

1. 원본 DataSet 불러오기

원본 csv를 utf-8-sig 인코딩으로 복사하기

```
from pathlib import Path

csvs = {
    "dataset/공공자전거 대여소 정보(25년 6월 기준).csv":
"dataset/usage_stations.csv",
    "dataset/서울특별시 공공자전거 이용정보(시간대별)_202506.csv":
"dataset/usage_time.csv",
    "dataset/서울특별시 공공자전거 이용정보(월별)_25.1-6.csv":
"dataset/usage_month.csv",
    "dataset/서울특별시 공공자전거 이용정보(일별)_2506.csv":
"dataset/usage_day.csv"
}

for src, dst in zip(csvs.keys(), csvs.values()):
    # utf-8로 읽고, utf-8-sig로 쓰면 BOM이 자동으로 붙음

        try:
            with Path(src).open("r", encoding="cp949", errors="strict",
newline="\n") as fin,
                Path(dst).open("w", encoding="utf-8-sig", errors="strict",
newline="\n") as fout:
                    for line in fin:
                        fout.write(line)
        except:
            with Path(src).open("r", encoding="utf-8", errors="strict",
newline="\n") as fin,
                Path(dst).open("w", encoding="utf-8-sig", errors="strict",
newline="\n") as fout:
                    for line in fin:
                        fout.write(line)
    print("done:", dst)

done: dataset/usage_stations.csv
done: dataset/usage_time.csv
done: dataset/usage_month.csv
done: dataset/usage_day.csv
```

2. 데이터 이해

1) 라이브러리 임포트

```
import numpy as np
import pandas as pd
```

2) csv 살펴보기

대여소별 이용정보: usage_stations

```
stations = pd.read_csv("dataset/usage_stations.csv")
stations.head()

   대여소 번호  보관소(대여소)명  자치구          상세주소  위도 \
0      301    경복궁역 7 번출구 앞 종로구  서울특별시 종로구 사직로 지하 130  경복궁역 7 번출
구 앞 37.575794
1      302    경복궁역 4 번출구 뒤 종로구  서울특별시 종로구 사직로 지하 130  경복궁역 4 번출
구 뒤 37.575947
2      303    광화문역 1 번출구 앞 종로구  서울특별시 종로구 세종대로 지하 189  세종로공원
37.571770
3      305    종로구청 옆 종로구  서울특별시 종로구 세종로 84-1  37.572559
4      307    서울역사박물관 앞 종로구  서울특별시 종로구 새문안로 55  서울역사박물관
앞 37.570000

   경도  설치 시기  QR  LCD  운영 방식
0  126.971451  2015-10-07  NaN  20.0    QR
1  126.974060  2015-10-07  NaN  12.0    QR
2  126.974663  2015-10-07  NaN   8.0    QR
3  126.978332  2015-01-07  NaN  16.0    QR
4  126.971100  2015-10-07  NaN  11.0    QR

# 결측치 열 확인
stations.isna().sum()

대여소 번호      0
보관소(대여소)명      0
자치구      0
상세주소      0
위도        0
경도        0
설치 시기      0
QR           1704
LCD          1070
운영 방식      0
dtype: int64
```

```
# 결측치 채우기  
stations.fillna(0, inplace=True)
```

이용정보 분석

```
usage_time = pd.read_csv("dataset/usage_time.csv")  
usage_day = pd.read_csv("dataset/usage_day.csv")  
usage_month = pd.read_csv("dataset/usage_month.csv")  
  
# 시간대별 이용정보  
usage_time.head()  
"""  
서울 열린데이터 광장 확인 결과( 시간대별 이용정보)  
- 대여시간: 시간대(1 시간 단위)  
- 이용건수, 운동량, 탄소량, 이동거리: 특정 대여 일자·시간·대여소·대여구분코드·성별·연령  
대조건이 같은 대여의 합산  
"""
```

