|  |  |
| --- | --- |
| 교육 제목 | **파이선을 활용한 빅데이터 분석** |
| 교육 일시 | 20211105 09:00 ~ 18:00 |
| 교육 장소 | 자택 온라인 수업 |
| **교육 내용** | |
| 오전 | # class definition, class create  # class class name  class Aclass:  def \_\_init\_\_(self.num1, num2):  selfnum1 = num1  selfnum2 = num2  print("class create")    def get\_sum(self):  print("class 내부 함수 실행")  return(self.num1 + self.num2)    a = Aclass(10, 20) # 생성자 => class 내부의 \_\_init\_\_ 함수가 실행  print(a.num1, a.num2)  b = Aclass(30, 40)  print(b.num1, b.num2, b\_get\_sum())  a\_list = [a, b]  a\_list[0].get\_sum  # 상속  class Parent:  def \_\_init\_\_(self):  self.value = "test"  print ("Parent 클레스의 \_\_init()\_\_메서드가 호출되었습니다.")  def test(self):  print ("Parent 클레스의 test() 메서드입니다.")    # 자식 클래스를 선언합니다.  class Child(Parent):  def \_\_init\_\_(self):  Parent.\_\_init\_\_(self)  print("Child 클레스의 \_\_init\_\_메서드가 호출되었습니다.")  def test(self):  print("Child 클레스의 test() 메서드입니다.")    # 자식 클래스의 인스턴스를 생성하고 부모의 메서드를 호출합니다.  child = Child()  child.test()  print(child.value) |
| 오후 | # 파이선을 이용한 빅데이터 분석에 대한 이해  # http://bit.ly/3GR4B16  # numpy : 배열의 연산등을 쉽게 처리하도록 도와주는 패키지  import numpy as np  np.arange(-10, 5, 0.5).reshape(5,6)  # list 에서 array를 생성하는 방법  a\_list = [1, 2, 3, 4, 5, 6]  a\_array = np.array(a\_list)  print(type(a\_list), type(a\_array), a\_list, a\_array)  a\_array.shape  # 1부터 100까지의 수중에서 3의 배수인 배열로 생성  # 위에서 생성된 배열의 값 중에서 임의로 5개의 값을 출력  # 위의 배열의 사이즈를 확인한 후 행이 5인 2차원 배열을 생성  # 배열의 원소 중에서 5의 배수인 출력 진행  arr = np.arange(1, 101)  arr = arr[arr%3 == 0]  print("arr : ", arr)  print("임의의 수 5개 출력 : ", np.random.choice(arr, t, replace = False))  arr\_1 = arr.reshape(3, int(arr.size / 3))  print("5의 배수만 출력 : ", arr\_1[arr\_1%5 == 0]) |