한국음성학회 실험음성학연구회 하계 워크숍
성신여자대학교 수정관 413호

성신여자대학교
SUNGSHIN UNIVERSITY

Python을 활용한 대용량 데이터 전처리

Tae-Jin Yoon

Sungshin Women's University

Caveat

- This is not an introduction to Python workshop.
- I will show you how to handle a JSON file using the basics of the Python Programming Language.
- And you will learn what can be achieved by watching how python is applied to the JSON file.

Read map

01 JSON

02 Pandas

03 Parselmouth

Part 1.
Why do we need to know how to deal with json?

공개 음성 데이터 포털

AI Hub

https://www.aihub.or.kr

https://corpus.korean.go.kr/request/reausetMain.do

Python Overview

문화체육관광부 국립국어원 **모두의 말뭉치**

윤태진 님 | 내 정보 관리 | 나가기

말뭉치 신청 | 사용자 참여 | 말뭉치 활용 | 알립니다

모두의 말뭉치

미래를 준비하는 소중한 우리말 자원

신규

분석

맞춤법 교정 말뭉치 2…

(버전 1.0) 온라인 대화 자료를 대상으로 한국어 처리 도구가 분석할 수 있는 수준으로 오탈자 등을 교정한…

신청하기

신문 말뭉치 2022

(버전 1.0) 2021년 생산된 신문 기사 중 매체로부터 저작권 이용을 허락받은 기사를 기계 분석 가능한 형식으로…

신청하기

일상 대화 음성 말뭉치…

(버전 1.0) 일상 대화의 음성(PCM 파일)과 전사 자료로 구성된 말뭉치입니다.

신청하기

일상 대화 말뭉치 2021

(버전 1.0) 특정 주제 또는 제시 자료로 자유롭게 대화를 나눈 일상 대화 말뭉치입니다.

신청하기

JSON 화일이 불편합니다. 답변

4

ari0***

Json 파일 말고 한글화일로 제공 해 줄수 있나요 ?
컴퓨터 전문가들이 해야 하는 분야로 보입니다.
아래한글이나 워드문서로 제공을 하면 일반적으로 편하게 사용 할 수있을 듯 한데,
생소한 JSON 화일로 제공이 되고 있어서
다운받은 자료가 그림의 떡 입니다.

관리자

안녕하세요. 모두의 말뭉치 운영자입니다.
국립국어원 말뭉치에 관심을 가져 주셔서 감사합니다.

JSON 형식의 자료를 기타 형식의 자료로 변환하여 배포할 계획은 없습니다.
다만, 2022년도에 '모두의 말뭉치 활용 지원 자료 구축' 사업이 진행되고 있습니다.
해당 사업을 통해 모두의 말뭉치를 활용하는 방법(동영상 강의 등)이 공개될 예정이므로 이를 적극 활용해 주시기 바랍니다.

고맙습니다.

모두의 말뭉치

@user-im2uw8fy8z 구독자 305명 동영상 9개

2022년 국립국어원 '모두의 말뭉치' 행사 운영을 위한 유튜브 >

구독

홈 | 동영상 | 실시간 | 재생목록 | 커뮤니티 | 채널 | 정보

[모두의 말뭉치 활용 방법] 어휘 의미 분석 말뭉치를 사용한 '먹다'의 출현 환경 탐색 (6회차)

조회수 834회 • 6개월 전

[모두의 말뭉치 활용 방법] 일상 대화 말뭉치를 사용한 '완전'의 부사적 용법 탐색 (5회차)

조회수 719회 • 6개월 전

[모두의 말뭉치 활용 방법] 분석 말뭉치 활용하기 (4회차)

조회수 910회 • 6개월 전

[모두의말뭉치 활용 방법] 원시 말뭉치 활용하기 (3회차)

조회수 1천회 • 6개월 전

[모두의 말뭉치 활용 방법] 말뭉치 파일 탐색하기 (2회차)

조회수 1.3천회 • 6개월 전

[모두의 말뭉치 활용 방법] 모두의 말뭉치 소개 및 파일 신청하기 (1회차)

조회수 1.3천회 • 7개월 전

모두의 말뭉치 홍보 영상 (국문)

조회수 478회 • 7개월 전

https://www.youtube.com/@user-im2uw8fy8z/videos

의견 제시

모두의 말뭉치에 전하실 개선 의견을 제시해 주세요.
제시해 주신 소중한 의견은 모두의 말뭉치 개선을 위해 활용하겠습니다.

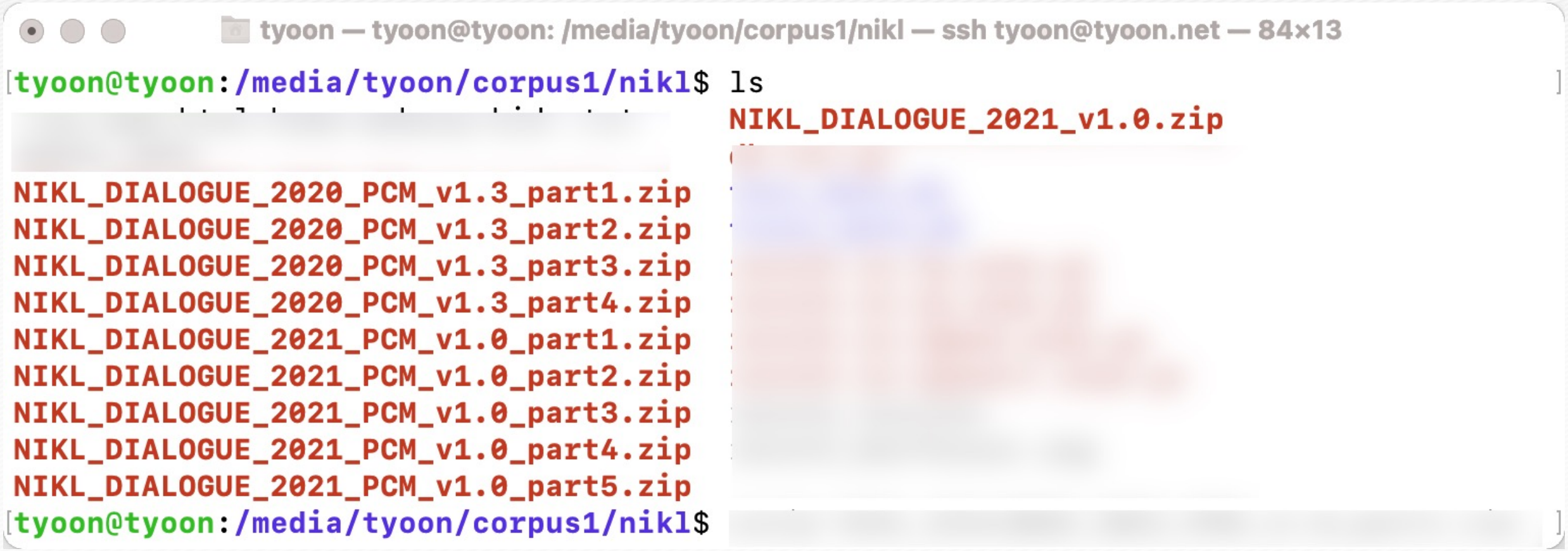
의견 제시 | 말뭉치 제안

제목		작성자	작성 날짜
이용 불편	1	xenovi***	2023.04.28.

진입 장벽이 너무 높아서 이용하기 불편합니다.
압축 풀기부터 안 되는 선생님들도 계신 것 같은데 파이썬을 활용해서 분석해야 하니 많은 연구자들이 일반적으로 접근하기 어렵습니다.
(맥과 리눅스라고 하셨는데 윈도우에서도 안 열리시는 분들이 계십니다.)
많은 비용과 노고를 들여서 만든 자료인데 결국 테크를 활용할 수 있는 몇몇 연구자들만 볼 수 있도록 되어 있습니다.
또한 동영상 학습 자료를 제공한다고 하셨으나 중요한 내용은 특히나 설명이 너무 압축적인 데다가 빨리 지나갑니다.
그리고 코드를 작성하셨는데 코드만 따로 오픈되어 있지 않아서 홈페이지에 올려 놓으신 스크립트에서 코드만 따로 다시 찾아야 합니다.
코드 오픈을 해 주시면 좋겠습니다.
용량이 크고 서버를 운영하는 데에 어려움이 있겠으나 접근 방법이나 사용법을 좀 더 합리적이며 용이하게 해주셨으면 합니다.
지금은 사용하는 사람들만 사용하게 되어 있는 것 같습니다.

Part 2.
Data at a glance

- mkdir NIKL_DIALOGUE_2021
- unzip …



tyoon — tyoon@tyoon: /media/tyoon/corpus1/nikl — ssh tyoon@tyoon.net — 91×19

```
tyoon@tyoon:/media/tyoon/corpus1/nikl$ unzip NIKL_DIALOGUE_2021_v1.0.zip
Archive:  NIKL_DIALOGUE_2021_v1.0.zip
   creating: NIKL_DIALOGUE_2021_v1.0/
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000001.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000002.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000003.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000004.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000005.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000006.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000007.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000008.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000009.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000010.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000011.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000012.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000013.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000014.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000015.json
  inflating: NIKL_DIALOGUE_2021_v1.0/SDRW2100000016.json
```

Directory (or folder) 구조

```
(base) tyoon@TAEs-MacBook-Pro 실험음성학연구회2023 % tree data
data
└── SDRW2100000001
    ├── SDRW2100000001.1.1.1.pcm
    ├── SDRW2100000001.1.1.10.pcm
    ├── SDRW2100000001.1.1.100.pcm
    ├── SDRW2100000001.1.1.101.pcm
    ├── SDRW2100000001.1.1.102.pcm
    ├── SDRW2100000001.1.1.103.pcm
    ├── SDRW2100000001.1.1.122.pcm
    ├── SDRW2100000001.1.1.123.pcm
    ├── SDRW2100000001.1.1.124.pcm
    ├── SDRW2100000001.1.1.125.pcm
    ├── SDRW2100000001.1.1.126.pcm
    ├── SDRW2100000001.1.1.127.pcm
    ├── SDRW2100000001.1.1.128.pcm
    ├── SDRW2100000001.1.1.129.pcm
    ├── SDRW2100000001.1.1.13.pcm
    ├── SDRW2100000001.1.1.74.pcm
    ├── SDRW2100000001.1.1.75.pcm
    ├── SDRW2100000001.1.1.76.pcm
    ├── SDRW2100000001.1.1.77.pcm
    ├── SDRW2100000001.1.1.96.pcm
    ├── SDRW2100000001.1.1.97.pcm
    ├── SDRW2100000001.1.1.98.pcm
    └── SDRW2100000001.1.1.99.pcm
├── SDRW2100000001.json
├── SDRW2100000001_test copy.json
├── SDRW2100000001_test.json
├── SDRW2100000001_test.txt
└── SDRW2100000002
    ├── SDRW2100000002.1.1.1.pcm
    ├── SDRW2100000002.1.1.10.pcm
    ├── SDRW2100000002.1.1.100.pcm
    ├── SDRW2100000002.1.1.101.pcm
    ├── SDRW2100000002.1.1.102.pcm
    ├── SDRW2100000002.1.1.103.pcm
```

What we will achieve today!

