

```
In [1]: # 라이브러리
import pandas as pd
import numpy as np
from datetime import datetime, timedelta
import os
```

```
In [2]: # 데이터 불러오기
seat_level=pd.read_csv('seat_level.csv')
```

```
In [3]: # 원본 데이터 유지
df=seat_level.copy()
```

```
In [4]: # 열 전체 출력 옵션
pd.set_option('display.max_columns', None)
```

## 시간 변경

```
In [5]: #날짜랑 시간 합치고 날짜형으로 변경
df['play_date_time']=df['real_play_date']+' '+df['play_st_time']
df['play_date_time'] = pd.to_datetime(df['play_date_time'], format='%Y-%m-%d %H:%M:%S')
```

```
In [6]: #시간이 변경 된 공연 모두 변경 후로 바꿈
date_pairs=[('2019-04-27T20:00:00.000000000', '2019-04-27T17:00:00.000000000'),
            ('2020-12-24T20:00:00.000000000', '2020-12-24T19:00:00.000000000'),
            ('2022-05-03T19:30:00.000000000', '2022-05-03T20:00:00.000000000'),
            ('2021-09-11T20:00:00.000000000', '2021-09-11T19:00:00.000000000'),
            ('2021-07-16T19:30:00.000000000', '2021-07-16T19:00:00.000000000'),
            ('2021-08-21T20:00:00.000000000', '2021-08-21T19:30:00.000000000'),
            ('2019-05-21T20:00:00.000000000', '2019-05-21T19:30:00.000000000')]

cols_to_change=['title', 'entry_age', 'hold', 'agency', 'sponser',
               'R_price', 'S_price', 'A_price', 'B_price', 'C_price', 'glee', 'n

for pair in date_pairs:
    incorrect_date, correct_date = pair
    for col in cols_to_change:
        df.loc[df['play_date_time']==incorrect_date,col]=df[df['play_date_time']==correct_date,col]
```

```
In [7]: # play_st_time도 변경 후 시간으로 통일
df.loc[df.real_play_date=='2019-04-27', 'play_st_time']='20:00:00'
df.loc[df.real_play_date=='2019-05-21', 'play_st_time']='20:00:00'
df.loc[df.real_play_date=='2020-12-24', 'play_st_time']='20:00:00'
df.loc[df.real_play_date=='2021-07-16', 'play_st_time']='19:30:00'
df.loc[df.real_play_date=='2021-08-21', 'play_st_time']='20:00:00'
df.loc[df.real_play_date=='2021-09-11', 'play_st_time']='20:00:00'
df.loc[df.real_play_date=='2022-05-03', 'play_st_time']='19:30:00'
```

## 할인 내역 정리

```
In [8]: # 오류 방지 데이터 복제
data=df.copy()
```

```
In [9]: # price가 0이 아니면서 할인내역에서 % 알 수 없지만 도메인으로 찾은 것들 채움
data.loc[data.discount_type.isin(['골드회원+음반패키지', '골드회원+음반 패키지', '골드
data.loc[data.discount_type.isin(['골드회원+음반패키지', '골드회원+음반 패키지', '골드
data.loc[data.discount_type.isin(['블랙프라이데이 이벤트']), 'discount_percent']=40
```

```

data.loc[data.discount_type.isin(['싹튀우미 할인']), 'discount_percent']=60
data.loc[data.discount_type.isin(['특판A']), 'discount_percent']=20
data.loc[data.discount_type.isin(['특판B']), 'discount_percent']=30
data.loc[data.discount_type.isin(['특판D']), 'discount_percent']=50
# origin_price1이 비어있기 때문에 할인을 이용하여 정가 복구
origin_price1=[]
for j in data.index.tolist():
    price = data.loc[j, 'price']
    calculated_price1 = price * (1/(1-(data.loc[j, 'discount_percent'] / 100))) if n
    origin_price1.append(calculated_price1)
data['origin_price1']=origin_price1
# 5000원 할인인 할인 내역 정가 복구
data.loc[data.discount_type.isin(['초/중/고/대학생 할인(본인만)']), 'origin_price1']=

