

In [1]: import pandas as pd  
import numpy as np

## Merge()

In [8]: # 첫 번째 DataFrame 생성  
df1 = {  
 'ID': [1, 2, 3, 4],  
 '이름': ['김철수', '이영희', '박지민', '최수빈'],  
 '도시': ['서울', '부산', '대구', '인천']  
}  
df1 = pd.DataFrame(data1)  
  
# 두 번째 DataFrame 생성  
df2 = {  
 'ID': [1, 2, 3, 5],  
 '직업': ['개발자', '디자이너', '회계사', '마케터'],  
 '급여': [5000, 4500, 6000, 5500] }  
  
df2 = pd.DataFrame(data2)

In [ ]:

## how

In [11]: # DataFrame 1 (위와 동일)  
data1 = {  
 'ID': [1, 2, 3, 4],  
 '이름': ['김철수', '이영희', '박지민', '최수빈'],  
 '도시': ['서울', '부산', '대구', '인천']  
}  
df1 = pd.DataFrame(data1)  
  
# DataFrame 2 (위와 동일)  
data2 = {  
 'ID': [1, 2, 3, 5],  
 '직업': ['개발자', '디자이너', '회계사', '마케터'],  
 '급여': [5000, 4500, 6000, 5500]  
}  
df2 = pd.DataFrame(data2)

In [39]: #left

In [40]: #right

In [41]: #outer

## Concat

In [15]: # 첫 번째 DataFrame 생성  
data1 = {  
 '이름': ['김철수', '이영희'],  
 '나이': [25, 30],  
 '도시': ['서울', '부산']  
}  
df1 = pd.DataFrame(data1)  
  
# 두 번째 DataFrame 생성  
data2 = {  
 '이름': ['박지민', '최수빈'],  
 '나이': [22, 28],  
 '도시': ['대구', '인천']  
}  
df2 = pd.DataFrame(data2)

In [42]: # 두 DataFrame을 세로(행) 방향으로 이어 붙이기 (기본값: axis=0)

In [43]: # 인덱스 재설정 (reset\_index)

## Groupby & Agg()

In [19]: # 데이터프레임 생성  
data = {  
 '상점명': ['송도점', '부평점', '구월점', '송도점', '부평점', '구월점', '송도점', '부평점', '구월점', '송도점'],  
 '제품카테고리': ['음료', '스낵', '음료', '베이커리', '음료', '스낵', '음료', '베이커리', '음료', '스낵'],  
 '제품명': ['콜라', '새우깡', '사이다', '단팥빵', '커피', '포카칩', '주스', '소보로빵', '녹차', '오징어땅콩'],  
 '판매수량': [10, 15, 8, 5, 12, 20, 7, 6, 9, 18],  
 '단가': [1500, 1200, 1400, 2000, 2500, 1800, 2200, 2300, 1700, 1300],  
 '판매일': pd.to\_datetime([  
 '2025-06-01', '2025-06-01', '2025-06-02', '2025-06-02', '2025-06-03',  
 '2025-06-03', '2025-06-04', '2025-06-04', '2025-06-05', '2025-06-05'  
 ])  
}  
df = pd.DataFrame(data)  
  
# '총판매액' 컬럼 추가  
df['총판매액'] = df['판매수량'] \* df['단가']

In [44]: # 각 상점별 총 판매액 계산

In [45]: # 제품 카테고리별 평균 판매수량 및 최대 단가

In [46]: # 상점별, 제품 카테고리별 판매 건수

In [47]: # 가장 많이 팔린 제품 TOP 3 (상점 무관)

In [48]: # 날짜별 총 판매액 및 판매수량

## Value\_counts()

In [27]: # 가상의 판매 데이터 생성  
data = {  
 '상점명': ['송도점', '부평점', '구월점', '송도점', '부평점', '구월점', '송도점', '부평점', '송도점', '연수점'],  
 '제품카테고리': ['음료', '스낵', '음료', '베이커리', '음료', '스낵', '음료', '베이커리', '음료', '음료'],  
 '결제수단': ['카드', '현금', '카드', '현금', '카드', '현금', '카드', '현금', '카드', '카드'],  
 '판매수량': [10, 15, 8, 5, 12, 20, 7, 6, 9, 11]  
}  
df = pd.DataFrame(data)

In [49]: #결제수단

In [50]: # 제품카테고리' 컬럼의 상위 2개 카테고리만 출력

## Cut

In [34]: # 가상의 학생 점수 데이터 생성  
data = {  
 '학생ID': range(101, 111),  
 '점수': [65, 88, 72, 95, 50, 78, 81, 60, 90, 75],  
 '학년': [1, 2, 1, 3, 1, 2, 3, 1, 2, 3]  
}  
df = pd.DataFrame(data)

In [51]: # 점수를 4개 구간으로 나누기 (자동 계산)

In [52]: # 점수를 특정 기준에 따라 학점으로 분류  
# right=True가 기본값이므로, 60점은 D에 포함됩니다. (60점 이상 70점 미만)

In [53]: # 점수를 특정 기준에 따라 학점으로 분류 (오른쪽 경계 포함 안함)  
# right=False로 설정하여 오른쪽 경계를 포함하지 않습니다. (예: 60점은 F에 포함)

In [54]: # 점수를 등급으로 나누고 구간 라벨 지정 (정렬 없음)  
# ordered=False를 사용하여 카테고리 순서를 강제하지 않습니다.