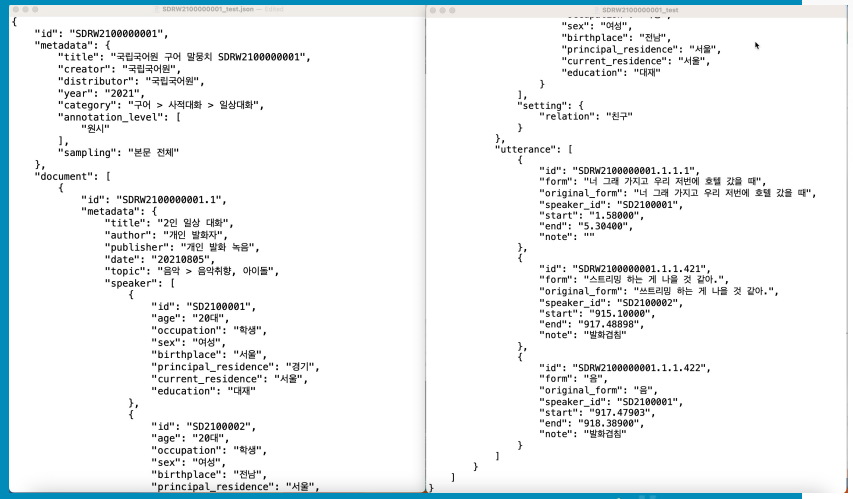
SDRW2100000001_test.json — Edited


```
{
    "id": "SDRW2100000001",
    "metadata": {
        "title": "국립국어원 구어 말뭉치 SDRW2100000001",
        "creator": "국립국어원",
        "distributor": "국립국어원",
        "year": "2021",
        "category": "구어 > 사적대화 > 일상대화",
        "annotation_level": [
            "원시"
        ],
        "sampling": "본문 전체"
    },
    "document": [
        {
            "id": "SDRW2100000001.1",
            "metadata": {
                "title": "2인 일상 대화",
                "author": "개인 발화자",
                "publisher": "개인 발화 녹음",
                "date": "20210805",
                "topic": "음악 > 음악취향, 아이돌",
                "speaker": [
                    {
                        "id": "SD2100001",
                        "age": "20대",
                        "occupation": "학생",
                        "sex": "여성",
                        "birthplace": "서울",
                        "principal_residence": "경기",
                        "current_residence": "서울",
                        "education": "대재"
                    },
                    {
                        "id": "SD2100002",
                        "age": "20대",
                        "occupation": "학생",
                        "sex": "여성",
                        "birthplace": "전남",
```


SDRW2100000001_test


```
                        "sex": "여성",
                        "birthplace": "전남",
                        "principal_residence": "서울",
                        "current_residence": "서울",
                        "education": "대재"
                    }
                ],
                "setting": {
                    "relation": "친구"
                }
            },
            "utterance": [
                {
                    "id": "SDRW2100000001.1.1.1",
                    "form": "너 그래 가지고 우리 저번에 호텔 갔을 때",
                    "original_form": "너 그래 가지고 우리 저번에 호텔 갔을 때",
                    "speaker_id": "SD2100001",
                    "start": "1.58000",
                    "end": "5.30400",
                    "note": ""
                },
                {
                    "id": "SDRW2100000001.1.1.421",
                    "form": "스트리밍 하는 게 나을 것 같아.",
                    "original_form": "쓰트리밍 하는 게 나을 것 같아.",
                    "speaker_id": "SD2100002",
                    "start": "915.10000",
                    "end": "917.48898",
                    "note": "발화겹침"
                },
                {
                    "id": "SDRW2100000001.1.1.422",
                    "form": "음",
                    "original_form": "음",
                    "speaker_id": "SD2100001",
                    "start": "917.47903",
                    "end": "918.38900",
                    "note": "발화겹침"
                }
            ]
        }
    ]
```

Our Goal




```
{
    "id": "SDRW2100000001",
    "metadata": {
        "title": "국립국어원 구어 말뭉치 SDRW2100000001",
        "creator": "국립국어원",
        "distributor": "국립국어원",
        "year": "2021",
        "category": "구어 > 사적대화 > 일상대화",
        "annotation_level": [
            "원시"
        ],
        "sampling": "본문 전체"
    },
    "document": [
        {
            "id": "SDRW2100000001.1",
            "metadata": {
                "title": "2인 일상 대화",
                "author": "개인 발화자",
                "publisher": "개인 발화 녹음",
                "date": "20210805",
                "topic": "음악 > 음악취향, 아이돌",
                "speaker": [
                    {
                        "id": "SD2100001",
                        "age": "20대",
                        "occupation": "학생",
                        "sex": "여성",
                        "birthplace": "서울",
                        "principal_residence": "경기",
                        "current_residence": "서울",
                        "education": "대재"
                    },
                    {
                        "id": "SD2100002",
                        "age": "20대",
                        "occupation": "학생",
                        "sex": "여성",
                        "birthplace": "전남",
                        "principal_residence": "서울",
                        "current_residence": "서울",
                        "education": "대재"
                    }
                ],
                "setting": {
                    "relation": "친구"
                }
            },
            "utterance": [
                {
                    "id": "SDRW2100000001.1.1.1",
                    "form": "너 그래 가지고 우리 저번에 호텔 갔을 때",
                    "original_form": "너 그래 가지고 우리 저번에 호텔 갔을 때",
                    "speaker_id": "SD2100001",
                    "start": "1.58000",
                    "end": "5.30400",
                    "note": ""
                },
                {
                    "id": "SDRW2100000001.1.1.421",
                    "form": "스트리밍 하는 게 나을 것 같아.",
                    "original_form": "쓰트리밍 하는 게 나을 것 같아.",
                    "speaker_id": "SD2100002",
                    "start": "915.10000",
                    "end": "917.48898",
                    "note": "발화겹침"
                },
                {
                    "id": "SDRW2100000001.1.1.422",
                    "form": "음",
                    "original_form": "음",
                    "speaker_id": "SD2100001",
                    "start": "917.47903",
                    "end": "918.38900",
                    "note": "발화겹침"
                }
            ]
        }
    ]
}
```


	doc_id	docs_title	docs_creator	docs_distributor	docs_year	docs_category	docs_annotation_level
0	SDRW2100000001	국립국어원 구어 말뭉치 SDRW2100000001	국립국어원	국립국어원	2021	구어 > 사적대화 > 일상대화	원시
1	SDRW2100000001	국립국어원 구어 말뭉치 SDRW2100000001	국립국어원	국립국어원	2021	구어 > 사적대화 > 일상대화	원시
2	SDRW2100000001	국립국어원 구어 말뭉치 SDRW2100000001	국립국어원	국립국어원	2021	구어 > 사적대화 > 일상대화	원시

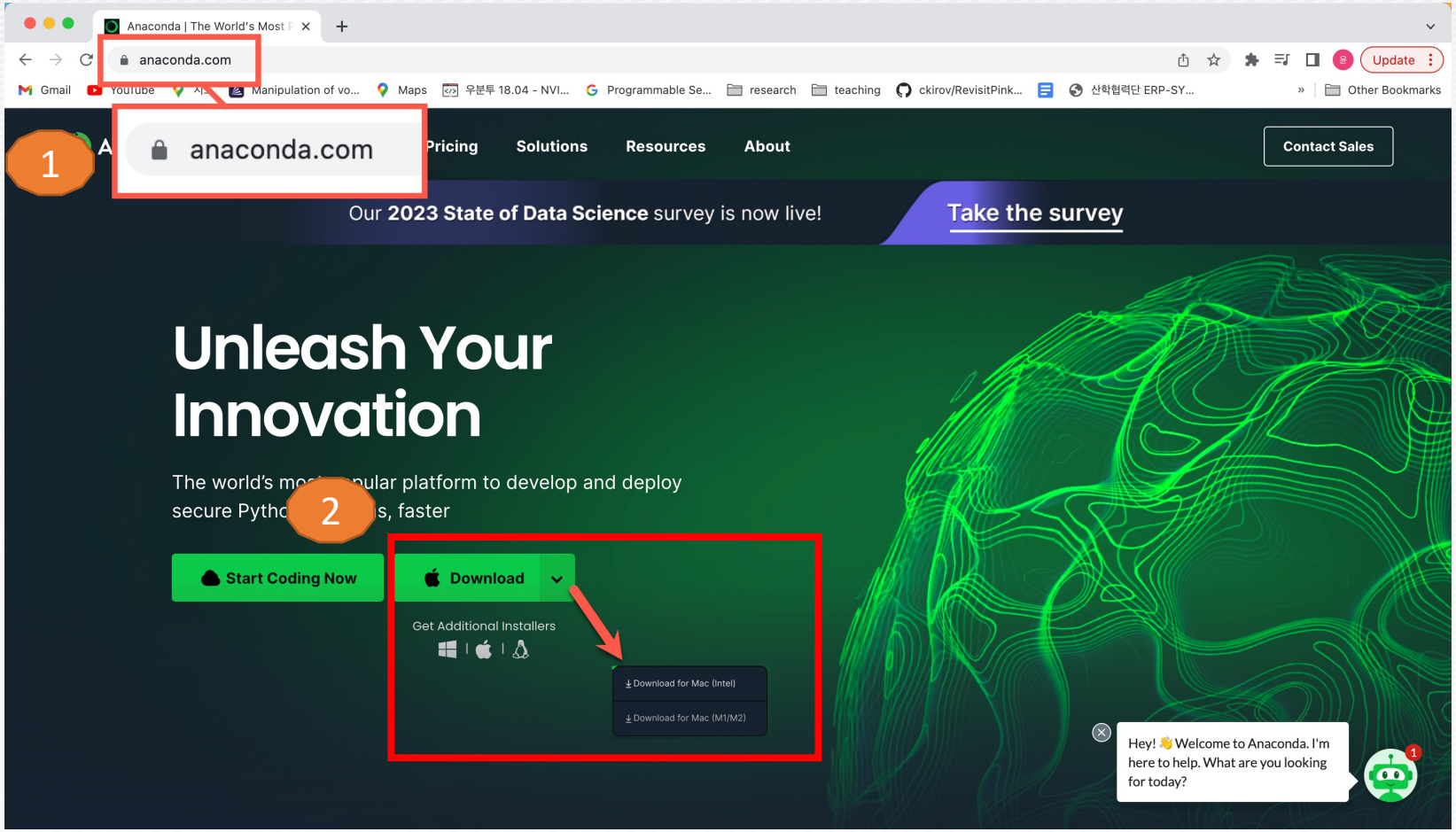
docs_sampling	utt_meta_id	utt_title	...	utt_form	utt_original_form	speaker_index	speaker_age	speaker_occupation
본문 전체	SDRW2100000001.1	2인 일상 대화	...	너 그래 가지고 우리 저번에 호텔 갔을 때	너 그래 가지고 우리 저번에 호텔 갔을 때	speaker_1	20대	학생
본문 전체	SDRW2100000001.1	2인 일상 대화	...	음	음	speaker_1	20대	학생
본문 전체	SDRW2100000001.1	2인 일상 대화	...	스트리밍 하는 게 나을 것 같아.	쓰트리밍 하는 게 나을 것 같아.	speaker_2	20대	학생

speaker_sex	speaker_birthplace	speaker_principal_residence	speaker_current_residence	speaker_education
여성	서울	경기	서울	대재
여성	서울	경기	서울	대재
여성	전남	서울	서울	대재

