|  |  |
| --- | --- |
| 교육제목 | 데이터 기반 인공지능 시스템 엔지니어 양성 과정 |
| 교육일시 |  |
| 교육장소 | YGL 학과장 |
| **교육내용** | |
| 오전 | 1. Dictionary   리스트 : 인덱스를 기반으로 값을 저장, 변수 선언은 []  딕셔너리 : 키를 기반으로 값을 저장, 변수 선언은 {}  Ex)  -딕셔너리 선언  dictionary = {  "name" : "7D 건조 망고",  "type" : "당절임",  "ingredient" : ["망고", "설탕", "메타중아황산나트륨", "치자황색소"],  "origin" : "필리핀"  }  -출력  print("name:" , dictionary["name"])  print("type:" , dictionary["type"])  print("ingredient:", dictionary["ingredient"])  print("origin:", dictionary["origin"])  print()  -값 변경  dictionary["name"] = "8D 건조 망고"  print("name:", dictionary["name"])  -딕셔너리에 있는 값을 제거할땐 ‘del’  del dictionary["origin"]  특정 값이 어떤 자료형인지 확인하는 방법  type("문자열") is str : 문자열인지 확인 type([]) is list : 리스트인지 확인 type({}) is dict : 딕셔너리인지 확인 ex) type(dict\_a) if type(list\_a) == list: print("list\_a는 list") if type(dict\_a) == dict: print("dict\_a는 dict")  character = {  "name" : "기사",  "level" : 12,  "items": {  "sword": "불꽃의 검",  "armor": "풀플레이트"  },  "skill" : ["베기","세게 베기","아주 세게 베기"]  }  문)  for key in character:  (key-> "items", character["items"]->{"sword": "불꽃의 검", "armor" : "풀플레이트"}  ( key-> "skill" , character["skill"]->["베기","세게 베기","아주 세게 베기"]  if type(character[key]) == list:  for value in character[key]:  print(key, " : ", value)  elif type(character[key]) == dict: (딕셔너리는 값을 키값으로 가져오니까)  for key1 in character[key]:  print(key1, " : ", character[key][key1])  else:  print(key, " : ", character[key])   1. 범위함수 range()   for i in range(5): # 0부터 4(5-1)까지  print(str(i) + "= 반복 변수") # str(i) = i를 문자열로 변환.  print()  for i in range(5,10): # 5부터 9(10-1)까지  print(str(i) + "= 반복 변수")  print()  for i in range(0, 10, 3): # 0~9(10-1)까지 3의 간격으로  print(str(i) + "=반복 변수")  print()   1. While 조건식   Ex1)  i = 0  while i<10: # 조건식을 탈출하는 구문이 while 처리문 안에 존재해야 한다.  print(".",end="")  i+=1 # i=i+1  ex2)  array = [273, 32, 103, 57, 32, 52]  value = 32  ( del array[1] : 1 인덱스의 리스트값을 제거)  while value in array: # 값을 list에서 모두 지우기  array.remove(value) # list의 해당하는 값을 모두 제거  문1) 기호 숫자 두 개를 입력받아 계산하는 프로그램을 작성  기호가 '#'이면 프로그램 종료  기호는 '+','-','\*','/'만 실행, 다른 기호는 계속해서 진행.  char\_list= ['+','-','\*','/']  while True:  input\_char = input("> 부호 입력 ( #이면 종료, +,-,\*,/) >")  if input\_char == '#':  break  elif input\_char in char\_list:  while True:  input\_num = input(" 두 개의 숫자 입력 > ").split()  if len(input\_num) == 2:  break    value1 = int(input\_num[0])  value2 = int(input\_num[1])  if input\_char == '+':  print("{} {} {} = {}".format(value1, input\_char, value2,value1+value2))  elif input\_char == '-':  print("{} {} {} = {}".format(value1, input\_char, value2,value1-value2))  elif input\_char == '\*':  print("{} {} {} = {}".format(value1, input\_char, value2,value1\*value2))  else:  if value1 != 0 and value2 != 0:  print("{} {} {} = {}".format(value1, input\_char, value2, value1/value2))  else:  print("0은 나누기 안됨")  print("program end")  문2) 빈칸을 채워 키와 값으로 이루어진 각 리스트를 조합해 하나의 딕셔너리를 만들어보자.  key\_list = ["name", "hp", "mp", "lev"]  value\_list = ["기사", 200, 30, 5]  character = {}  # 원래는 character["name"]="기사", 일일히 입력하기엔 번거로움.  for i in range(len(key\_list)): # len(key\_list) <- 4, range(4) <- 0~3 index 사용  character[key\_list[i]] = value\_list[i]  print(character) |
