



지능로봇공학과

INTELLIGENT ROBOT

담당 교수	임종관 교수님
학과	지능로봇학과
학번	1558021
이름	이영섭

목 차

code07-01	3P
code07-02	4P
code07-03	5P
code07-04	6P
code07-05	7P
code07-06	8P
code07-07	9P~10P
code07-08	11P
code07-09	12P
code07-10	13P
응용예제1	14P
응용예제2	15P~16P
참고문헌	17P

※CODE07-01

```
a, b, c, d = 0, 0, 0, 0 #변수를 선언한다
hap = 0 #변수를 선언한다
a = int(input("1번째 숫자 : ")) #정수를 입력받는다
b = int(input("2번째 숫자 : ")) #정수를 입력받는다
c = int(input("3번째 숫자 : ")) #정수를 입력받는다
d = int(input("4번째 숫자 : ")) #정수를 입력받는다
hap = a + b + c + d #hap에 a,b,c,d를 다 더한값을 저장한다
print(" 합계 ==> %d" % hap) #출력
```

```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\#영섭\Desktop\#2학년 1학기\#고급프로그래밍\#출석과제6\#Code0
7-01.py =====
1번째 숫자 : 1
2번째 숫자 : 2
3번째 숫자 : 3
4번째 숫자 : 4
합계 ==> 10
>>> |
```

※CODE07-02

```
aa = [0, 0, 0, 0] #list변수를 선언한다
hap = 0 #변수를 선언한다
aa[0] = int(input("1번째 숫자 : ")) #list변수에 정수를 입력받는다
aa[1] = int(input("2번째 숫자 : ")) #list변수에 정수를 입력받는다
aa[2] = int(input("3번째 숫자 : ")) #list변수에 정수를 입력받는다
aa[3] = int(input("4번째 숫자 : ")) #list변수에 정수를 입력받는다
hap = aa[0] + aa[1] + aa[2] + aa[3] #hap에 aa의 리스트에 있는 숫자를 모두 더한 값을 저장한다
print("합계 ==> %d" % hap) #출력
```

```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\#영섭\Desktop\#2학년 1학기\#고급프로그래밍\#출석과제6\#Code07-02.py =====
1번째 숫자 : 1
2번째 숫자 : 2
3번째 숫자 : 3
4번째 숫자 : 4
합계 ==> 10
>>> |
```

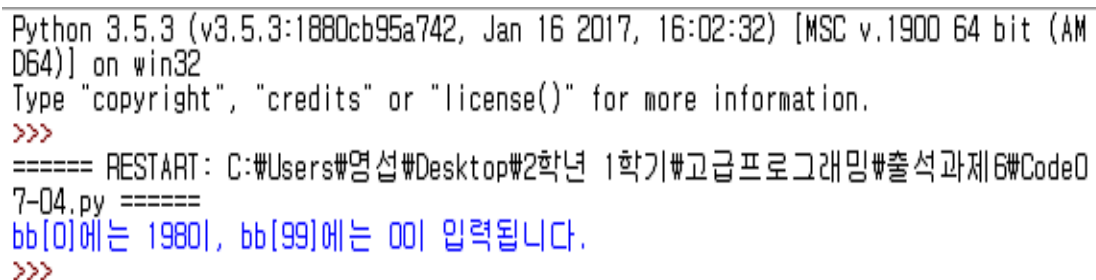
※CODE07-03

```
aa = [] #list변수를 선언한다
for i in range(0, 4) : #i가0부터3까지 1씩증가하며 반복문을 실행 한다
    aa.append(0) #aa의 list에 값을 추가한다
hap = 0 #변수를 선언한다
for i in range(0, 4) : #i가0부터3까지 1씩증가하며 반복문을 실행 한다
    aa[i] = int(input(str(i+1) + "번째 숫자 : " )) #aa[0]부터 aa[3]까지 정수를 입력받는다
hap = aa[0] + aa[1] + aa[2] + aa[3] #aa의 list에 있는수를 모두더한다
print("합계 ==> %d" % hap) #출력
```

```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\명섭\Desktop\2학년 1학기\고급프로그래밍\출석과제6\Code07-03.py =====
1번째 숫자 : 1
2번째 숫자 : 2
3번째 숫자 : 3
4번째 숫자 : 4
합계 ==> 10
>>> |
```