Part 3.
Anaconda & Jupyter Notebook

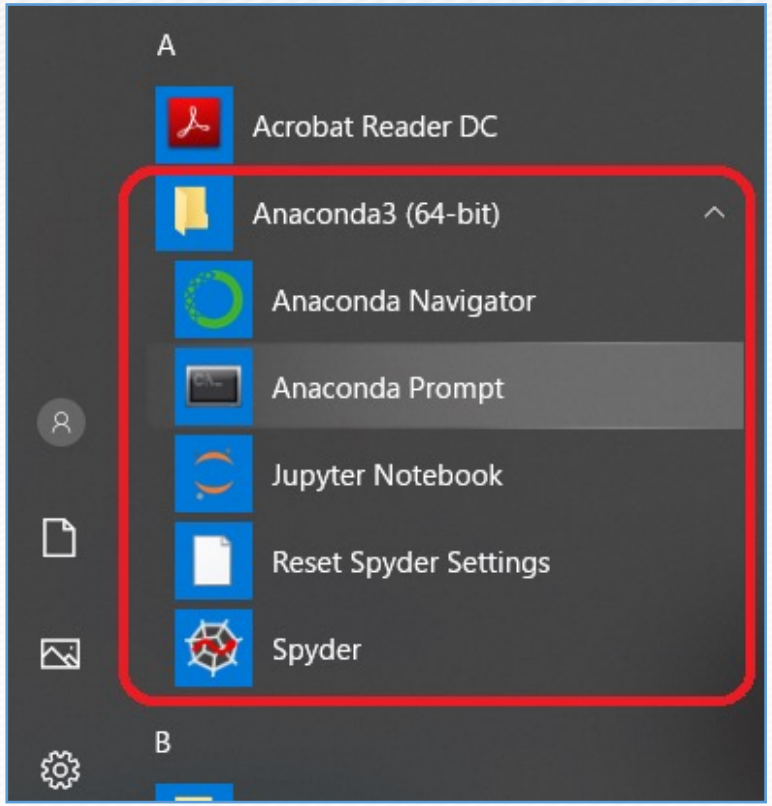
1 Anaconda

- Anaconda: Python을 용이하게 사용할 수 있도록 해주는 생태계
- https://www.anaconda.com

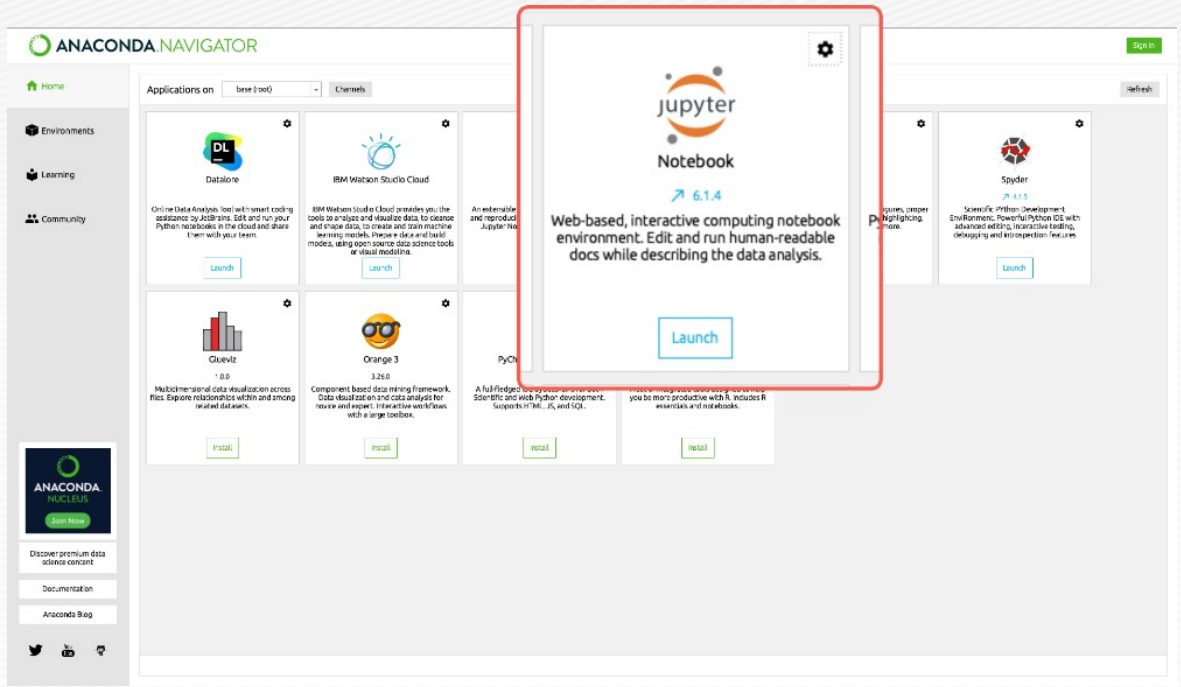


2 Jupyter notebook

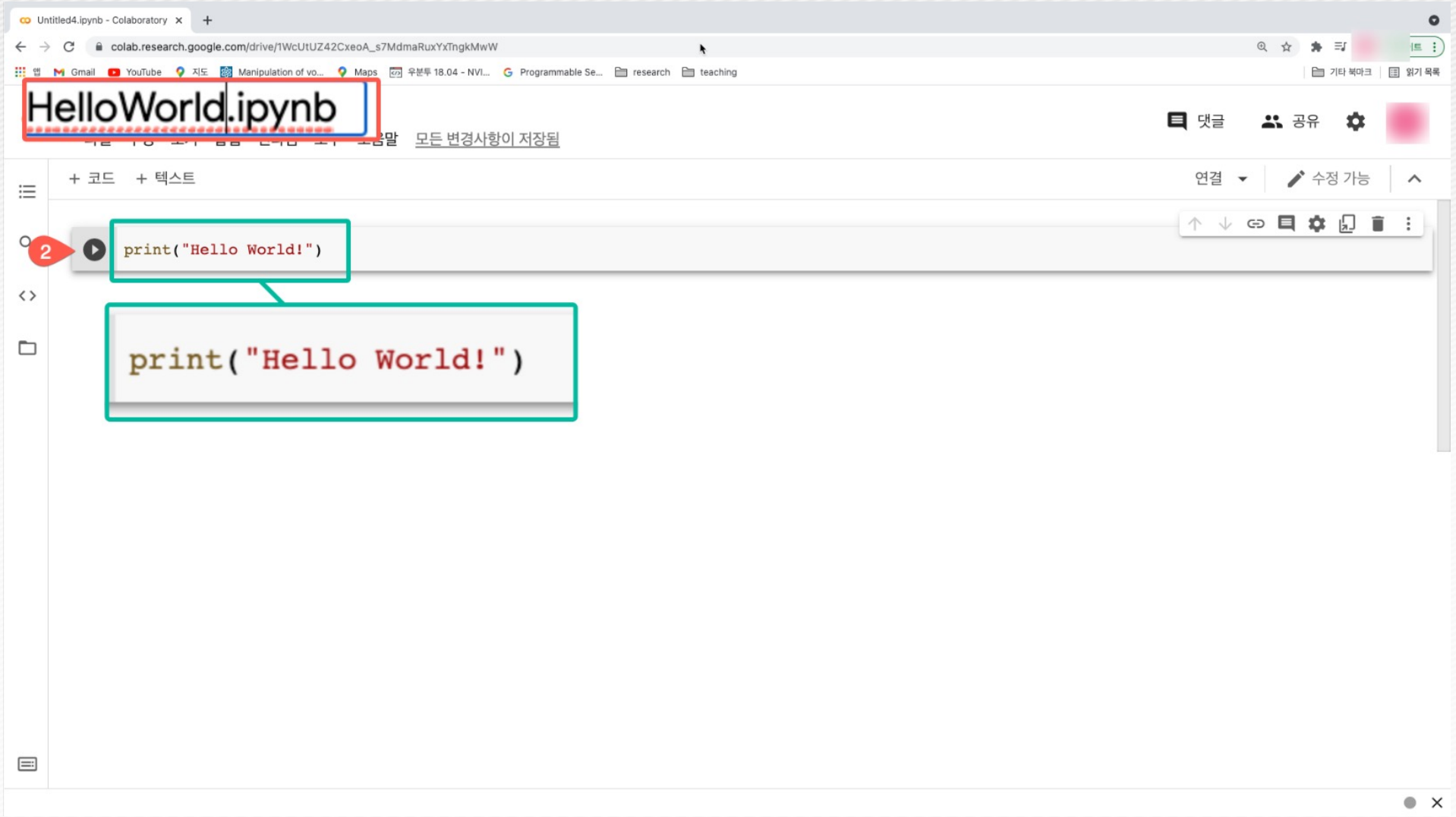
window



mac



4 실습(3)


Untitled4.ipynb - Colaboratory
colab.research.google.com/drive/1WcUtUZ42CxeoA_s7MdmaRuxYxTngkMwW
HelloWorld.ipynb
1
댓글
공유
모든 변경사항이 저장됨
+ 코드
+ 텍스트
연결
수정 가능
2
print("Hello World!")
print("Hello World!")