```

```

In [10]: # 복원
df=data.copy()

```

## 공연정보를 변경

### 2019

```

In [11]: #2019-02-21 - 홈페이지 오류
df.loc[df.real_play_date=='2019-02-21', 'title']='KBS교향악단 제739회 정기연주회'
df.loc[df.real_play_date=='2019-02-21', 'R_price']=120000
df.loc[df.real_play_date=='2019-02-21', 'S_price']=90000
df.loc[df.real_play_date=='2019-02-21', 'A_price']=60000
df.loc[df.real_play_date=='2019-02-21', 'B_price']=30000
df.loc[df.real_play_date=='2019-02-21', 'C_price']=10000

```

```

In [12]: #2019-11-24 - 홈페이지 오류
df.loc[df.real_play_date=='2019-02-23', 'R_price']=121000
df.loc[df.real_play_date=='2019-02-23', 'S_price']=99000
df.loc[df.real_play_date=='2019-02-23', 'A_price']=77000
df.loc[df.real_play_date=='2019-02-23', 'B_price']=55000

```

```

In [13]: #2019-02-23 - 홈페이지 오류
df.loc[df.real_play_date=='2019-02-23', 'title']='라파우 블레하츠 & 김봄소리 듀오 콘서트'
df.loc[df.real_play_date=='2019-02-23', 'R_price']=121000
df.loc[df.real_play_date=='2019-02-23', 'S_price']=99000
df.loc[df.real_play_date=='2019-02-23', 'A_price']=77000
df.loc[df.real_play_date=='2019-02-23', 'B_price']=55000

```

### 2020

```

In [14]: #2020-02-01 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-01', 'title']='신영옥 데뷔 30주년 콘서트'
df.loc[df.real_play_date=='2020-02-01', 'R_price']=200000
df.loc[df.real_play_date=='2020-02-01', 'S_price']=150000
df.loc[df.real_play_date=='2020-02-01', 'A_price']=70000

```

```

In [15]: #2020-02-02 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-02', 'title']='영혼을 울리는 음악회 7'
df.loc[df.real_play_date=='2020-02-02', 'R_price']=100000
df.loc[df.real_play_date=='2020-02-02', 'S_price']=70000
df.loc[df.real_play_date=='2020-02-02', 'A_price']=50000
df.loc[df.real_play_date=='2020-02-02', 'B_price']=20000