※CODE07-04

```
aa = [] #list변수를 선언한다
bb = [] #list변수를 선언한다
value = 0 #변수를 선언한다
for i in range(0, 100) : #i가 0부터99까지 1씩증가하며 반복문을 실행한다
    aa.append(value) #aa의 list에 value값을 저장한다
    value += 2 #value의값이 2증가한다
for i in range(0, 100) : #i가 0부터99까지 1씩증가하며 반복문을 실행한다
    bb.append(aa[99 - i]) #bb의 list에 aa값을 거꾸로 저장한다
print("bb[0]에는 %d이, bb[99]에는 %d이 입력됩니다." % (bb[0], bb[99]))#출력
```



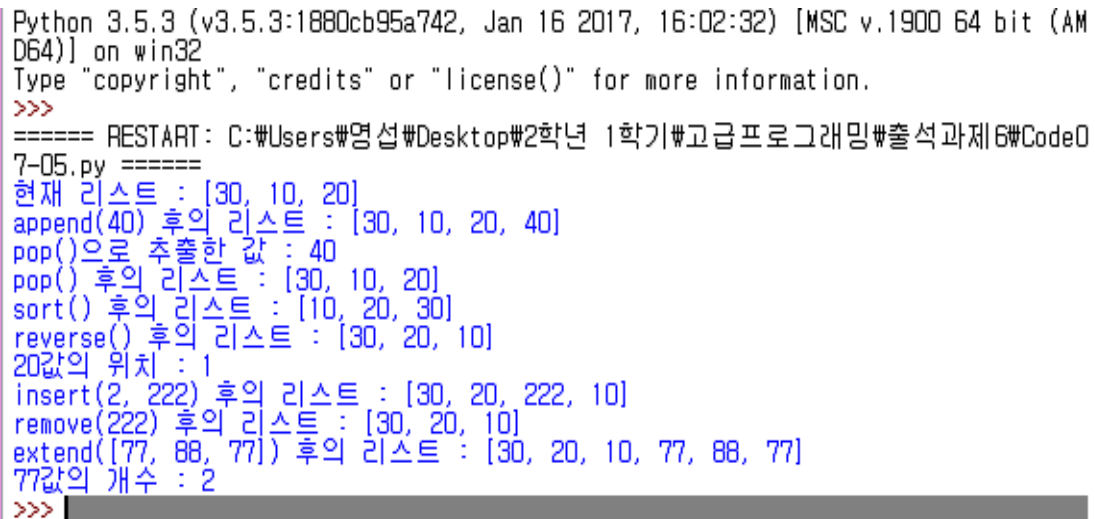
```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\명섭\Desktop\2학년 1학기\고급프로그래밍\출석과제6\Code07-04.py =====
bb[0]에는 1980이, bb[99]에는 0이 입력됩니다.
>>>
```

※CODE07-05

```

myList = [30, 10, 20] #변수선언(값저장)
print("현재 리스트 : %s" % myList) #list출력
myList.append(40) #myList에 40을 저장
print("append(40) 후의 리스트 : %s" % myList) #mylist 출력
print("pop()으로 추출한 값 : %s" % myList.pop()) #mylist의 맨뒤의 값출력
print("pop() 후의 리스트 : %s" % myList) #mylist 출력
myList.sort() #list정렬
print("sort() 후의 리스트 : %s" % myList) #정렬한 mylist출력
myList.reverse() #list를 역순으로 정렬
print("reverse() 후의 리스트 : %s" % myList) #역순으로 정렬한mylist를 출력
print("20값의 위치 : %d" % myList.index(20)) #mylist안의 20값의 위치를 출력한다
myList.insert(2, 222) #mylist의 지정된 위치에 222를 집어넣는다
print("insert(2, 222) 후의 리스트 : %s" % myList) #mylist출력
myList.remove(222) #mylist에서 222를 지운다
print("remove(222) 후의 리스트 : %s" % myList) #mylist출력
myList.extend([77, 88, 77]) #mylist뒤에 [77, 88, 77] 추가한다
print("extend([77, 88, 77]) 후의 리스트 : %s" % myList) #mylist출력
print("77값의 개수 : %d" % myList.count(77)) #mylist의 77의 개수를 파악하여 출력