Part 4.
Python Overview

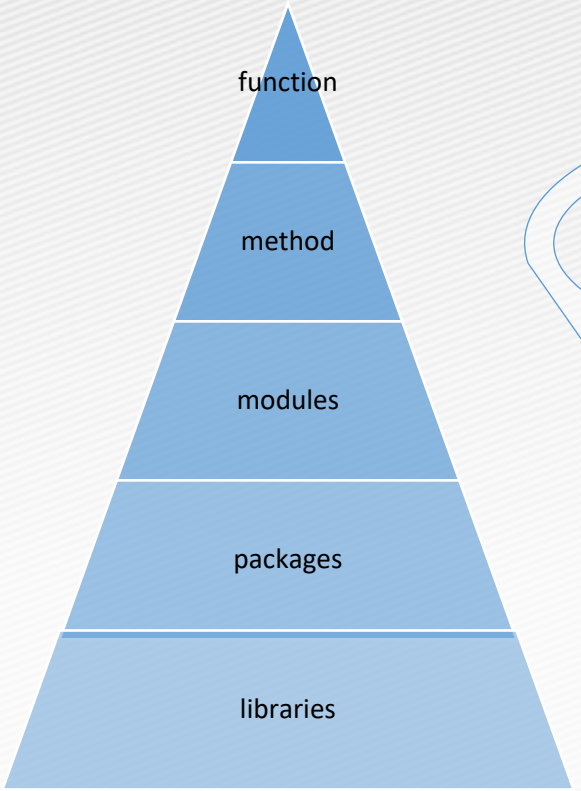
Guido van Rossum
귀도 반 로섬

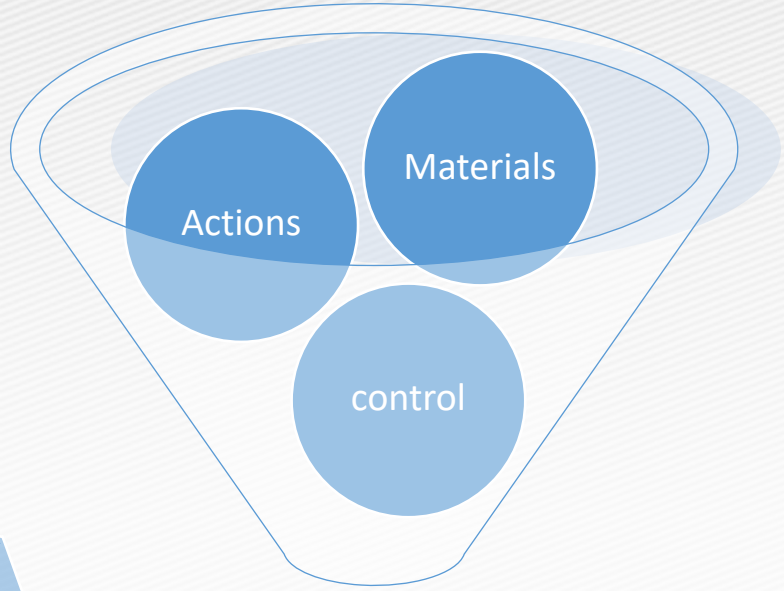
89 크리스마스 연휴

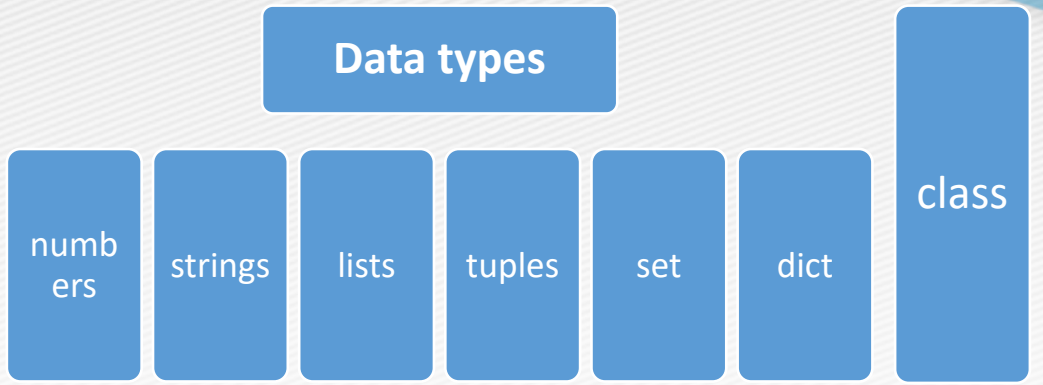
99 DARPA

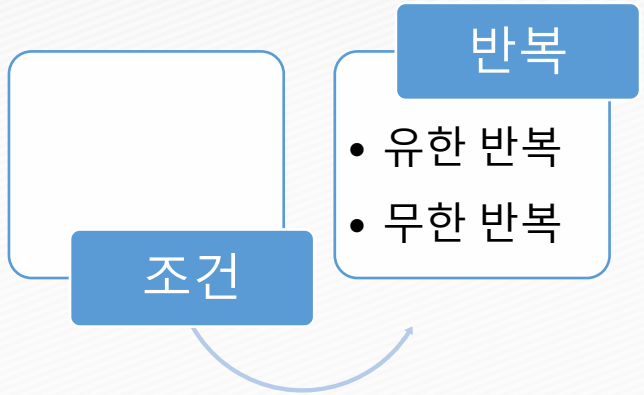
Computer Programming for Everybody

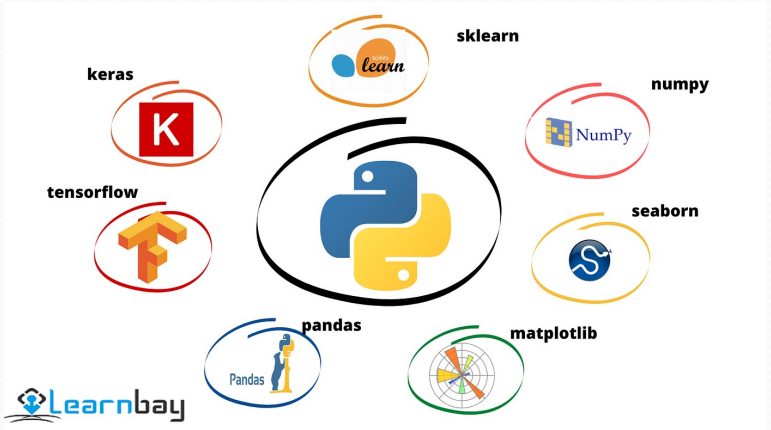
https://www.python.org/doc/essays/cp4e/











출처 https://realpython.com/python-basics/

```
import json
```

library

```
fname = "./data/SDRW2100000001_test.json”
```

variable

str(ing) data type

```
f = open(fname):
```

instance

class

```
    text = f.read()
```

method

```
f.close()
print(text)
```

function

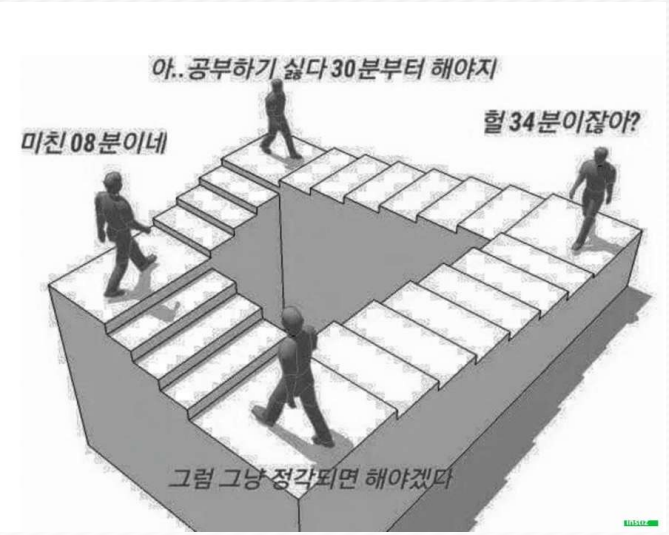
반복문

- for loop
- while loop

유한 loop

Angry Birds

무한 loop



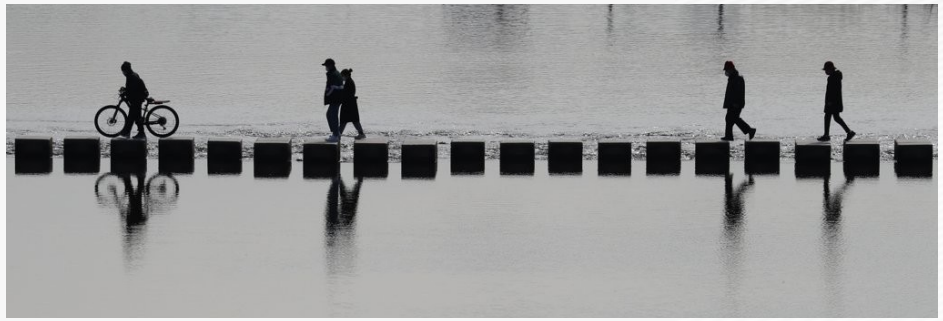
Penrose Stairs

https://www.linkedin.com/pulse/gamification-lessons-from-dr-angry-birds-gurpreet-bajaj

http://www.koreapas.com/bbs/view.php?id=gofun&page=1447&sn1=&divpage=1&sn=off&ss=on&sc=on&select_arrange=headnum&desc=asc&no=234553

list.append(data)

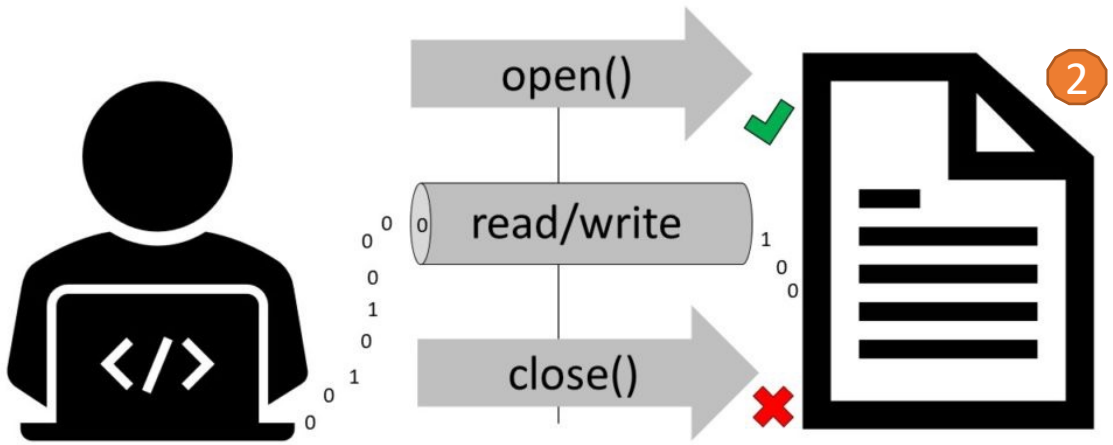
```
capital_letter_list = []
words = ['Alice', 'alice', 'ALICE', 'aLIce', 'aLICE']

for word in words:
    capital = word.upper()
    capital_letter_list.append(capital)

print(capital_letter_list)
```



```
['ALICE', 'ALICE', 'ALICE', 'ALICE', 'ALICE']
```

출처: https://www.futurelearn.com/info/courses/secondary-programming-pedagogy/0/steps/68413

파일 입출력 과정



1
```
fname = "./data/SDRW2100000001_test.json"
f = open(f)
text = f.read()
f.close()
print(text)
```

2
```
with open(fname) as f:
    text = f.read()
print(text)
```

https://blog.finxter.com/simple-guide-to-python-close/

JSON: The Fat-Free Alternative to XML.
- Douglas Crockford

JavaScript Object Notation (JSON)

- a lightweight data-interchange format based on the syntax of JavaScript objects.
- a text-based, human-readable, language-independent format for repres enting structured object data for easy transmission or saving.
- Despite the fact that it's syntax closely resembles JavaScript objects, JSO N can be used independently outside JavaScript.
- http://www.json.org/

https://medium.com/python-pandemonium/json-the-python-way-91aac95d4041

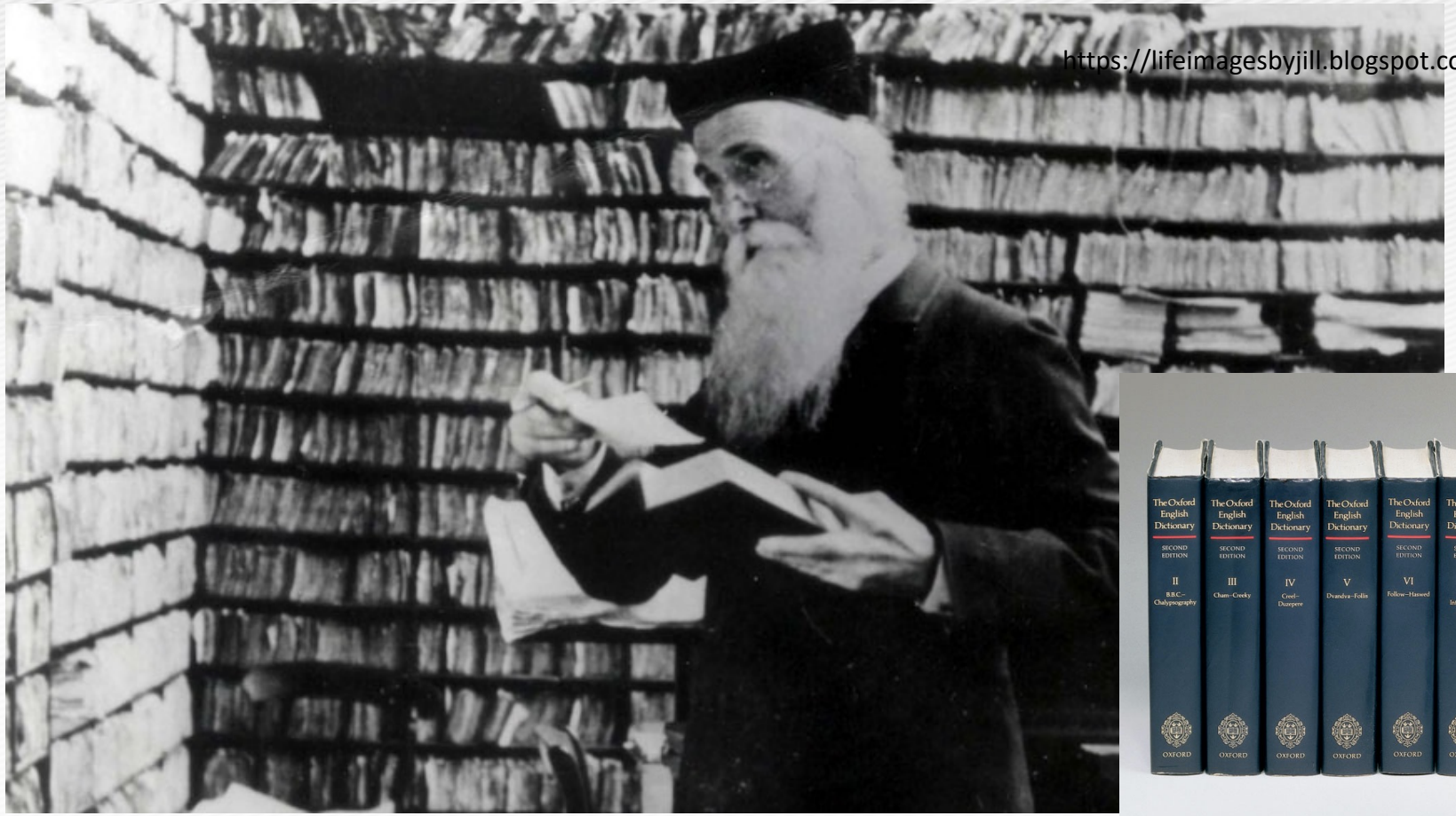
- **loads() — to deserialize a JSON document to a Python object.**
- load() — to deserialize a JSON formatted stream (which supports reading from a file) to a Python object.

```
fname = "./data/SDRW2100000001_test.json"
with open(fname) as f:
    cnt = f.read()
print(cnt)
```

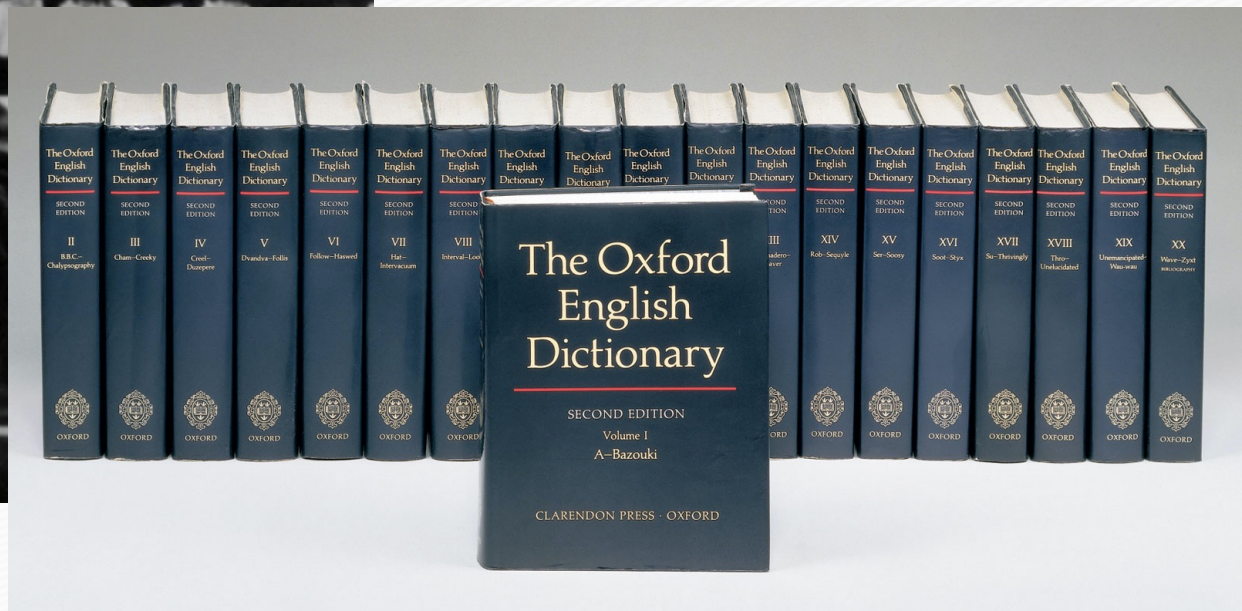

```
{
    "id": "SDRW2100000001",
    "metadata": {
        "title": "국립국어원 구어 말뭉치 SDRW2100000001",
        "creator": "국립국어원",
        "distributor": "국립국어원",
        "year": "2021",
        "category": "구어 > 사적대화 > 일상대화",
        "annotation_level": [
            "원시"
        ],
        "sampling": "본문 전체"
    },
    "document": [
        {
```

Dictionary

James Murray & The Oxford English Dictionary



https://lifeimagesbyjill.blogspot.com/2021/02/the-dictionary-of-lost-words-and-oxford.html



https://www.abc.net.au/news/2020-04-09/history-of-the-oxford-english-dictionary/12010628

dict

- key와 value를 쌍으로 갖는 기본 자료형
- key값을 이용하여 value값을 참조

key: unique & immutable

```
d = { key: value, key: value, …}
```

value: any datatype

```
# p 31. dict
dict_ex = {'하나':'one', '둘':{'two': 'duo'}, '셋': {"삼": 'trio'}}
dict_ex['하나']
```

