

## [데이터프레임과 시리즈 (Pandas)-I]

### 1. Pandas 패키지

- 1차원 구조를 갖는 시리즈, 2차원 구조를 갖는 프레임 제공
- pandas 패키지 불러오기: **import pandas as pd** (pandas 패키지는 주로 pd라는 별칭으로 자주 사용됨)

### 2. 데이터프레임 만드는 방법: **pd.DataFrame(df)**

- (1) 딕셔너리 / 리스트를 이용하여 만들기
- (2) csv파일 불러오기: **pd.read\_csv("파일경로/파일명.csv")**
- (3) sklearn.datasets 모듈 데이터를 데이터프레임으로 변환  
cf) **from sklearn import datasets** : sklearn.datasets 불러오기

### 3. 행과 열의 이름 지정하기

- (1) 열 이름 지정: **df.columns = ['바꿀 열 이름1', '바꿀 열 이름2', ... ]**
- (2) 행 이름 지정: **df.index = ['바꿀 행 이름1', '바꿀 행 이름2', ... ]**

### 4. 부분 데이터 조회

- (1) **df.loc['행의 범위', '열의 범위']** : columns, index로 조회
  - (2) **df.iloc[from, to, by]** : 인덱스 번호로 조회 / from부터 to전까지(to미포함) by씩 증가
  - (3) **df.loc[조건]** : 조건으로 조회하기
- ex) iris\_df 데이터 프레임에서 Species가 virginica인 행만 출력하기 => **iris\_df.loc[iris\_df.Species=='virginica']**

### 5. 데이터 추가와 삭제

- (1) 데이터프레임 요소 삭제: **df.drop(삭제할위치, axis= )** # axis=0 행별 / axis= 1 열별
- (2) 데이터프레임 요소 추가:  
ex) Series를 이용한 열 추가 - member\_df 데이터프레임에 Age 열 추가하기  
**member\_df["Age"] = pd.Series([20, 25, 30], index = [0, 1, 3])**

### 6. 정렬:

**df.sort\_index(inplace=False, axis= , ascending=True)**  
# **inplace=False**(기본값): 실제 데이터엔 적용되지 않는다. / **ascending=True**(기본값): 오름차순

### 7. 기초 통계 분석에 쓰이는 함수들

<b>count</b>	결측치를 제외한 데이터의 수	<b>min / max</b>	최소값 / 최대값
<b>sum / cumprod</b>	합 / 누적합	<b>mean / median</b>	평균 / 중앙값
<b>quantile</b>	분위 수	<b>corr</b>	상관관계
<b>var / std</b>	분산 / 표준편차	<b>describe</b>	요약 통계량 제시