```

```

In [16]: #2020-02-04 - 홈페이지 오류로 찾을 수 없음
df.loc[df.real_play_date=='2020-02-04','title']=''
df.loc[df.real_play_date=='2020-02-04','R_price']=120000
df.loc[df.real_play_date=='2020-02-04','S_price']=90000
df.loc[df.real_play_date=='2020-02-04','A_price']=60000
df.loc[df.real_play_date=='2020-02-04','B_price']=30000

In [17]: #2020-02-08 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-08','title']='너에게 들려주고 싶은 클래식이야기 -
df.loc[df.real_play_date=='2020-02-08','R_price']=50000
df.loc[df.real_play_date=='2020-02-08','S_price']=30000
df.loc[df.real_play_date=='2020-02-08','A_price']=20000
df.loc[df.real_play_date=='2020-02-08','B_price']=10000

In [18]: #2020-02-09 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-09','title']='너에게 들려주고 싶은 클래식이야기 -
df.loc[df.real_play_date=='2020-02-09','R_price']=70000
df.loc[df.real_play_date=='2020-02-09','S_price']=50000
df.loc[df.real_play_date=='2020-02-09','A_price']=30000
df.loc[df.real_play_date=='2020-02-09','B_price']=10000

In [19]: #2020-02-12 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-12','title']='제 12회 대한민국오페라대상 수상자 &
df.loc[df.real_play_date=='2020-02-12','R_price']=200000
df.loc[df.real_play_date=='2020-02-12','S_price']=150000
df.loc[df.real_play_date=='2020-02-12','A_price']=100000
df.loc[df.real_play_date=='2020-02-12','B_price']=50000
df.loc[df.real_play_date=='2020-02-12','C_price']=30000

In [20]: #2020-02-15 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-15','title']='레티시아모레노 with 뉴탱고퀸텟 피아
df.loc[df.real_play_date=='2020-02-15','R_price']=120000
df.loc[df.real_play_date=='2020-02-15','S_price']=90000
df.loc[df.real_play_date=='2020-02-15','A_price']=60000
df.loc[df.real_play_date=='2020-02-15','B_price']=30000

In [21]: #2020-02-18 - 홈페이지 오류로 찾을 수 없음
df.loc[df.real_play_date=='2020-02-18','title']=''
df.loc[df.real_play_date=='2020-02-18','R_price']=70000
df.loc[df.real_play_date=='2020-02-18','S_price']=50000
df.loc[df.real_play_date=='2020-02-18','A_price']=30000

In [22]: #2020-02-19 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-19','title']='제23회 뮤즈 윈드 오케스트라 정기연
df.loc[df.real_play_date=='2020-02-19','R_price']=100000
df.loc[df.real_play_date=='2020-02-19','S_price']=70000
df.loc[df.real_play_date=='2020-02-19','A_price']=50000
df.loc[df.real_play_date=='2020-02-19','B_price']=20000

In [23]: #2020-02-20 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-20','title']='French Fantasia II '
df.loc[df.real_play_date=='2020-02-20','R_price']=80000
df.loc[df.real_play_date=='2020-02-20','S_price']=60000
df.loc[df.real_play_date=='2020-02-20','A_price']=40000
df.loc[df.real_play_date=='2020-02-20','B_price']=20000

In [24]: #2020-02-21 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-21','title']='2020 서울시향 모차르트 교향곡 제36
df.loc[df.real_play_date=='2020-02-21','R_price']=70000
df.loc[df.real_play_date=='2020-02-21','S_price']=50000