Handling JSON data like a dict

```
docs['id']
```

```
'SDRW2100000002'
```

```
print("***metadata***")
print(docs['metadata'])
print("***metadata/title***")
print(docs['metadata']['title'])

docs_id = docs['id']
docs_meta = docs['metadata']
docs_title = docs['metadata']['title']
```

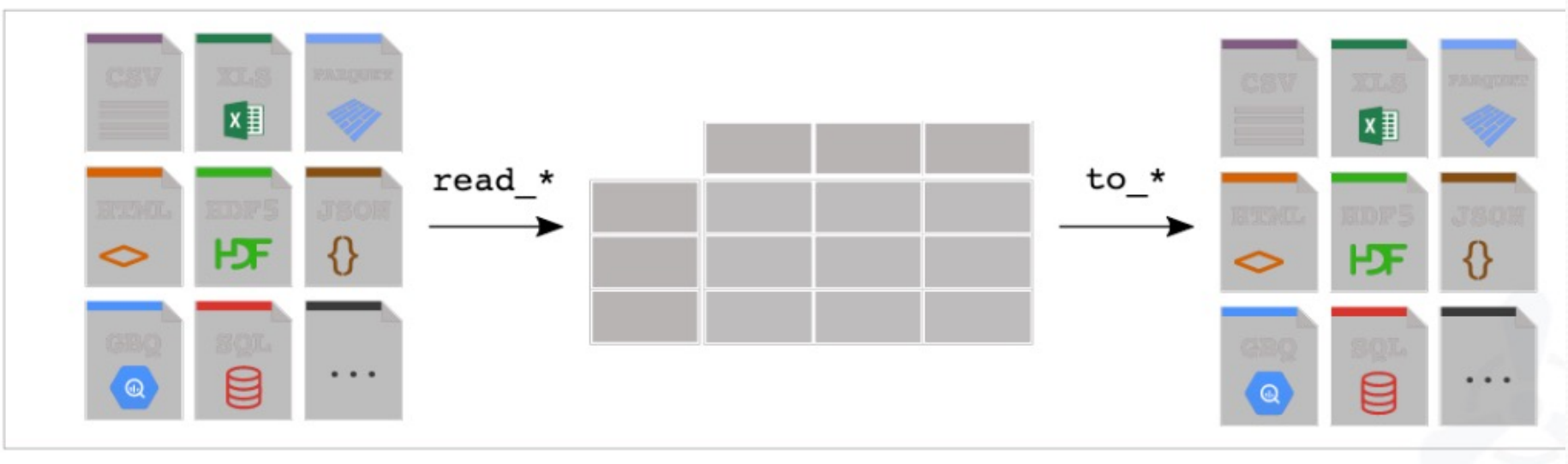
```
***metadata***
{'title': '국립국어원 구어 말뭉치 SDRW2100000002', 'creator': '국립국어원', 'distributor': '국립국어원',
'year': '2021', 'category': '구어 > 사적대화 > 일상대화', 'annotation_level': ['원시'], 'sampling':
'본문 전체'}
***metadata/title***
국립국어원 구어 말뭉치 SDRW2100000002
```

pandas 설치 및 사용법

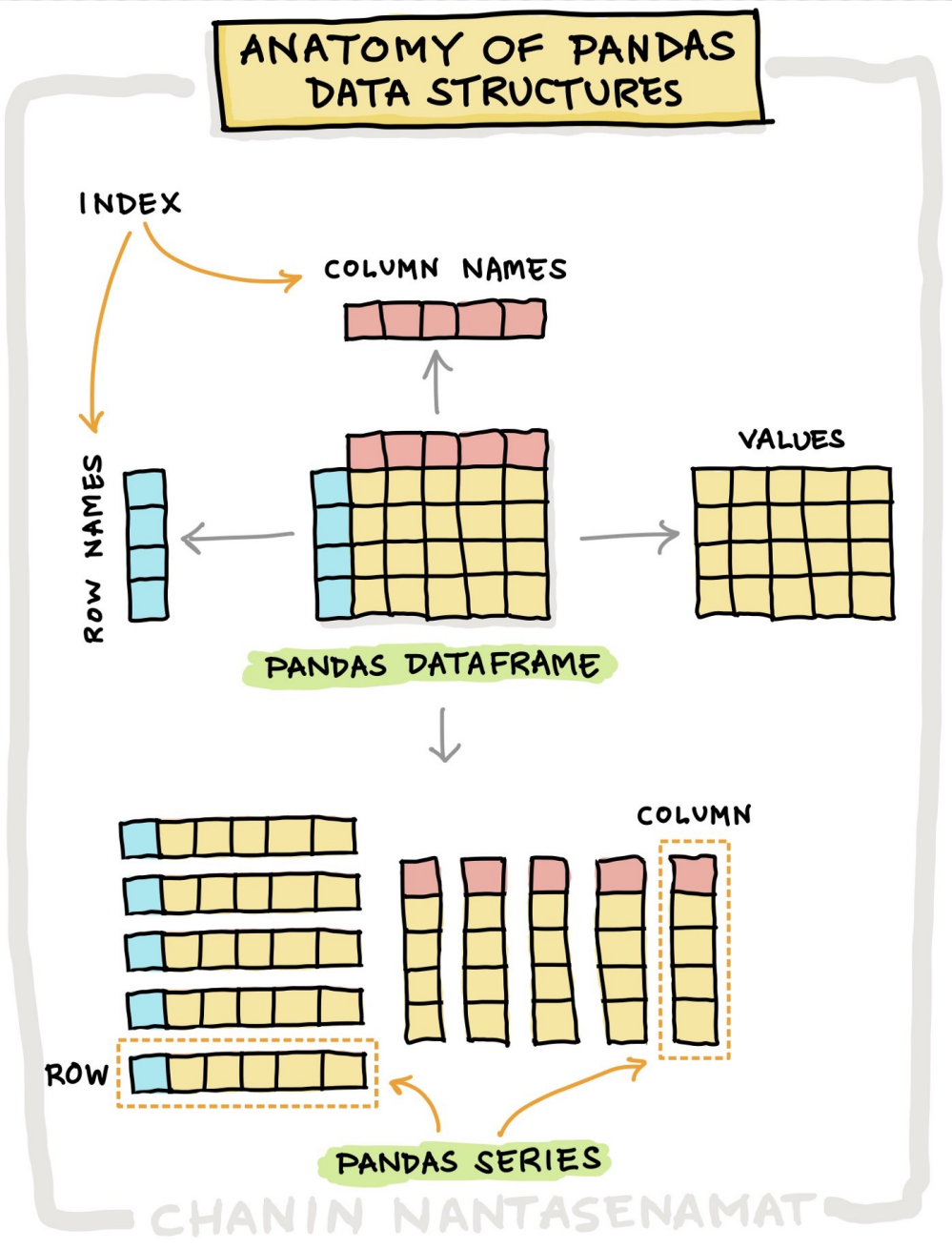
Pandas: Excel on Steroid

```
import pandas as pd
```

Pandas의 주요 특징
- 자유로운 데이터 변환
- 엑셀처럼 활용도 높은 DataFrame 객체
- 데이터 구조에 대한 변환



https://www.kdnuggets.com/2020/11/pandas-steroids-dask-python-data-science.html


ANATOMY OF PANDAS DATA STRUCTURES
INDEX
COLUMN NAMES
ROW NAMES
VALUES
PANDAS DATAFRAME
COLUMN
ROW
PANDAS SERIES
CHANIN NANTASENAMAT

A List to DataFrame

From List to DataFrame¶

```
import pandas as pd

list_name = ['item_1', 'item_2', 'item_3',
...]

df = pd.DataFrame(list_name, columns=
['column_name'])
```

https://datatofish.com/list-to-dataframe/

From List to DataFrame

```
import pandas as pd
sample_list = ['phonetics', 'phonology', 'syntax', 'morphology']
df = pd.DataFrame(sample_list, columns=['subject'])
df
```

	subject
0	phonetics
1	phonology
2	syntax
3	morphology

docs['id'] to DataFrame

```
import pandas as pd
docs_id_list = []
docs_id_list.append(docs['id'])
docs_id_df = pd.DataFrame(docs_id_list, columns = ['doc_id'])
docs_id_df
```

	doc_id
0	SDRW2100000001

metadata 처리하기

```
docs['metadata']
```

```
{'title': '국립국어원 구어 말뭉치 SDRW2100000001',
 'creator': '국립국어원',
 'distributor': '국립국어원',
 'year': '2021',
 'category': '구어 > 사적대화 > 일상대화',
 'annotation_level': ['원시'],
 'sampling': '본문 전체'}
```

```
docs['metadata']['title']
```

```
'국립국어원 구어 말뭉치 SDRW2100000001'
```

```
docs_title = docs['metadata']['title']
```

```
docs['metadata']['creator']
```

```
'국립국어원'
```

```
docs_creator = docs['metadata']['creator']
docs_creator
```

```
'국립국어원'
```

From JSON to LIST

```
docs_title_list = []; docs_creator_list = []
docs_title_list.append(docs_title)
docs_creator_list.append(docs_creator)
print("TITLE:", docs_title_list)
print("CREATOR:", docs_creator_list)
```

```
TITLE: ['국립국어원 구어 말뭉치 SDRW2100000002']
CREATOR: ['국립국어원']
```

Concatenating DataFrames

Concat(): DataFrame 붙이기

- pd.concat()함수는 데이터프레임을 말그대로 물리적으로 이어 붙여 주는 함수

	docs_title
0	국립국어원 구어 말뭉치 SDRW2100000001

	docs_creator
0	국립국어원

concat

	docs_title	docs_creator
0	국립국어원 구어 말뭉치 SDRW2100000001	국립국어원

https://yganalyst.github.io/data_handling/Pd_12/

```
docs_title_df = pd.DataFrame(docs_title_list,
                             columns = ['docs_title'])
docs_title_df
```

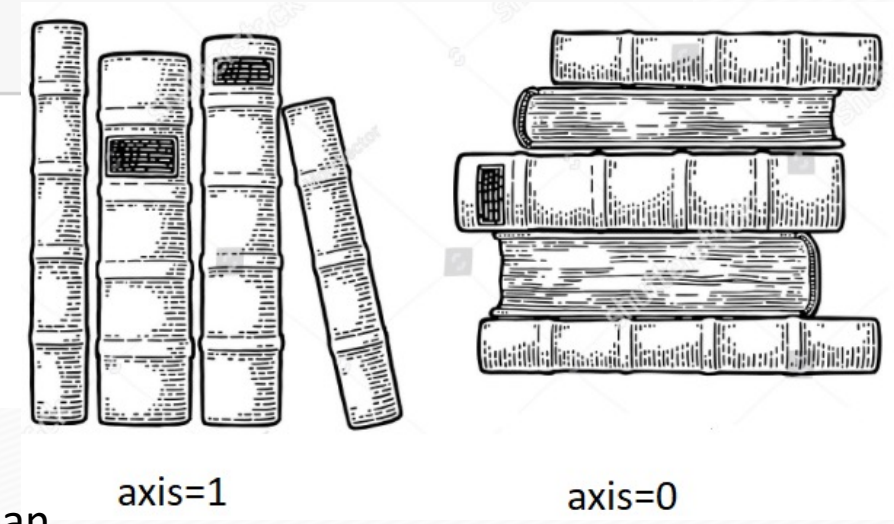
	docs_title
0	국립국어원 구어 말뭉치 SDRW2100000001

```
docs_creator_df = pd.DataFrame(docs_creator_list,
                               columns = ['docs_creator'])
docs_creator_df
```

	docs_creator
0	국립국어원

```
docs_metadata_df = pd.concat([docs_title_df, docs_creator_df],
                             axis=1)
docs_metadata_df
```

	docs_title	docs_creator
0	국립국어원 구어 말뭉치 SDRW2100000001	국립국어원



https://stackoverflow.com/questions/22149584/what-does-axis-in-pandas-mean

DIY

```
docs['metadata']
```

```
{'title': '국립국어원 구어 말뭉치 SDRW2100000001',
 'creator': '국립국어원',
 'distributor': '국립국어원',
 'year': '2021',
 'category': '구어 > 사적대화 > 일상대화',
 'annotation_level': ['원시'],
 'sampling': '본문 전체'}
```

list

document 처리하기

What is the datatype?

```
docs['document']
```

```
[{'id': 'SDRW2100000001.1',
  'metadata': {'title': '2인 일상 대화',
   'author': '개인 발화자',
   'publisher': '개인 발화 녹음',
   'date': '20210805',
   'topic': '음악 > 음악취향, 아이돌',
   'speaker': [{'id': 'SD2100001',
     'age': '20대',
     'occupation': '학생',
     'sex': '여성',
     'birthplace': '서울',
```

```
'utterance': [{'id': 'SDRW2100000001.1.1.1',
  'form': '너 그래 가지고 우리 저번에 호텔 갔을 때',
  'original_form': '너 그래 가지고 우리 저번에 호텔 갔을 때',
  'speaker_id': 'SD2100001',
  'start': '1.58000',
  'end': '5.30400',
  'note': ''},
 {'id': 'SDRW2100000001.1.1.421',
  'form': '스트리밍 하는 게 나을 것 같아.',
  'original_form': '쓰트리밍 하는 게 나을 것 같아.',
  'speaker_id': 'SD2100002',
  'start': '915.10000',
  'end': '917.48898',
  'note': '발화겹침'},
 {'id': 'SDRW2100000001.1.1.422',
  'form': '음',
  'original_form': '음',
  'speaker_id': 'SD2100001',
  'start': '917.47903',
  'end': '918.38900',
  'note': '발화겹침'}]}]
```

```
print(type(docs['document']))
```

```
<class 'list'>
```