분석해볼만한 것: 시간대별, 요일별 사용량

```
      대여일자 대여시간 대여소번호      대여소명 대여구분코드 성별 연령대코드 이용  
건수 \  
0  2025-06-01      0    1153  1153.  발산역 1 번, 9 번 인근 대여소  정기권 NaN ~10  
대  1  
1  2025-06-01      0     949      949.  연신내역 1 번 출구  정기권 NaN  20 대  
1  
2  2025-06-01      0    1029      1029.  성내어울터  정기권 NaN  20 대  
2  
3  2025-06-01      0    1150      1150.  송정역 1 번출구  정기권 NaN  20  
대  2  
4  2025-06-01      0     511      511.  서울숲역 5 번 출구 옆  정기권 NaN  20  
대  1  
  
      운동량 탄소량  이동거리  
0   15.86   0.14   616.00  
1   35.78   0.32  1390.00  
2   84.72   0.88  3780.26  
3   84.98   0.59  2552.56  
4   85.20   0.77  3310.00
```

```
# 일별 이용정보  
usage_day.head()
```

```
      대여일자 대여소번호      대여소 대여구분코드 성별 연령대 이용건수 운동량  
\  
0  2025-06-01    1027  1027.  프라자 아파트 앞  정기권 NaN ~10 대   1  
55.60
```

```

1 2025-06-01    1153  1153. 발산역 1 번, 9 번 인근 대여소 정기권 NaN ~10 대 1
15.86
2 2025-06-01    2376      2376. 수서역 6 번출구 앞 정기권 NaN ~10 대 1
11.71
3 2025-06-01    1967      1967. 참새공원(백곡경노당) 정기권 NaN ~10 대
1 69.18
4 2025-06-01    1160      1160. 양천향교역 7 번출구앞 정기권 NaN ~10 대 1
30.71

```

	탄소량	이동거리(M)	이용시간(분)
0	0.50	2160.08	70
1	0.14	616.00	8
2	0.13	580.00	4
3	0.62	2687.50	18
4	0.33	1410.00	12

월별 이용정보

```
usage_month.head()
```

```
"""

```

서울 열린데이터 광장 확인 결과(일별, 월별 이용정보)

- 대여시간: 시간대(1 시간 단위)
- 이용건수, 운동량, 탄소량, 이동거리, 이용시간: 특정 대여 일자·시간·대여소·대여구분코드
- 성별·연령대 조건이 같은 대여의 합산

```
"""

```

	대여일자	대여소번호	대여소명	대여구분코드	성별	연령대코드	이용건수	운동량	탄
0	202501	102	102.	망원역 1 번출구 앞	일일권	Nan	20 대	24	1510.39
15.41									
1	202501	102	102.	망원역 1 번출구 앞	일일권	Nan	30 대	22	1150.04
10.93									
2	202501	102	102.	망원역 1 번출구 앞	일일권	Nan	40 대	2	141.11
1.15									
3	202501	102	102.	망원역 1 번출구 앞	일일권	Nan	50 대	3	279.03
2.22									
4	202501	102	102.	망원역 1 번출구 앞	일일권	Nan	기타	7	858.12
8.29									

	이동거리(M)	이용시간(분)
0	66468.87	716
1	47128.28	649
2	4946.15	33
3	9584.22	80
4	35708.33	304

```

na_time = usage_time.isna().sum()
na_day = usage_day.isna().sum()
na_month = usage_month.isna().sum()

```

```
print(na_time[na_time != 0])
```

```
print(na_day[na_day != 0])
print(na_month[na_month != 0])

"""
알수 있는 것: 성별이 입력되지 않은 사용자 정보가 있음
운동량과 탄소량은 인과관계에 있음( 하나가 정해지면 나머지 하나가 정해짐)
조사 결과: 따릉이는 운동량과 사용자 정보( 연령, 몸무게 등)를 바탕으로 자동으로 탄소량을 계산함
"""


```

```
성별 944528
운동량 16772
탄소량 16772
dtype: int64
성별 588812
운동량 6695
탄소량 6695
dtype: int64
성별 185977
dtype: int64
```