| 오후 | 1. 파이선만의 고유 기능   리스트에 적용 가능한 기본 함수 : min(), max(), sum()  리스트 뒤집기 : reversed()  현재 인덱스가 몇번째인지 확인 : enumerate()  딕셔너리로 쉽게 반복문 작성 : items()  리스트 안에 for문 사용하기 : 리스트 내포  numbers = [ 102, 53, 273, 32, 77, 55]  sum(numbers) # 함수에서 Shift+Tab누르면 정보 나옴.  sum(numbers,start=2)  min(numbers)  max(numbers)  print(list(reversed(numbers)))  print(numbers[ : :-1]) # reversed()와 같음 : [시작:끝:맨뒤부터]  print(sum(numbers,start=2), sum(numbers,start=0))   1. enumerate() 함수   numbers = [ 102, 53, 273, 32, 77, 55]  list(enumerate(numbers))  print(numbers)  for i in range(len(numbers)):  print("{}번째 요소 : {}".format(i, numbers[i]))  print("\nenumerate")  for i, value in enumerate(numbers): # enumerame : list의 인덱스와 값을 리턴  print("{}번째 요소 : {}".format(i,value))  print()  문) 키보드로부터 임의의 갯수의 숫자를 입력받아 리스트에 저장 한 후  print("{}번째 요소 : {}".format(i,value)) 형식으로 결과 출력  전체 숫자의 최솟값과 최댓값, 합계를 구하시오  1. 문자열을 space로 분리하여 리스트화, 문자열을 숫자로 변환.  2. 출력 형식대로, 합계, 최솟값, 최댓값 구함.  input\_num = list(input("랜덤으로 숫자 입력 :").split())  for i, value in enumerate(input\_num):  input\_num[i] = int(value)  print("{}번째 요소 : {}".format(i, value))    print("최소값 : {}\t 최댓값 : {} \t 합계 : {}\n".format(min(input\_num),max(input\_num),sum(input\_num)))   1. 리스트 내포   : list 안에 for 문장 사용. => [표현식 for 변수 in 반복 if 조건식]  array = []  for i in range(0, 20, 2):  array.append( i \* i)  print(array)  # [실행문 for 변수(i,j...) in 반복]  print()  array1 = [i\*i for i in range(0,20,2)] # 반복자료에서 아이템 1개를 가져와서 i에 입력 후 i\*i 실햄.  print("array1 : ", array1)  print()  # 조건식을 조합  array2 = [i\*i for i in range(0,20) if i%2 == 0] # 반복 범위 뒤에 if 조건식 추가 가능.  print("array2 : ", array2)  # 여러개의 숫자를 입력받아 숫자가 6의 배수면 list\_6에, 3의 배수면 list\_3에, 2의 배수면  #list\_2에. 단, 각 리스트에 공통되는 숫자는 없습니다.  list\_1 = input("숫자 입력 : ").split()  # 문자를 숫자로 변환  for i, value in enumerate(list\_1):  list\_1[i] = int(value)  ★★★★★★★★★★조건에 맞는 리스트 작성(중요)★★★★★★★★★★★★  list\_2 = [ i for i in list\_1 if i%6 and i%2 == 0] # i%6의 결과값(나머지)이 0일경우 false출력됨. 그 외 True. -> 6의 배수가 아니고  # 3의 배수이면 True and True.  # list\_1에서 i 넣고 if조건에 맞을경우 i에 다시 넣음.  list\_3 = [ i for i in list\_1 if i%6 and i%3 == 0]  list\_6 = [ i for i in list\_1 if i%6 == 0]  #★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★★  #결과 출력  print("-list\_2:{},\n-list\_3:{},\n-list\_6:{}\n".format(list\_2,list\_3,list\_6))   1. 함수   : def 함수명():  실행문  def print\_func(value, \*values, n=3): # value : 매개변수, values : 가변 매개변수(단 하나), n : 기본 매개변수(여러개 사용 가능)  for i in range(n):  for item in values:  print(value," : ", item)  return "break"  return "end"  문) 기호 숫자 두 개를 입력받아 계산하는 프로그램을 작성  기호가 '#'이면 프로그램 종료  기호는 '+','-','\*','/'만 실행, 다른 기호는 계속해서 진행.  각각의 연산을 함수로 만들어서 호출. plus\_func, minus\_func, mul\_func, div\_func  def input\_func():  while True:  input\_char = input("부호 입력 기호가 '#'이면 프로그램 종료 :")  if input\_char in buho\_list:  return input\_char  def input\_number():  while True:  input\_number = input("두 수 입력 : ").split()  if len(input\_number) == 2 :  return input\_number    def plus\_func(values):  return( values[0] + values[1])  def minus\_func(values):  return( values[0] - values[1])  def mul\_func(values):  return( values[0] \* values[1])  def div\_func(values):  return( values[0] / values[1])  buho\_list=["+","-","\*","/","#"]  while True:  buho\_char = input\_func() # 부호입력 # + - \* /  if buho\_char == "#":  break  else:  values = input\_number() # = 이 나오면 오른쪽부터 해석..?  for i, value in enumerate(values): #문자를 숫자로 형변환  values[i] = int(value)  if buho\_char == "+":  result = plus\_func(values)  elif buho\_char == "-":  result = minus\_func(values)  elif buho\_char == "\*":  result = mul\_func(values)  else:  if values[1] == 0:  print("0으로 나눌 수 없습니다.")  continue  result = div\_func(values)  print("{} {} {} = {}".format(values[0],buho\_char,values[1],result)) |