Accessing data via indexing

```
docs['document'][0]
```

```
docs['document'][0]['metadata']['speaker']
```

```
[{'id': 'SD2100001',
  'age': '20대',
  'occupation': '학생',
  'sex': '여성',
  'birthplace': '서울',
  'principal_residence': '경기',
  'current_residence': '서울',
  'education': '대재'},
 {'id': 'SD2100002',
  'age': '20대',
  'occupation': '학생',
  'sex': '여성',
  'birthplace': '전남',
  'principal_residence': '서울',
  'current_residence': '서울',
  'education': '대재'}]
```

Accessing data via for-loop

```
for doc in docs['document']:
    print(doc)
    print('--'*10)
    print(doc['id'])
    print('---'*10)
    print(doc['metadata'])
```

{'id': 'SDRW2100000001.1', 'metadata': {'title': '2인 일상 대화', 'author': '개인 발화자', 'publisher': '개인 발화 녹음', 'date': '20210805', 'topic': '음악 > 음악취향, 아이돌', 'speaker': [{'id': 'SD2100001', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '서울', 'principal_residence': '경기', 'current_residence': '서울', 'education': '대재'}, {'id': 'SD2100002', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '전남', 'principal_residence': '서울', 'current_residence': '서울', 'education': '대재'}], 'setting': {'relation': '친구'}}, 'utterance': [{'id': 'SDRW2100000001.1.1.1', 'form': '너 그래 가지고 우리 저번에 호텔 갔을 때', 'original_form': '너 그래 가지고 우리 저번에 호텔 갔을 때', 'speaker_id': 'SD2100001', 'start': '1.58000', 'end': '5.30400', 'note': ''}, {'id': 'SDRW2100000001.1.1.421', 'form': '스트리밍 하는 게 나을 것 같아.', 'original_form': '쓰트리밍 하는 게 나을 것 같아.', 'speaker_id': 'SD2100002', 'start': '915.10000', 'end': '917.48898', 'note': '발화겹침'}, {'id': 'SDRW2100000001.1.1.422', 'form': '음', 'original_form': '음', 'speaker_id': 'SD2100001', 'start': '917.47903', 'end': '918.38900', 'note': '발화겹침'}]}

SDRW2100000001.1

{'title': '2인 일상 대화', 'author': '개인 발화자', 'publisher': '개인 발화 녹음', 'date': '20210805', 'topic': '음악 > 음악취향, 아이돌', 'speaker': [{'id': 'SD2100001', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '서울', 'principal_residence': '경기', 'current_residence': '서울', 'education': '대재'}, {'id': 'SD2100002', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '전남', 'principal_residence': '서울', 'current_residence': '서울', 'education': '대재'}], 'setting': {'relation': '친구'}}

Processing speakers

Accessing the speaker information

1

```
speaker = docs['document'][0]['metadata']['speaker']
print(speaker)
```

2

```
for doc in docs['document']:
    speakers = doc['metadata']['speaker']
    print(speakers)
```

```
[{'id': 'SD2100001', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '서울', 'p
rincipal_residence': '경기', 'current_residence': '서울', 'education': '대재'}, {'id': 'SD210000
2', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '전남', 'principal_residenc
e': '서울', 'current_residence': '서울', 'education': '대재'}]
```

- I want to make a speaker index.

```
for doc in docs['document']:
    speakers = doc['metadata']['speaker']
    print(speakers)
    print('='*20)
    speaker_index = 1
    for sp in speakers:
        print(f"speaker_{speaker_index}",
              sp['id'],
              sp['age'],
              sp['occupation'],
              sp['sex'],
              sp['birthplace'])
        speaker_index += 1
```

```
[{'id': 'SD2100001', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '서울', 'principal_residence': '경기', 'current_residence': '서울', 'education': '대재'}, {'id': 'SD2100002', 'age': '20대', 'occupation': '학생', 'sex': '여성', 'birthplace': '전남', 'principal_residence': '서울', 'current_residence': '서울', 'education': '대재'}]
====================
speaker_1 SD2100001 20대 학생 여성 서울
speaker_2 SD2100002 20대 학생 여성 전남
```

```
for doc in docs['document']:
    #speakers = doc['metadata']['speaker']
    speaker_index = 1
    for sp in speakers:
        print(f"speaker_{speaker_index}",
              sp['id'],
              sp['age'],
              sp['occupation'],
              sp['sex'],
              sp['birthplace'], end=" ")
        print(sp['principal_residence'],
              sp['current_residence'],
              sp['education'])
        speaker_index += 1
```

```
speaker_1 SD2100001 20대 학생 여성 서울 경기 서울 대재
speaker_2 SD2100002 20대 학생 여성 전남 서울 서울 대재
```

What if I want to print on the same line?

Speakers: From JSON to LIST

```
speaker_indices = []; speaker_ids = []; speaker_ages = []
speaker_occupations = []; speaker_principal_residences = []
speaker_sexes = []; speaker_birthplaces = []
speaker_current_residences = []; speaker_educations = []

for doc in docs['document']:
    #print(doc['metadata']['speaker'])
    speakers = doc['metadata']['speaker']
    #print(type(speakers), dir(list(speakers)))
    #print(speakers)
    speaker_index = 1

    speaker_list = []

    for sp in speakers:
        print(f"speaker_{speaker_index}", sp['id'], sp['age'], end=" ")
        print(sp['occupation'], sp['sex'], sp['birthplace'], end=" ")
        print(sp['principal_residence'], sp['current_residence'], end=" ")
        print(sp['education'])
        speaker_indices.append(f"speaker_{speaker_index}")
        speaker_ids.append(sp['id'])

        speaker_index += 1
print(speaker_indices, speaker_ids)
```

Speakers: From List to DataFrame¶

```
speaker_index_df = pd.DataFrame(speaker_indices,
                                columns = ['speaker_index'])
speaker_id_df = pd.DataFrame(speaker_ids,
                             columns = ['speaker_id'])
```

```
speaker_index_df
```

	speaker_index
0	speaker_1
1	speaker_2

```
speaker_id_df
```

	speaker_id
0	SD2100001
1	SD2100002

```
speaker_df = pd.concat([speaker_index_df, speaker_id_df], axis=1)
speaker_df
```

	speaker_index	speaker_id
0	speaker_1	SD2100001
1	speaker_2	SD2100002

Python

Jupyter Lab

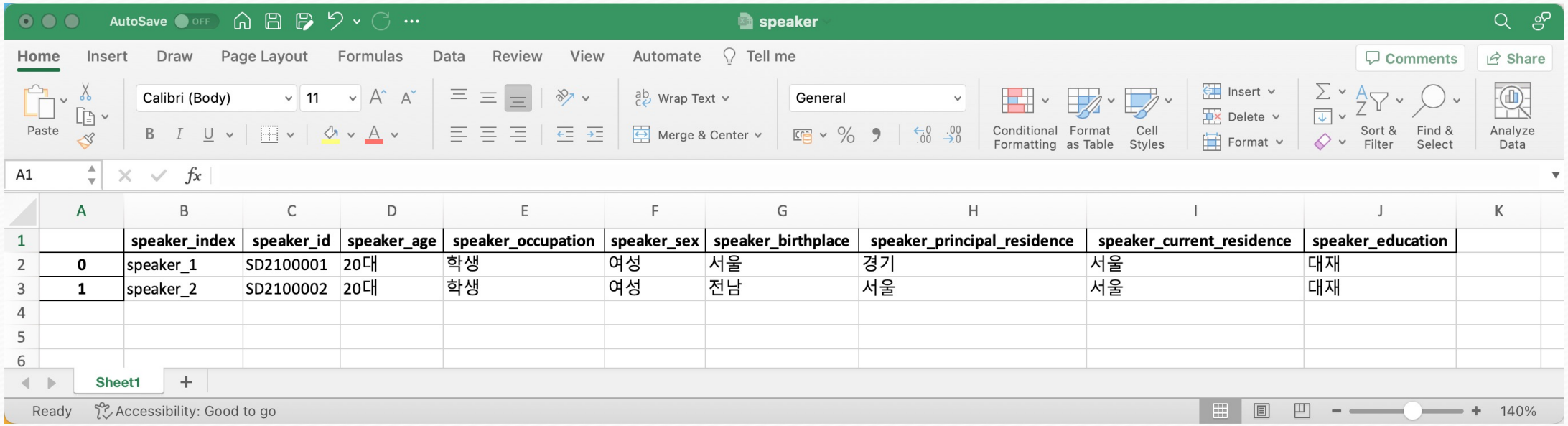
pandas
to_csv()

to_csv() &
to_excel()

Speakers: to_csv() & to_excel¶

```
speaker_df.to_csv('speaker.csv')
```

```
speaker_df.to_excel('speaker.xlsx')
```



	A	B	C	D	E	F	G	H	I	J	K
1		speaker_index	speaker_id	speaker_age	speaker_occupation	speaker_sex	speaker_birthplace	speaker_principal_residence	speaker_current_residence	speaker_education	
2	0	speaker_1	SD2100001	20대	학생	여성	서울	경기	서울	대재	
3	1	speaker_2	SD2100002	20대	학생	여성	전남	서울	서울	대재	
4											
5											
6											

Speakers: DIY

다른 화자 정보들도 DataFrame의 columns로 만들어 보기

- speaker_1 SD2100001 20대 학생 여성 서울 경기 서울 대재
- speaker_2 SD2100002 20대 학생 여성 전남 서울 서울 대재

Setting: DIY

setting 처리하기?

```
'setting': {'relation': '친구'}},
```

```
docs['document'][0]['metadata']['setting']
```

```
{'relation': '친구'}
```

```
for doc in docs['document']:
    setting = doc['metadata']['setting']
    print(setting)
```

```
{'relation': '친구'}
```

- Merge하는 부분에 대한 예를 보여주기 전...

```
# combining speaker_df and utterance_df
#########################################
final_df = pd.merge(speaker_df, utt_df, on="Speaker", how="outer")
df_lists.append(final_df)
final_df.to_excel(basename)
```

Utterance Processing

```
docs['document'][0]['utterance']
```


```
[{'id': 'SDRW2100000001.1.1.1',
  'form': '너 그래 가지고 우리 저번에 호텔 갔을 때',
  'original_form': '너 그래 가지고 우리 저번에 호텔 갔을 때',
  'speaker_id': 'SD2100001',
  'start': '1.58000',
  'end': '5.30400',
  'note': ''},
 {'id': 'SDRW2100000001.1.1.421',
  'form': '스트리밍 하는 게 나을 것 같아.',
  'original_form': '쓰트리밍 하는 게 나을 것 같아.',
  'speaker_id': 'SD2100002',
  'start': '915.10000',
  'end': '917.48898',
  'note': '발화겹침'},
 {'id': 'SDRW2100000001.1.1.422',
  'form': '음',
  'original_form': '음',
  'speaker_id': 'SD2100001',
  'start': '917.47903',
  'end': '918.38900',
  'note': '발화겹침'}]
```

```python
for doc in docs['document']:
    #print(doc['utterance'])
    utts = doc['utterance']
    utt_index = 1
    for utt in utts:
        print(utt)
        print('-'*60)
        utt_id = utt['id']
        utt_form = utt['form']
        utt_oform = utt['original_form']
        # for merge
        speaker_id = utt['speaker_id']
        utt_start = utt['start']
        utt_end = utt['end']
        utt_note = utt['note']
        utt_index += 1
        print(utt_id, '|', speaker_id, '|', utt_form, '|',
              utt_oform, '|', utt_start, '|', utt_end, '|',
              utt_note, '|' )
        print('-'*60)
```

```
{'id': 'SDRW2100000001.1.1.1', 'form': '너 그래 가지고 우리 저번에 호텔 갔을 때', 'original_form': '너 그래 가지고 우리 저번에 호텔 갔을 때', 'speaker_id': 'SD2100001', 'start': '1.58000', 'end': '5.30400', 'note': ''}
------------------------------------------------------------
SDRW2100000001.1.1.1 | SD2100001 | 너 그래 가지고 우리 저번에 호텔 갔을 때 | 너 그래 가지고 우리 저번에 호텔 갔을 때 | 1.58000 | 5.30400 |  |
------------------------------------------------------------
{'id': 'SDRW2100000001.1.1.421', 'form': '스트리밍 하는 게 나을 것 같아.', 'original_form': '쓰트리밍 하는 게 나을 것 같아.', 'speaker_id': 'SD2100002', 'start': '915.10000', 'end': '917.48898', 'note': '발화겹침'}
------------------------------------------------------------
SDRW2100000001.1.1.421 | SD2100002 | 스트리밍 하는 게 나을 것 같아. | 쓰트리밍 하는 게 나을 것 같아. | 915.10000 | 917.48898 | 발화겹침 |
------------------------------------------------------------
{'id': 'SDRW2100000001.1.1.422', 'form': '음', 'original_form': '음', 'speaker_id': 'SD2100001', 'start': '917.47903', 'end': '918.38900', 'note': '발화겹침'}
------------------------------------------------------------
SDRW2100000001.1.1.422 | SD2100001 | 음 | 음 | 917.47903 | 918.38900 | 발화겹침 |
------------------------------------------------------------
```

