|  |  |
| --- | --- |
| 교육 제목 | **판다스 문법** |
| 교육 일시 | 2021-09-29 |
| 교육 장소 | 집 |
| **교육 내용** | |
|  | 시리즈 : 여러 개의 원소를 선택(인덱스 리스트 활용)  리스트형태([]) 안에 입력하면 짝을 이루슨 원소 데이터 모두 반환  정수형 위치 인덱스는 0부터 시작  딕셔너리 -> 시리즈 변환 pandas.Series( 딕셔너리 )  인덱스 구조 : 인덱스는 자기 짝을 이루는 원소 순서와 주소 저장  행 추가 : append로 행추가 loc[], iloc[] 를 사용하여 행 추가 가능  열 추가 : []연산자 메소드, df.insert()방법, df.assign()방법, df.loc()방법  인덱스 활용  특정 열 -> 행 인덱스로 설정  : DataFrame 객체.set\_index( [‘열 이름’ ] 또는 ‘열 이름’)    판다스 실습해보기    간단한 예제    서준의 점수를 바꾸기  우현과 인아 점수 출력하기  인아 점수 변경하기  서준 점수 출력하기  수학, 음악 점수만 출력하기 실습    인덱스 활용 -> set\_index() 메소드를 사용하여 데이터프레임의 특정 열을 행 인덱스로 설정, 새로운 객체를 반환 -> reindex() 데이터프레임의 행 인덱스를 새로운 배열로 재지정, 기존 객체 변경x 새로운 데이터프레임 객체를 반환  인덱스 재배열  reindex() 메소드 사용하면  행 인덱스 초기화  행 인덱스 기준으로 데이터프레임 정렬    Reindex() 함수로 인덱스 초기화후 기존 인덱스 열로 이동  행 인덱스 정렬  sort : 인덱스로 정렬, 특정 칼럼의 값으로 정렬    산술연산  내부적 3단계 프로세스를 거침  1. 행.열 인덱스를 기준으로 모든 원소를 정렬한다  2. 동일한 위치에 있는 원소끼리 일대일로 대응시킨다  3. 일대일 대응되는 원소끼리 연산을 처리 이때 대응값 없으면 NaN 출력      **산술연산 실습 코드**  **데이터 프레임 연산**  **: 행/열 인덱스를 기준으로 정렬, 일대일 대응되는 원소끼리 연산**    **데이터 프레임연산 실습1**      100 ~ 200 번째 값 출력  **데이터 프레임연산 실습2**  **데이터 입출력**  **입출력 도구 : 다양한 형태 외부 파일 읽어와 데이터프레임 변환 함수 제공**  **file Format, Reader, Writer**    **csv 파일: 데이터값 쉼표 , 로 구분 하고 있다는 의미**  **CSV(commaseparated values)라고 부르는 텍스트**  **read\_csv()함수에 확장자 .csv 를 포함하여 파일경로 입력하면**  **CSV 파일 읽어와 데이터프레임으로 변환**  **Csv 파일 읽어 오기**  **: file\_path = “파일위치/파일이름.csv”**    **해당 위치 파일 읽어 오기 실습**  **read\_csv(옵션,...)**  **- path = 파일의 위치 포함한 파일명**  **- sep : 필드를 구분하는 구분자 ','**  **- header : 헤더가 정의되어 있는지, None -> 헤더 정의 x**  **- index\_col : 인덱스로 사용될 컬럼명, None 인덱스 없음**  **- names : 컬럼 이름으로 사용될 문자열 리스트**  **- skiprows : 처음 행 부터 skip 하고자 하는 행수**  **- skipfooter : 마지막 행 부터 skip 하고자 하는 행수**  **- encoding : 텍스트 인코딩 종류를 지정 'utf-8'**  **Json 파일 읽어 오기**  **: pd.read\_json(파일위치/파일이름.json)**    **Url 파일 읽어 오기**  **: url = “파일위치/파일이름.html”**    **외부 파일 가져오기**  **: 해당 모듈 설치 후 해당 모듈 사용법 따를 것**  **외부 파일 예제**    **Googlemaps 모듈 사용하여 위도 경도 출력하기**    **파일 처리하는 방법 총 정리**  **읽어오기 : -read\_csv, read\_html, read\_excel, read\_json**  **저장하기**  **: df.csv(파일명), df.to\_html, df.to\_excel, df.to\_json**    **json파일은 구조가 Datafame과 동일**  **csv 파일은 2차원 배열로 -> Datafame으로 변경해서 사용**    **df = pandas.DataFrame ..**  **행 df.loc[인덱스명,] df.iloc[인덱스첨자] 여러개 [start : end, [, ..,]**  **열 df.열명 df[열명]**    **원소 df.loc[[인덱스명],[컬럼명]] df.iloc[[인덱스첨자],[컬럼의 순서 정수]]**  **오늘의 내용 정리 문제**  **#seaborn 에서 dataset "iris"를 불러와서**  **1. 'species'컬럼을 인덱스로 설정**  **2. 나머지 데이터의 합과 평균을 데이터 프레임에 추가**  **3. 변경된 데이터를 파일에 csv 형식으로 저장**  **4. 저장된 파일을 프로그램으로 불러 옴**  **5. 불러온 데이터를 출력해서 확인** |