```

```
df.loc[df.real_play_date=='2020-02-21', 'A_price']=30000
df.loc[df.real_play_date=='2020-02-21', 'B_price']=20000
df.loc[df.real_play_date=='2020-02-21', 'C_price']=10000
```

```
In [25]: #2020-02-22 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-22', 'title']='제37회 유나이티드필하모닉오케스트라
df.loc[df.real_play_date=='2020-02-22', 'normal']=20000
```

```
In [26]: #2020-02-23 - 홈페이지 오류
df.loc[df.real_play_date=='2020-02-23', 'title']="베토벤 최고의 걸작 '장엄미사'"
df.loc[df.real_play_date=='2020-02-23', 'R_price']=100000
df.loc[df.real_play_date=='2020-02-23', 'S_price']=70000
df.loc[df.real_play_date=='2020-02-23', 'A_price']=50000
df.loc[df.real_play_date=='2020-02-23', 'B_price']=30000
df.loc[df.real_play_date=='2020-02-23', 'C_price']=20000
```

```
In [27]: #2020-05-02 - 무관중
df.loc[df.real_play_date=='2020-05-02', 'R_price']=90000
df.loc[df.real_play_date=='2020-05-02', 'S_price']=70000
df.loc[df.real_play_date=='2020-05-02', 'A_price']=50000
df.loc[df.real_play_date=='2020-05-02', 'B_price']=30000
df.loc[df.real_play_date=='2020-05-02', 'C_price']=10000
```

```
In [28]: #2020-05-29 - 무관중
df.loc[df.real_play_date=='2020-05-29', 'R_price']=90000
df.loc[df.real_play_date=='2020-05-29', 'S_price']=70000
df.loc[df.real_play_date=='2020-05-29', 'A_price']=50000
df.loc[df.real_play_date=='2020-05-29', 'B_price']=30000
df.loc[df.real_play_date=='2020-05-29', 'C_price']=10000
```

```
In [29]: #2020-06-03 - 무관중, 예측
df.loc[df.real_play_date=='2020-06-03', 'R_price']=70000
df.loc[df.real_play_date=='2020-06-03', 'S_price']=50000
df.loc[df.real_play_date=='2020-06-03', 'A_price']=30000
df.loc[df.real_play_date=='2020-06-03', 'B_price']=10000
```

```
In [30]: #2020-06-26 - 3,4만원 추가
df.loc[df.real_play_date=='2020-06-26', 'R_price']=70000
df.loc[df.real_play_date=='2020-06-26', 'S_price']=50000
df.loc[df.real_play_date=='2020-06-26', 'A_price']=40000
df.loc[df.real_play_date=='2020-06-26', 'B_price']=30000
df.loc[df.real_play_date=='2020-06-26', 'C_price']=10000
```

```
In [31]: #2020-07-16 - 3,5만원 추가
df.loc[df.real_play_date=='2020-07-16', 'R_price']=90000
df.loc[df.real_play_date=='2020-07-16', 'S_price']=70000
df.loc[df.real_play_date=='2020-07-16', 'A_price']=50000
df.loc[df.real_play_date=='2020-07-16', 'B_price']=30000
df.loc[df.real_play_date=='2020-07-16', 'C_price']=10000
```

```
In [32]: #2020-07-17 무관중, 예측
df.loc[df.real_play_date=='2020-07-17',
      ['R_price', 'S_price', 'A_price', 'B_price']]=[70000, 50000, 30000, 10000]
```

```
In [33]: #2020-09-05 홈페이지에서 찾을 수 없음
df.loc[df.real_play_date=='2020-09-05',
      ['R_price', 'S_price', 'A_price']]=[130000, 110000, 55000]
```

```

In [34]: #2020-09-24 - 무관중, 예측
df.loc[df.real_play_date=='2020-09-24',['R_price', 'S_price', 'A_price', 'B_price',

In [35]: #2020-10-31 - 3,5만원 추가
df.loc[df.real_play_date=='2020-10-31',
        ['R_price', 'S_price', 'A_price', 'B_price', 'C_price', 'glee']]=[90000, 7000

In [36]: #2020-11-19 - 3,6만원 추가
df.loc[df.real_play_date=='2020-11-19',
        ['R_price', 'S_price', 'A_price', 'B_price', 'C_price', 'normal', 'glee']]=[1

In [37]: #2020-12-05 - 무관중
df.loc[df.real_play_date=='2020-12-05',
        ['R_price', 'S_price', 'A_price', 'B_price', 'C_price', 'normal']]=[90000, 70

In [38]: #2020-12-12 - 무관중
df.loc[df.real_play_date=='2020-12-12',
        ['R_price', 'S_price', 'A_price']]=[50000, 35000, 20000]

In [39]: #2020-12-16 - 무관중
df.loc[df.real_play_date=='2020-12-16',
        ['R_price', 'S_price', 'A_price']]=[50000, 35000, 20000]

In [40]: #2020-12-24 - 3,6만원 추가
df.loc[df.real_play_date=='2020-12-24',
        ['R_price', 'S_price', 'A_price', 'B_price', 'C_price', 'normal']]=[120000, 9

In [41]: #2020-12-25 - 전석초대로 나온는데 아마 무관중, 예측
df.loc[df.real_play_date=='2020-12-25',
        ['R_price', 'S_price', 'A_price']]=[80000, 60000, 40000]

```

## 2021

```

In [42]: #2021-08-25 - 홈페이지에 아무것도 나와있지 않음
df.loc[df.real_play_date=='2021-08-25',
        ['R_price', 'S_price']]=[30000, 15000]

In [43]: #2021-09-16 - 홈페이지에 아무것도 나와있지 않음
df.loc[df.real_play_date=='2021-09-16',
        ['R_price', 'S_price', 'A_price', 'B_price', 'C_price']]=[100000, 70000, 5000

In [44]: #2021-12-12 - 홈페이지 오류
df.loc[df.real_play_date=='2021-12-12',
        ['R_price', 'S_price', 'A_price', 'B_price']]=[110000, 90000, 70000, 50000]

