Pandas 치트시트

Pandas란?

Pandas는 Python에서 데이터 조작 및 분석을 위한 핵심 라이브러리입니다. `Series` (1차원)와 `DataFrame` (2차원)이라는 두 가지 주요 데이터 구조를 제공합니다.

```
import pandas as pd
```

데이터 구조 생성

Series

```
s = pd.Series([1, 3, 5, np.nan, 6, 8])
```

DataFrame

- 딕셔너리로부터 생성:

```
data = {'Name': ['Tom', 'Nick', 'Juli'], 'Age': [20, 21, 19]}
df = pd.DataFrame(data)
```

- CSV 파일 읽기: `df = pd.read_csv('data.csv')`
- Excel 파일 읽기: `df = pd.read_excel('data.xlsx')`

데이터 확인 및 탐색

- `df.head(n)`: 처음 n개의 행 보기 (기본값 5)
- `df.tail(n)`: 마지막 n개의 행 보기
- `df.info()`: DataFrame의 요약 정보 (인덱스, 컬럼, non-null 값, 메모리 사용량)
- `df.describe()`: 수치형 데이터의 기술 통계 요약
- `df.shape`: (행, 열)의 개수 (튜플)
- `df.columns`: 컬럼 이름 목록
- `df.index`: 인덱스 정보
- `df.sort_values(by='column_name')`: 특정 컬럼 기준으로 정렬

데이터 선택 (Selection)

컬럼 선택

- `df['column_name']` (Series 반환)
- `df[['col1', 'col2']]` (DataFrame 반환)

행 선택

- `df[0:3]`: 0부터 2번 인덱스까지의 행 슬라이싱

라벨 기반 선택 (`.loc`)

- `df.loc[label]`: 단일 행 선택
- `df.loc[:, ['A', 'B']]`: 모든 행의 A, B 컬럼 선택
- `df.loc['20230102':'20230104', ['A', 'B']]`: 날짜 범위와 컬럼으로 선택

위치 기반 선택 (`.iloc`)

- `df.iloc[3]`: 3번 위치의 행 선택
- `df.iloc[3:5, 0:2]`: 행과 열을 정수 위치로 슬라이싱

조건부 선택

- `df[df['A'] > 0]`: A 컬럼의 값이 0보다 큰 행만 선택
- `df[df['country'].isin(['USA', 'Canada'])]`: 여러 조건 중 하나라도 만족하는 행 선택

데이터 조작

결측치 처리 (Missing Data)

- `df.dropna()`: 결측치가 있는 행 제거
- `df.fillna(value=5)`: 결측치를 특정 값으로 채우기
- `pd.isna(df)`: 결측치 여부를 불리언 값으로 확인

데이터 변경

- `df['C'] = df['A'] + df['B']`: 새 컬럼 추가
- `df.at[index, 'column'] = value`: 특정 위치의 값 변경 (빠름)
- `df.iat[row_pos, col_pos] = value`: 정수 위치로 값 변경 (빠름)
- `df.apply(np.cumsum)`: 함수 적용

GroupBy

데이터를 그룹으로 묶어 집계 연산을 수행.

```
df.groupby('A').sum()
df.groupby(['A', 'B']).mean()
```

데이터 병합 (Merge, Concat, Join)

Concat

```
pd.concat([df1, df2])
```

Merge

SQL 스타일의 조인.

```
pd.merge(left, right, on='key', how='inner')
```

- `how`: `inner`, `left`, `right`, `outer`

데이터 입출력

- CSV로 저장: `df.to_csv('output.csv', index=False)`
- Excel로 저장: `df.to_excel('output.xlsx', sheet_name='Sheet1')`

시계열 데이터 (Time Series)

- 날짜 범위 생성: `dates = pd.date_range('20230101', periods=6)`
- 리샘플링 (Resampling): `ts.resample('M').sum()` (월별 합계)

피벗 테이블 (Pivot Tables)

```
pd.pivot_table(df, values='D', index=['A', 'B'], columns=['C'])
```