * FROM uwin WHERE keyid IN (
  SELECT conid2 FROM rel_etc WHERE reltype='banui' AND conid1 IN (
    SELECT keyid FROM uwin WHERE name='가속' AND sensetag='01_00_00'
  )
);
# 만약 나오지 않으면 쿼리 두번째 줄에 있는 `conid1`과 `conid2`를 바꾸어서 입력해본다.

```

rel_vn 테이블

파일: [rel_vn.csv](#)

용언의 의미제약 정보가있는 테이블이다.

필드 목록

- **RelType**: 관계 종류(격조사)
- **Verb**(INTEGER): 용언
- **Noun**(INTEGER): 명사

쿼리 예제

'마시다'와 함께 사용할 수 있는 낱말 찾기

```
SELECT * FROM uwin WHERE keyid IN (  
  SELECT noun FROM rel_vn WHERE verb IN (  
    SELECT keyid FROM uwin WHERE name='마시다'  
  )  
);
```