Utterance: from list to DataFrame

```
utt_ids = []; utt_forms = []; utt_oforms = []
utt_speaker_ids = []; utt_starts = []; utt_ends = []
utt_notes = []; utt_indices = []
for doc in docs['document']:
    utts = doc['utterance']
    utt_index = 1
    for utt in utts:
        utt_id = utt['id']
        utt_form = utt['form']
        utt_oform = utt['original_form']
        utt_speaker_id = utt['speaker_id']
        utt_start = utt['start']
        utt_end = utt['end']
        if utt['note'] == '': utt['note'] = "NA"
        utt_note = utt['note']
        utt_index += 1
        #print(utt_id, utt_speaker_id, utt_form, '|', utt_oform, '|',
        #utt_start, utt_end, utt_note)
        utt_ids.append(utt_id); utt_forms.append(utt_form);
        utt_oforms.append(utt_oform)
        utt_speaker_ids.append(utt_speaker_id);
        utt_starts.append(utt_start)
        utt_ends.append(utt_end);
        utt_notes.append(utt_note);
        utt_indices.append('utt_index')
utt_ids_df = pd.DataFrame(utt_ids, columns = ['utt_id'])
utt_forms_df = pd.DataFrame(utt_forms, columns = ['utt_form'])
utt_speaker_ids_df = pd.DataFrame(utt_speaker_ids,
                                  columns = ['speaker_id'])
```

Utterance: concatenation

```
utt_ids_df
```

	utt_id
0	SDRW2100000001.1.1.1
1	SDRW2100000001.1.1.421
2	SDRW2100000001.1.1.422

```
utt_forms_df
```

	utt_form
0	너 그래 가지고 우리 저번에 호텔 갔을 때
1	스트리밍 하는 게 나을 것 같아.
2	음

```
utt_speaker_ids_df
```

	speaker_id
0	SD2100001
1	SD2100002
2	SD2100001

```
utt_df = pd.concat([utt_ids_df, utt_speaker_ids_df, utt_forms_df], axis=1)
```

```
utt_df
```

	utt_id	speaker_id	utt_form
0	SDRW2100000001.1.1.1	SD2100001	너 그래 가지고 우리 저번에 호텔 갔을 때
1	SDRW2100000001.1.1.421	SD2100002	스트리밍 하는 게 나을 것 같아.
2	SDRW2100000001.1.1.422	SD2100001	음

Utterance: to_csv()

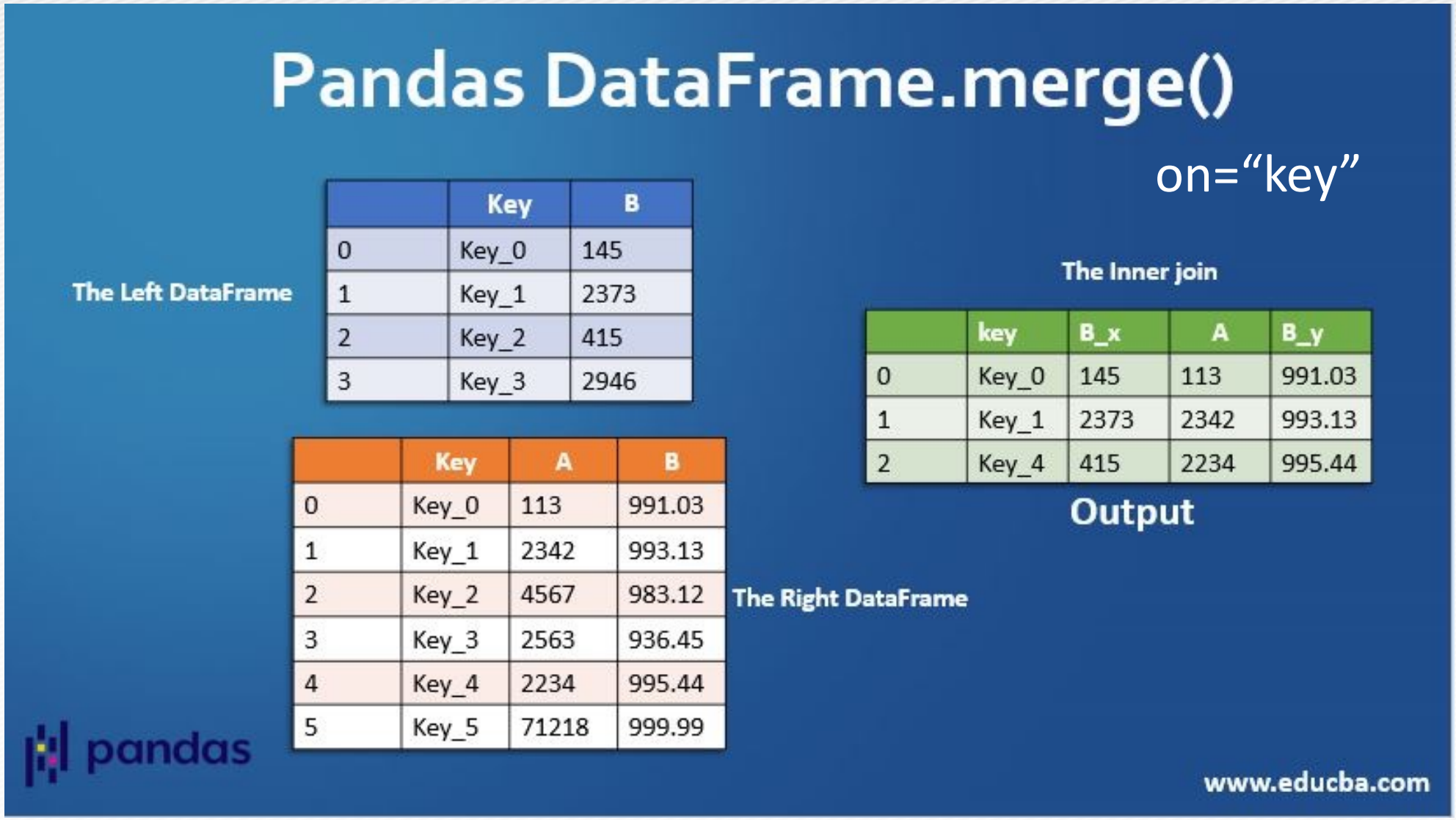
```
utt_df.to_csv('utterance.csv')
```

Merge speaker & utterance info

Merge

1. utterance dataframe에는 utt_speaker가 있고, 이를 토대로 speaker에 대한 정보를 붙여야 하지 않을까
2. 그렇다면 utt_df와 spk_df를 따로 처리한 후, utt_df의 speaker_id 정보를 토대로 spk_df의 speaker_id에 대한 정보를 가지고 와서 병합해야 하지 않을까?

pd.merge(): DataFrame 병합

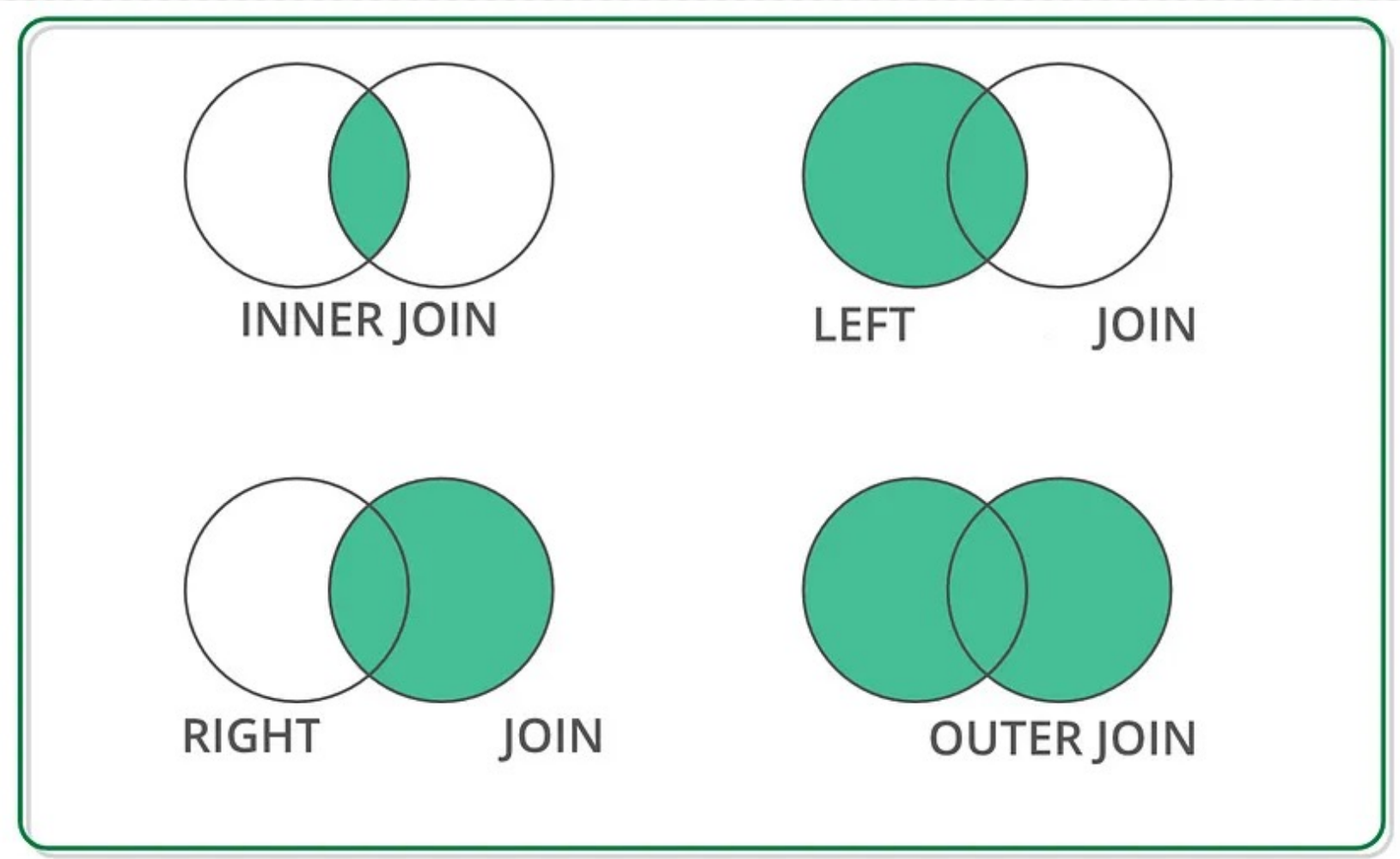


	Key	B
0	Key_0	145
1	Key_1	2373
2	Key_2	415
3	Key_3	2946

	key	B_x	A	B_y
0	Key_0	145	113	991.03
1	Key_1	2373	2342	993.13
2	Key_4	415	2234	995.44

	Key	A	B
0	Key_0	113	991.03
1	Key_1	2342	993.13
2	Key_2	4567	983.12
3	Key_3	2563	936.45
4	Key_4	2234	995.44
5	Key_5	71218	999.99

How to merge?



https://medium.com/swlh/merging-dataframes-with-pandas-pd-merge-7764c7e2d46d

Merge on 'speaker_id

```
utt_df
```

	utt_id	speaker_id	utt_form
0	SDRW2100000001.1.1.1	SD2100001	너 그래 가지고 우리 저번에 호텔 갔을 때
1	SDRW2100000001.1.1.421	SD2100002	스트리밍 하는 게 나을 것 같아.
2	SDRW2100000001.1.1.422	SD2100001	음

```
speaker_df
```

	speaker_index	speaker_id
0	speaker_1	SD2100001
1	speaker_2	SD2100002

```
utt_df.merge(speaker_df, on="speaker_id")
```

	utt_id	speaker_id	utt_form	speaker_index
0	SDRW2100000001.1.1.1	SD2100001	너 그래 가지고 우리 저번에 호텔 갔을 때	speaker_1
1	SDRW2100000001.1.1.422	SD2100001	음	speaker_1
2	SDRW2100000001.1.1.421	SD2100002	스트리밍 하는 게 나을 것 같아.	speaker_2

