|  |  |
| --- | --- |
| 교육 제목 | **Part.03 데이터 살펴보기 & part.04 시각화 도구** |
| 교육 일시 | 21.09.30 |
| 교육 장소 | 온라인 (집) |
| **교육 내용** | |
| 오전 | **▣ 데이터 살펴보기**  -------------------------------------------------------------------------------------------  **∇ 데이터프레임 정보 확인하기**  \* df.shape -> 데이터프레임의 차원(행과 열의 개수)을 'tuple형태'로 보여줌.  \* df.info() -> 데이터에 대한 정보들(컬럼의 개수, 각 컬럼의 row 개수, 데이터 타입)을 'DataFrame 형태'로 보여줌.  \* df.describe() -> 데이터의 통계정보(최대, 최소, 중간, 평균, 사분위수, 각 column이 가진 데이터 수)를 'DataFrame 형태'로 보여줌  \* df.count() -> 각 column 이 가진 데이터 개수를 'Series 형태'로 보여줌.  -------------------------------------------------------------------------------------------  **∇ 통계함수 적용**  \* df.mean() -> 모든 열의 평균값  \* df['열이름'].mean() -> 특정열의 평균값  \* mean, max, min, std(표준편차), median, corr(두 변수사이에 상관계수 여부 파악) 모두 동일  -------------------------------------------------------------------------------------------  **∇ pandas 내장그래프 이용**  \* df.plot() 형태로 이용  \* 원하는 그래프 동류에 따라 df.plot(kind = '그래프형태')로 이용  -> line: 선그래프, bar: 수직막대그래프, barh: 수평막대그래프, hist: 히스토그램, box: 박스플랏  kide: 커널 밀도 그래프, area: 면적 그래프, pie: 파이 그래프, scatter: 산점도, hexbin: 고밀도 산점도  -------------------------------------------------------------------------------------------  **∇ 데이터 전처리**  \* 원하는 데이터만 추출하기, 데이터 타입 변환하기, 원하는 항목 추가하기 등 데이터 가공  ------------------------------------------------------------------------------------------- |
| 후 | **▣ 시각화 도구 (matplotlib)**  -------------------------------------------------------------------------------------------  **∇ 폰트 오류 해결하기**  import matplotlib.pyplot as plt  from matplotlib import font\_manager, rc  plt.rcParams['font.family'] = 'AppleGothic'  plt.rcParams['font.family'] = 'Malgun Gothic'  plt.rcParams['axes.unicode\_minus'] = False  -------------------------------------------------------------------------------------------  **∇ 데이터 전처리 (Null데이터 대체하기, 데이터 일치여부 확인)**  \* df.fillna(value = '원하는 데이터') Null을 입력 데이터로 교체  \* df.fillna(method = 'ffill') 앞의 값이 Null을 대체  \* df.fillna(method = 'bfill') 뒤의 값이 Null을 대체  \* df['전출지별'] == '서울특별시'  -------------------------------------------------------------------------------------------  **∇ 복습**  \* df = df.drop(['전출지별'], axis=1)  \* df.rename({'전입지별':'전입지'}, axis = 1, inplace = True)  \* df.set\_index('전입지', inplace = True)  -------------------------------------------------------------------------------------------  **∇**  \*  \*  \*  \*  \*  \*  ------------------------------------------------------------------------------------------- |