|  |  |
| --- | --- |
| 교육 제목 | **파이썬 클래스 사용법, Numpy\_1** |
| 교육 일시 | 2021년 11월 05일 |
| 교육 장소 | 비대면 수업 |
| **교육 내용** | |
| 내용 | 2021.11.05 9일차     1. 클래스    1. \_\_init\_\_ : 클래스 생성시 자동 실행    2. \_\_del\_\_ : 프로그램 종료시 자동 실행 -> 클래스 소멸 2. 클래스의 추가적인 구문    1. 어떤 클래스의 인스턴스인지 확인하기       1. isinstance(인스턴스, 클래스) : 상속 관계에 따라서 객체가 어떤 클래스로부터 만들어졌는지 확인. 인스턴스와 클래스가 동일하면 True 반환함.       3. 상속 : 어떤 클래스를 기반으로 그 속성과 기능을 물려받아 새로운 클래스를 만드는 것      1. 특수한 이름의 메서드 : \_\_???\_\_ 함수들 2. 클래스변수와 메서드    1. 클래스변수 : 클래스 구문 바로 아래의 단계에 변수를 선언 3. 가비지 컬렉터    1. 더 사용할 가능성이 없는 데이터를 메모리에서 제거하는 역할 4. 프라이빗 변수와 게터/세터      1. . 2. 데이터수집 파이프라인     2. 파이썬에서 다룰 수 있는 형태       1. 리스트, 튜플, 셋, 배열, 행렬, 프레임, 시리즈, 딕셔너리 3. 데이터의 구현 절차    1. 데이터의 적재       1. from sklearn.database import load iris       2. iris dataset = load iris()    2. 성과 측정 : 훈련데이터와 테스트 데이터 분리       1. from sklearn.model\_seletion import train\_test\_split       2. X\_train,X\_test,y\_train,y\_test = train\_test\_split(       3. iris\_dataset[‘data’], iris\_dataset[‘target’], random\_state=0)    3. 데이터 살펴보기       1. iris\_dataframe = pd.dataFrame(X\_train,       2. columns=iris\_dataset.feature\_names).       3. pd.plotting.scatter\_matrix(iris\_dataframe, c=y, train, figsize=(15,15),       4. marker=’o’,his\_kwds={‘bins’:20}, s=60,alpha=.8,       5. cmap=magal    4. 머신러닝 모델 선택       1. from sklearn.neighbors import KNeighborsClassfier       2. knn = KneighborsClassfier(n\_neighbors=1) # 모델 선택       3. knn.fit(X\_train,y\_train)    5. 예측하기       1. prediction = knn.predict(X\_new)    6. 모델 평가하기       1. knn.score(X\_test,Y\_test) 4. Numpy 패키지    1. 넘파이       1. 주로 다차원 배열이나 행렬과 수학 함수 지원       2. 벡터화 연산을 이용하여 간단한 코드로도 복잡한 선형대수연산을 수행       3. 배열 인덱싱을 사용한 질의 기능을 이용하여 간단한 코드로도 복합한 수식을 계산    2. 사용법       1. import numpy / import numpy as np    3. array 정의 및 사용하기    4. array의 연산      1. array의 인덱싱 2. 기술 통계 함수 3. 난수 발생과 카운팅     2. 실습 |