```



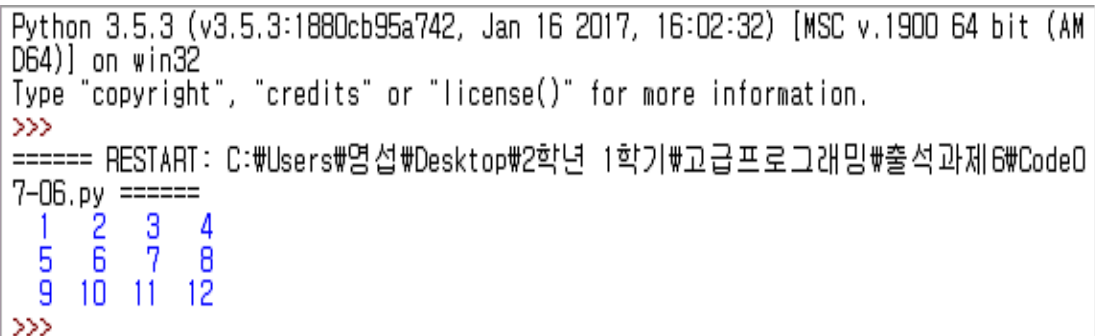
```

Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\#명섭\Desktop\#2학년 1학기\#고급프로그래밍\출석과제6\Code07-05.py =====
현재 리스트 : [30, 10, 20]
append(40) 후의 리스트 : [30, 10, 20, 40]
pop()으로 추출한 값 : 40
pop() 후의 리스트 : [30, 10, 20]
sort() 후의 리스트 : [10, 20, 30]
reverse() 후의 리스트 : [30, 20, 10]
20값의 위치 : 1
insert(2, 222) 후의 리스트 : [30, 20, 222, 10]
remove(222) 후의 리스트 : [30, 20, 10]
extend([77, 88, 77]) 후의 리스트 : [30, 20, 10, 77, 88, 77]
77값의 개수 : 2
>>>

```

※CODE07-06

```
list1 = [] #list변수 선언
list2 = [] #list변수 선언
value=1 #변수선언
for i in range(0, 3) : #i가 0부터2까지 증가하면서 반복문 실행
    for k in range(0, 4) : #k가 0부터3까지 증가하면서 반복문 실행
        list1.append(value) #list1에 value를 추가함
        value += 1 #value의값을 1증가시킴
    list2.append(list1) #list2에 list1을 추가함
    list1 = [] #list1 초기화
for i in range(0, 3) : #i가 0부터2까지 증가하면서 반복문 실행
    for k in range(0, 4) : #k가 0부터3까지 증가하면서 반복문 실행
        print("%3d" % list2[i][k], end = " ") #출력 [행][열]
    print("") #줄바꿈
```