```

## 2022

```

In [45]: #2022-05-15 - 홈페이지에 아무것도 안나옴
df.loc[df.real_play_date=='2022-05-15',
        ['R_price', 'S_price', 'A_price', 'B_price']]=[100000, 70000, 50000, 30000]

In [46]: #2022-05-28 - 홈페이지에 아무것도 안나옴
df.loc[df.real_play_date=='2022-05-28',
        ['R_price', 'S_price', 'A_price', 'B_price', 'C_price']]=[60000, 40000, 30000

```

```
In [47]: #2022-10-08 - 홈페이지에 아무것도 안나옴
df.loc[df.real_play_date=='2022-10-08',
       ['R_price', 'S_price', 'A_price', 'B_price']]=[110000, 90000, 70000, 50000]

In [48]: #2022-12-15 - 홈페이지에 아예 없음
df.loc[df.real_play_date=='2022-12-15',
       ['title', 'R_price', 'S_price', 'A_price', 'B_price']]=['파보 예르비 & 도이치']

In [49]: #2022-12-22 - 홈페이지에 가격이 안나옴
df.loc[df.real_play_date=='2022-12-22',
       ['R_price', 'S_price', 'A_price']]=[30000, 20000, 10000]

In [50]: #2022-12-23 - 홈페이지에 가격이 안나옴
df.loc[df.real_play_date=='2022-12-23',
       ['R_price', 'S_price', 'A_price', 'B_price', 'C_price']]=[150000, 120000, 100000, 80000, 60000]
```

## 2023

```
In [51]: #2023-02-19 - 홈페이지에 아예 없음
df.loc[df.real_play_date=='2023-02-19',
       ['title', 'R_price', 'S_price', 'A_price']]=['왕가위 영화음악 오케스트라 콘서트']
```

## 행을 변경

```
In [52]: # 50%로 적용
df.loc[(df.real_play_date=='2019-11-24') & (df.seat_level=='missing'), ['discount_percent', 'R_price', 'S_price', 'A_price', 'B_price', 'C_price']] = 50

In [53]: # 30%로 적용
df.loc[(df.real_play_date=='2021-05-30') & (df.discount_type.str.contains('장애')) & (df.seat_level=='missing'), ['discount_percent']] = 80

In [54]: # 30%로 적용
df.loc[(df.title=='KBS교향악단 제769회 정기연주회') & (df.seat_level=='missing'), ['discount_percent']] = 30

In [55]: # 30%로 적용
df.loc[(df.title=='KBS교향악단 제770회 정기연주회') & (df.seat_level=='missing'), ['discount_percent']] = 30

In [56]: # 30%로 적용
df.loc[(df.title=='KBS교향악단 제771회 정기연주회') & (df.seat_level=='missing'), ['discount_percent']] = 30

In [57]: # 30%로 적용
df.loc[(df.title=='KBS교향악단 제772회 정기연주회') & (df.seat_level=='missing'), ['discount_percent']] = 30

In [58]: # 30%로 적용
df.loc[(df.title=='KBS교향악단 제773회 정기연주회') & (df.seat_level=='missing'), ['discount_percent']] = 30

In [59]: # 30%로 적용
df.loc[(df.title=='2022 서울국제음악제 SIMF오케스트라 폐막음악회 with 오코 카무') & (df.seat_level=='missing'), ['discount_percent', 'seat_level']] = [30, 'R_price']
```

```

In [60]: # 30%로 적용
df.loc[(df.title=='미클로시 페레니 & 피닌 콜린스 듀오 콘서트 I SAC기획')&(df.seat_level=='discount_percent')]=30

In [61]: # 30%로 적용
df.loc[(df.title=='미클로시 페레니 & 피닌 콜린스 듀오 콘서트 II SAC기획')&(df.seat_level=='discount_percent')]=30

In [62]: # R석 50% 적용
df.loc[(df.title=='루돌프 부흐빈더 베토벤 피아노 소나타 전곡 I')&(df.price==65000)
        ['discount_percent','seat_level']]=[50, 'R_price']

In [63]: # S석 복구 - 골드회원 10%인데 50%인거
df.loc[(df.title=='루돌프 부흐빈더 베토벤 피아노 소나타 전곡 I')&(df.seat_level=='discount_percent', 'seat_level')]=[50, 'S_price']

In [64]: # S석 50% 적용
df.loc[(df.title=='루돌프 부흐빈더 베토벤 피아노 소나타 전곡 I')&(df.price==50000)
        ['discount_percent', 'seat_level']]=[50, 'S_price']

In [65]: # R석 50% 적용
df.loc[(df.title=='루돌프 부흐빈더 베토벤 피아노 소나타 전곡 II')&(df.price==65000)
        ['discount_percent','seat_level']]=[50, 'R_price']

In [66]: # S석 복구 - 골드회원 10%인데 50%인거
df.loc[(df.title=='루돌프 부흐빈더 베토벤 피아노 소나타 전곡 II')&(df.seat_level=='discount_percent', 'seat_level')]=[50, 'S_price']

In [67]: # S석 50% 적용
df.loc[(df.title=='루돌프 부흐빈더 베토벤 피아노 소나타 전곡 II')&(df.price==50000)
        ['discount_percent', 'seat_level']]=[50, 'S_price']

In [68]: #정가에 10000원, 20000원 비어서 결제 된것 복구
for index in df[(df.real_play_date=='2021-11-07')&(df.seat_level=='missing')].index:
    if df.loc[index, 'origin_price']==100000:
        df.loc[index, 'price']=(df.loc[index, 'origin_price']+20000)*(1-df.loc[index, 'discount_percent'])
    else:
        df.loc[index, 'price']=(df.loc[index, 'origin_price']+10000)*(1-df.loc[index, 'discount_percent'])

In [69]: # 15%로 적용
df.loc[(df.title=='손열음 모차르트 피아노 소나타 전곡 리사이틀 (5.2)')&(df.seat_level=='discount_percent')]=15

In [70]: # 15%로 적용
df.loc[(df.title=='손열음 모차르트 피아노 소나타 전곡 리사이틀 (5.6)')&(df.seat_level=='discount_percent')]=15

