|  |  |
| --- | --- |
| 교육 제목 | **Pandas 입문, 데이터 입출력** |
| 교육 일시 | 2021.09.29 |
| 교육 장소 | YGL-C6 |
| **교육 내용** | |
| 오전 | 1. 원소 값 변경   데이터프레임의 특정 원소를 선택하고 새로운 데이터 값을 지정해주면 원소 값이 변경 된다.    여기서 Seojoon의 수학 점수를 변경 시키는 방법    loc 메소드를 이용해서 seojoon의 데이터에 접근하고[‘math’] 컬럼에 접근해 데이터를 변경 시켜준다.  여러 개의 원소를 선택하여 새로운 값을 할당하면, 모든 원소를 한꺼번에 같은 값으로 변경할 수 있다 또한 선택한 원소의 개수에 맞춰 각기 다른 값을 배열 형태로 입력할 수도 있다.     1. 행, 열 위치 바꾸기   객체에 transpose() 또는 T 메소드를 사용해 행과열을 바꿀 수 있다.    이름이 컬럼으로 과목이 인덱스로 간 것을 확인 할 수 있다.   1. 행 인덱스 재배열   reindex() 메소드를 사용하면 데이터 프레임의 행 인덱스를 새로운 배열로 재지정할 수 있다. 기존 객체를 변경하지 않고 새로운 데이터프레임 객체를 반환한다.      인덱스 초기화 : reset\_index()메소드 사용  행 인덱스를 기준으로 데이터프레임 정렬 : sort\_index() 메소드 사용   * 오름차순 정렬 : sort\_index(ascending=True) * 내림차순 정렬 : sort\_index(ascendig= False)   모두 새로운 객체를 반환해준다.   1. 시리즈 연산   시리즈 객체에 어떤 숫자를 더하면 시리즈의 개별 dnjstdp rkrrkr tnt자를 더하고 계산한 결과를 시리즈 객체로 반환한다.     1. 데이터 프레임 연산   데이터프레임은 여러 시리즈가 한데 모인것이므로 시리즈의 확장 개념 |
| 오후 | 데이터입출력  외부파일 읽어오기   * 판다스는 다양한 형태의 외부 파일을 읽어와서 데이터 프레임으로 변환하는 함수를 제공한다.  1. CSV 파일   데이터 값을 쉼표로 구분하고 있다는 의미로 csv(comma-separated values) 라고 불린다.  pd.read\_csv(path) 의 메소드를 사용한다.  - path = 파일의 위치 포함한 파일명  - sep : 필드를 구분하는 구분자 ','  - header : 헤더가 정의 되어 있는지 None  - index\_col : 인덱스로 사용될 컬럼명, None 인덱스 없습니다.  - names : 컬럼 이름으로 사용 될 문자열 리스트  - skiprows : 처음 행 부터 skip 하고자 하는 행 수  - skipfooter : 마지막 행 부터 skip 하고자 하는 행 수  - encoding : 텍스트 인코딩 종류를 지정 'utf-8'     1. excel파일   엑셀의 행과 열은 데이터프레임의 행ㅏ 열로 일대일 대응 된다.  pd.read\_excel()메소드를 사용한다.    xlsx확장자의경우 engine 옵션에 openyxl  xls 확장자의 경우 xlrd를 옵션으로 사용.   1. JSON파일   json파일은 데이터 공유를 목적으로 개발된 특수한 파일 형식이다. 파이썬 딕셔너리와 비슷하게 key: value구조를 갖는데 구조가 중첩되는 방식에 따라 다르게 적용된다.    -json example  pd.read\_json()메소드를 사용한다.     1. HTML 파일   웹페이지에 있는 table태그에서 표 형식의 데이터를 모두 찾아서 데이터프레임으로 변환한다.  pd.read\_html(웹주소’ 또는 html파일 경로)    -sample.html을 chrome으로 열어서 html구문 확인    ‘   1. 웹 스크래핑   BeautifulSoup 등 웹 스크래핑 도구로 수집한 데이터를 판다스 데이터프레임으로 정리     1. API를 활용하여 데이터 수집하기   인터넷 서비스 업체에서 제공하는 API를 통해서 수집한 데이터를 판다스 데이터프레임으로 변환한다. 구글 지오코딩 API를 이용하여 구현     1. 데이터 저장    1. CSV파일로 저장 : 객체.to\_csv(‘파일이름’)    2. JSON파일로 저장 : 객체.to\_json(‘파일이름 ‘)    3. EXCEL파일로 저장 : 객체.to\_excel(‘파일이름’)    4. 여러 개의 데이터프레임을 하나의 엑셀 파일로 저장   writter = pd.ExcelWriter() # 엑셀 writter 객체 생성  df1.to\_excel(writter,sheet\_name = ‘sheet1’)  df2.to\_excel(writter,sheet\_name = ‘sheet2’)  메소드 사용 |