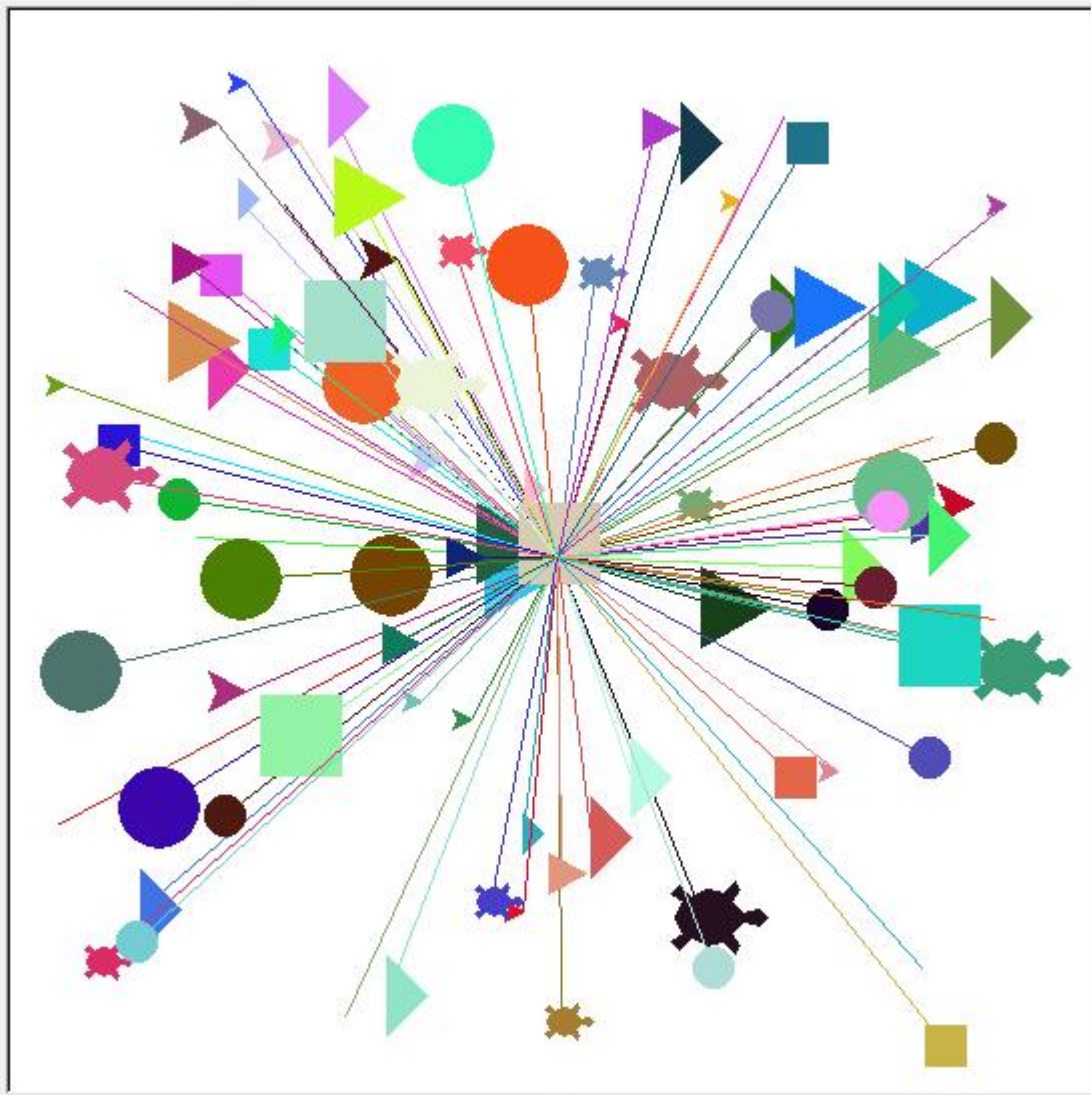
```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\#명섭\Desktop#2학년 1학기#고급프로그래밍#출석과제6#Code07-06.py =====
 1  2  3  4
 5  6  7  8
 9 10 11 12
>>>
```


※CODE07-07

```

import turtle #turtle모듈을 임포트한다
import random #random모듈을 임포트한다
myTurtle , tX, tY, tColor, tSize, tShape= [None] * 6 #변수를 선언한다
shapeList = [] #list변수를선언한다
playerTurtles = [] #list변수를선언한다
swidth, sheight=500, 500 #변수선언(값지정)
if __name__ == "__main__" : #maun함수(항상실행)
    turtle.title('거북 리스트 활용') #윈도창제목설정
    turtle.setup(width = swidth + 50, height = sheight + 50)#윈도창 크기설정
    turtle.screensize(swidth, sheight) #내부창 크기설정
    shapeList = turtle.getshapes() #shapeList에 turtle의 모든 모양을 가져온다
    for i in range(0, 100) : #가0부터 99까지반복문 실행
        random.shuffle(shapeList) #랜덤으로 shapeList를 섞는다
        myTurtle = turtle.Turtle(shapeList[0]) #첫번째있는 거북이모양을 myturtle
에 입력한다
        tX = random.randrange(-swidth / 2, swidth / 2) #-swidth / 2와swidth
/ 2 사이의 값을 랜덤으로 tX에 저장한다
        tY = random.randrange(-sheight / 2, sheight / 2) #-sheight / 2와
sheight / 2사이의 값을 랜덤으로 tY에 저장한다
        r = random.random(); g = random.random(); b = random.random()
#r,g,b의 값을 랜덤으로 저장한다
        tSize = random.randrange(1, 3) #tSize의 값을 1부터 3사이의 값에서 랜
덤으로 저장한다
        playerTurtles.append([myTurtle, tX, tY, tSize, r, g, b])#playerTurtles의
list에 [myTurtle, tX, tY, tSize, r, g, b]를 저장한다
        for tList in playerTurtles : #tList에 playerTurtles의값이 하나씩돌아가면서 반복문
을 실행한다
            myTurtle = tList[0] #거북이모양을불러온다
            myTurtle.color((tList[4], tList[5], tList[6])) #거북이의 색깔을 지정한다
            myTurtle.pencolor((tList[4], tList[5], tList[6])) #거북이 pen의 색을 지정
한다
            myTurtle.turtlesize(tList[3]) #turtle의 크기를 지정한다
            myTurtle.goto(tList[1], tList[2]) #turtle의 이동좌표를 설정한다
    turtle.done() #멈춤