```

## 기부 지우기

```

In [71]: # 기부콘서트 - 현정
df_dona=df[df['real_play_date']=='2020-05-21']

In [72]: df=df[~(df['real_play_date']=='2020-05-21')]

```

## 할인 역적용

```
In [73]: # 추가된 할인을 이 있기 때문에 다시 할인 역적용
origin_price1=[]

for j in df.index.tolist():
    price = df.loc[j, 'price']
    calculated_price1 = price * (1/(1-(df.loc[j, 'discount_percent'] / 100))) if not
    origin_price1.append(calculated_price1)

df['origin_price1']=origin_price1
```

## 좌석 등급 매기기

```
In [76]: %%time

# missing이 없는지 확인
data_d=df.copy()
import numpy as np
seat_level=[None]*data_d.shape[0]

for j in range(data_d.shape[0]):
    if (np.isnan(data_d.iloc[j,27]))&(set(data_d.iloc[j,36:47].tolist())=={0}):
        seat_level[j]='free'
    elif (np.isnan(data_d.iloc[j,27]))&(set(data_d.iloc[j,36:47].tolist())!={0}):
        seat_level[j]='unknown'
    else:
        diff=[(i-int(data_d.iloc[j,27])) for i in data_d.iloc[j,36:47].tolist() if
        if len([i for i in diff if ((i>=0)&(i<=2000)))==1:
            diff_value=[i for i in diff if ((i>=0)&(i<=2000))][0]
            seat_level[j]=data_d.columns[(data_d.iloc[j,36:47].tolist().index(int(diff_value)))]
        elif len([i for i in diff if ((i>=0)&(i<=2000)))==0:
            seat_level[j]='missing'
        else:
            seat_level[j]='more'

data_d['seat_level']=seat_level
data_d.loc[(data_d['seat_level']=='unknown')&(data_d['price']==0), 'seat_level']='free'

CPU times: total: 6min 16s
Wall time: 13min 5s
```

```
In [77]: df=data_d.copy()
```

```
In [78]: # missing 개수
len(df[(df.price!=0)&(df.seat_level=='missing')])
```

```
Out[78]: 0
```

## 저장

```
In [344... df.to_csv('unmissing_final.csv', index=False)
```