Incorrect!

```
pd.concat([docs_metadata_df, utt_df], axis=1)
```

	docs_title	docs_creator	utt_id	speaker_id	utt_form
0	국립국어원 구어 말뭉치 SDRW2100000001	국립국어원	SDRW2100000001.1.1.1	SD2100001	너 그래 가지고 우리 저번에 호텔 갔을 때
1	NaN	NaN	SDRW2100000001.1.1.421	SD2100002	스트리밍 하는 게 나을 것 같아.
2	NaN	NaN	SDRW2100000001.1.1.422	SD2100001	음

Putting meta info for each row of speaker or utterance info

```
doc_ids = []
utt_ids = []; utt_forms = []; utt_oforms = []
utt_speaker_ids = []; utt_starts = []; utt_ends = []
utt_notes = []; utt_indices = []
for doc in docs['document']:
    #print(doc['utterance'])
    utts = doc['utterance']
    utt_index = 1
    for utt in utts:
        #print(utt)
        utt_id = utt['id']
        utt_form = utt['form']
        utt_oform = utt['original_form']
        utt_speaker_id = utt['speaker_id']
        utt_start = utt['start']
        utt_end = utt['end']
        if utt['note'] == '': utt['note'] = "NA"
        utt_note = utt['note']
        utt_index += 1
        #print(utt_id, utt_speaker_id, utt_form, '|', utt_oform, '|', utt_start, utt_end, utt_note)
        utt_ids.append(utt_id); utt_forms.append(utt_form); utt_oforms.append(utt_oform)
        utt_speaker_ids.append(utt_speaker_id); utt_starts.append(utt_start)
        utt_ends.append(utt_end); utt_notes.append(utt_note); utt_indices.append('utt_index')
        # 기존의 docs_id
        doc_ids.append(docs['id'])

doc_ids_df = pd.DataFrame(doc_ids, columns = ['doc_id'])
utt_ids_df = pd.DataFrame(utt_ids, columns = ['utt_id'])
utt_forms_df = pd.DataFrame(utt_forms, columns = ['utt_form'])
utt_speaker_ids_df = pd.DataFrame(utt_speaker_ids, columns = ['utt_speaker_id'])
```

doc_id on each row

```
utt_df = pd.concat([doc_ids_df, utt_ids_df, utt_speaker_ids_df, utt_forms_df], axis=1)
```

```
utt_df
```

	doc_id	utt_id	utt_speaker_id	utt_form
0	SDRW2100000001	SDRW2100000001.1.1.1	SD2100001	너 그래 가지고 우리 저번에 호텔 갔을 때
1	SDRW2100000001	SDRW2100000001.1.1.421	SD2100002	스트리밍 하는 게 나을 것 같아.
2	SDRW2100000001	SDRW2100000001.1.1.422	SD2100001	음

DIY

- Can you do similarly with the other metadata?

glob

Handling files in a directory

- The glob command, short for *global*, originates in the earliest versions of Bell Labs' Unix.

```
import glob
glob.glob("data/*")
```

```
['data/SDRW2100000001',
 'data/SDRW2100000001.json',
 'data/SDRW2100000001_test.json',
 'data/SDRW2100000002',
 'data/SDRW2100000002.json']
```

```
for file in glob.glob('data/*.json'):
    print(file)
```

```
data/SDRW2100000001.json
data/SDRW2100000001_test.json
data/SDRW2100000002.json
```

```
for file in glob.glob('data/*[12].json'):
    print(file)
```

```
data/SDRW2100000001.json
data/SDRW2100000002.json
```

```
for file in glob.glob('data/*[12].json'):
    with open(file) as f:
        docs = json.loads(f.read())
    print(type(docs))
    print(docs)
    print("="*4000)
```

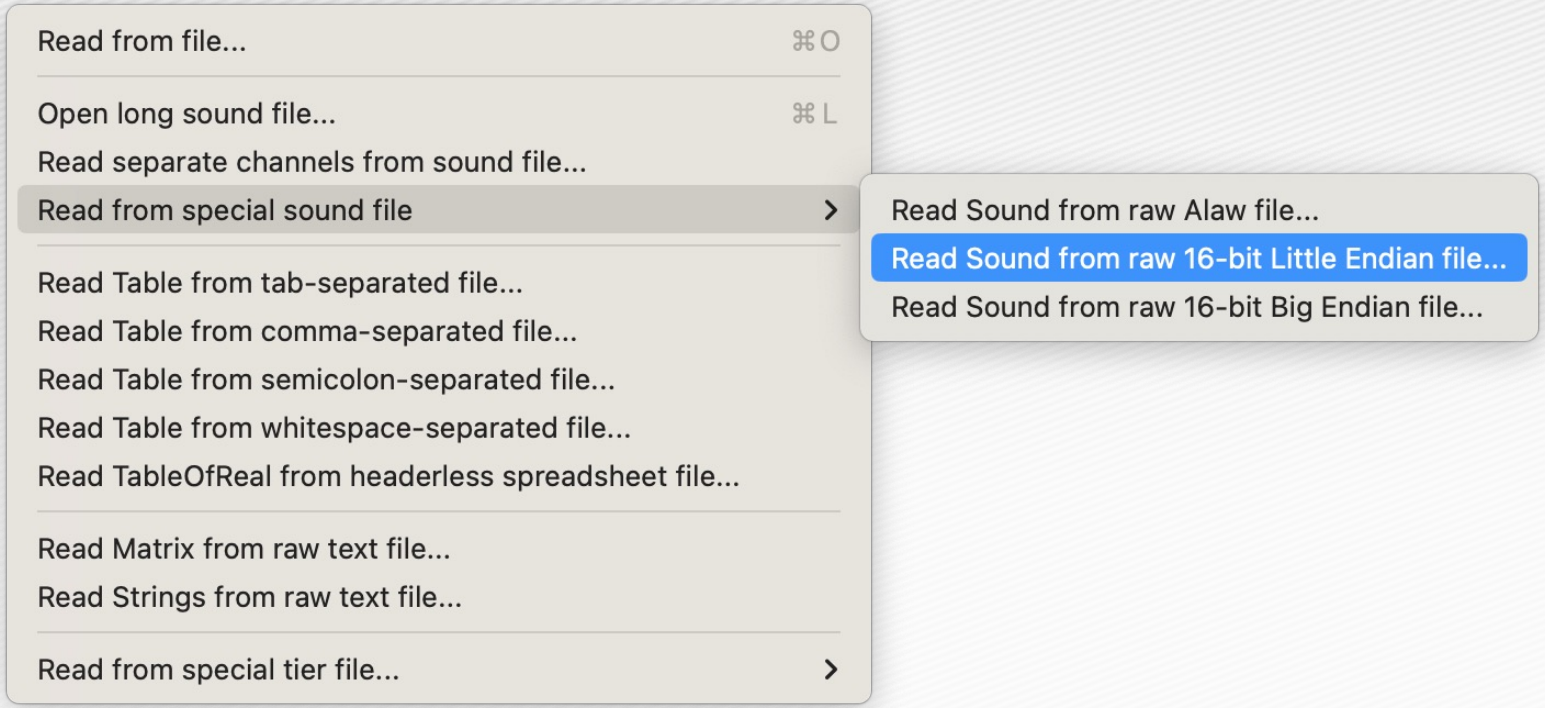
```
<class 'dict'>
{'id': 'SDRW2100000001', 'metadata': {'title': '국립국어원 구어 말뭉치 SDRW2100000001', 'creator': '국립국어원', 'distributo
r': '국립국어원', 'year': '2021', 'category': '구어 > 사적대화 > 일상대화', 'annotation_level': ['원시'], 'sampling': '본문 전
체'}, 'document': [{'id': 'SDRW2100000001.1', 'metadata': {'title': '2인 일상 대화', 'author': '개인 발화자', 'publisher':
'개인 발화 녹음', 'date': '20210805', 'topic': '음악 > 음악취향, 아이돌', 'speaker': [{'id': 'SD2100001', 'age': '20대', 'occu
pation': '학생', 'sex': '여성', 'birthplace': '서울', 'principal_residence': '경기', 'current_residence': '서울', 'educati
```

Putting All Together

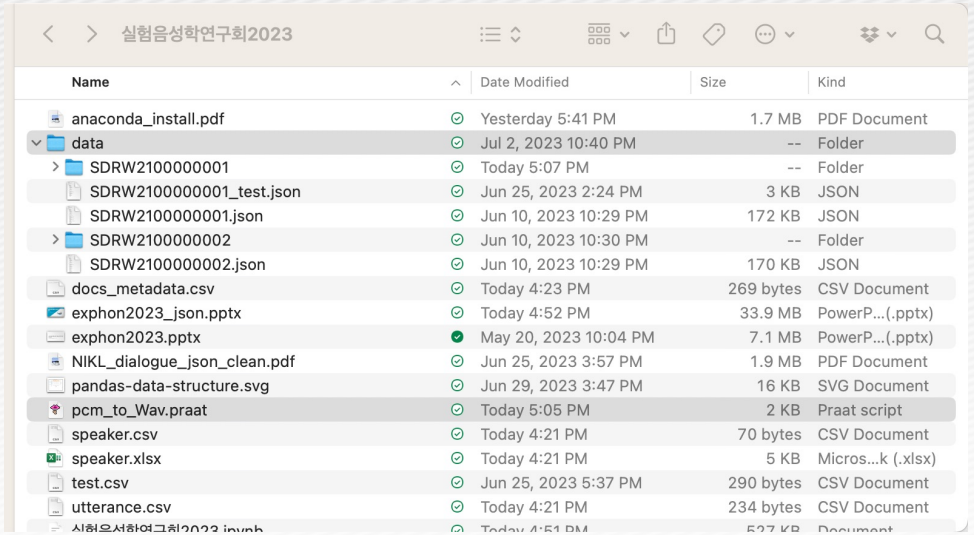
실험음성학연구회2023.ipynb

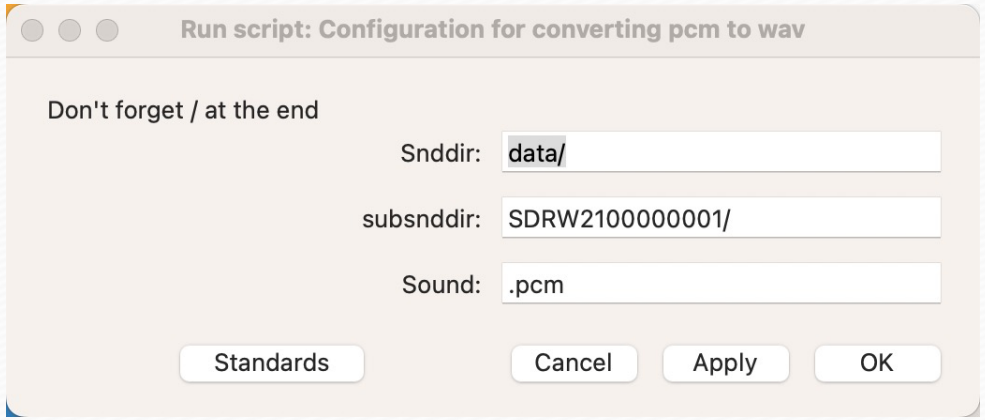
How to read pcm files?

How to open pcm file in Praat?



pcm_to_wave.praat





Script "/Users/tyoon/Dropbox/work/실험음성학연구회2023/pcm_to_Wav.praat"

File Edit Search Convert Font Run Help

```
form Configuration for converting pcm to wav
    comment Don't forget / at the end
    # 10000개 이상 처리하기는 힘듦. .이 _로 바뀜.
    sentence Snddir data/
    sentence subsnddir SDRW2100000001/
    sentence Sound .pcm
endform

clearinfo
Create Strings as directory list... dirlist 'snddir$'*1
num_dirs = Get number of strings
printline 'num_dirs'

for d from 1 to num_dirs
    select Strings dirlist
    dirname$ = Get string... 'd'
    printline 'dirname$'
    Create Strings as file list... filelist 'snddir$''dirname$'/*'sound$'
    num_files = Get number of strings
    printline 'num_files'
    for f from 1 to num_files
        select Strings filelist
        filename$ = Get string... 'f'
        basename$ = filename$-sound$
        printline 'snddir$''dirname$'/'filename$'
        Read Sound from raw 16-bit Little Endian file: "'snddir$''dirname$'/'filename$'"
        Save as WAV file: "'snddir$''dirname$'/'basename$'.wav"
        #selectObject: "Sound 'basename$'"
        Remove
    endfor
endfor
select all
Remove
printline "Converting pcm to wav files done..."
```

Thank you