```



※CODE07-08

```
singer = {} #딕셔너리변수 선언
singer['이름'] = '트와이스' #딕셔너리 추가
singer['구성원수'] = 9 #딕셔너리 추가
singer['데뷔'] = '서바이벌 식스틴' #딕셔너리 추가
singer['대표곡'] = 'SIGNAL' #딕셔너리 추가
for k in singer.keys() : #k에 singer의 키를 대입(반복문)
    print('%s --> %s' % (k, singer[k]))#출력
```

```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\명섭\Desktop\2학년 1학기\고급프로그래밍\출석과제6\Code07-08.py =====
데뷔 --> 서바이벌 식스틴
대표곡 --> SIGNAL
구성원수 --> 9
이름 --> 트와이스
>>> |
```

※CODE07-09

```
import operator #operator모듈을 임포트한다
trainDic, trainList = {}, [] #딕셔너리변수와 리스트변수를 선언한다
trainDic = {'Thomas':'토마스', 'Edward':'에드워드', 'Henry':'헨리', 'Gothen':'고든',
'James':'제임스'} #딕셔너리변수에 값을 저장한다
trainList = sorted(trainDic.items(), key=operator.itemgetter(0)) #trainList에 trainDic를 정
렬하여 저장한다
print(trainList) #출력
```

```
Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\명섭\Desktop\2학년 1학기\고급프로그래밍\출석과제6\Code0
7-09.py =====
[('Edward', '에드워드'), ('Gothen', '고든'), ('Henry', '헨리'), ('James', '제임
스'), ('Thomas', '토마스')]
>>>
```

※CODE 07-10

```
foods = {"떡볶이":"오뎅",
        "짜장면":"단무지",
        "라면":"김치",
        "피자":"피클",
        "맥주":"땅콩",
        "치킨":"치킨무",
        "삼겹살":"상추" }; #딕셔너리 변수 선언

while (True) : #무한루프
    myfood = input(str(list(foods.keys())) + " 중 좋아하는 음식은?") #myfood에 문자열
    을 입력받는다
    if myfood in foods : #if조건문 (문자열 입력받은 것이 foods에 있을 경우)
        print("<%s> 공합 음식은 <%s>입니다." % (myfood, foods.get(myfood))) #출력
    elif myfood == "끝" : #elif조건문(끝을입력했을 경우 종료)
        break #반복문 탈출
    else : #else if와elif가 충족하지 않을때
        print("그런 음식이 없습니다. 확인해 보세요.") #출력
```

```
type copyright , credits or license() for more information.
>>>
===== RESTART: C:\Users\명섭\Desktop\2학년 1학기\고급프로그래밍\출석과제6\Code0
7-10.py =====
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 피자
<피자> 공합 음식은 <피클>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 짜장면
<짜장면> 공합 음식은 <단무지>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 치킨
<치킨> 공합 음식은 <치킨무>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 삼겹살
<삼겹살> 공합 음식은 <상추>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 라면
<라면> 공합 음식은 <김치>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 맥주
<맥주> 공합 음식은 <땅콩>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 떡볶이
<떡볶이> 공합 음식은 <오뎅>입니다.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 피자
그런 음식이 없습니다. 확인해 보세요.
['피자', '짜장면', '치킨', '삼겹살', '라면', '맥주', '떡볶이'] 중 좋아하는 음식
은? 끝
>>>
```

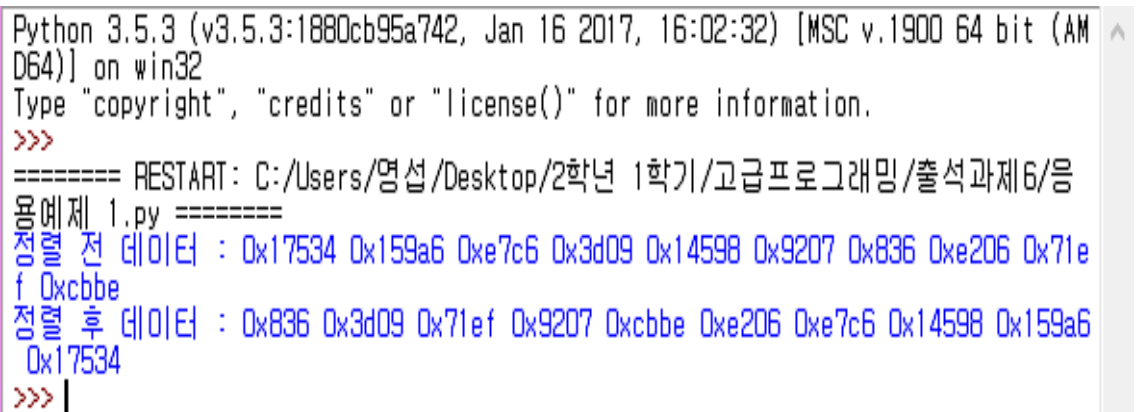
※응용예제 1

```

import random #random모듈을 임포트한다
data = [] #date리스트변수를 선언한다
i, k = 0, 0 #변수를 선언한다
if __name__ == "__main__" : #main함수(항상실행)
    for i in range (0, 10) : #i가 0부터9까지 1씩 증가하며 반복
        tmp = hex( random.randrange(0, 100000)) # tmp에 0,100000까지수를 입력받
아 16진수로 저장
        data.append(tmp) #date에 저장한다
    print('정렬 전 데이터 : ', end = '') #출력
    [print(num, end = ' ') for num in data] #??나열출력??
    for i in range(0, len(data)-1) : # i가 0부터 date의 list수-2까지 1씩 증가하며 반복
        for k in range(i+1, len(data)) : # k가 i+1부터 date의 list수-1까지 1씩 증가하며
반복
            if int(data[i], 16) > int(data[k], 16) : #if 조건문
                tmp = data[i] #tmp에 data[i]를 저장한다
                data[i] = data[k] #data[i]에 data[k]를 저장한다
                data[k] = tmp #data[k]에 tmp를 저장한다

    print('\n정렬 후 데이터 : ', end = '') #출력
    [print(num, end = ' ') for num in data] #??나열출력??

```



```

Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/명섭/Desktop/2학년 1학기/고급프로그래밍/출석과제6/응용예제 1.py =====
정렬 전 데이터 : 0x17534 0x159a6 0xe7c6 0x3d09 0x14598 0x9207 0x836 0xe206 0x71ef 0xcbbe
정렬 후 데이터 : 0x836 0x3d09 0x71ef 0x9207 0xcbbe 0xe206 0xe7c6 0x14598 0x159a6 0x17534
>>> |

```

※응용예제2

```

import operator #operator모듈을 임포트한다
trainTupleList = [('토마스', 5), ('헨리', 8), ('에드워드', 9), ('에밀리', 5), ('토마스', 4), ('헨리', 7), ('토마스', 3), ('에밀리', 8), ('퍼시', 5), ('고든', 13)] #trainTupleList에 값을 지정해준다
trainDic, trainList = {}, [] #list변수와 딕셔너리변수를 선언한다
tmpTup = None #변수를 선언한다
totalRank, currentRank = 1, 1 #변수를 선언한다
if __name__ == "__main__" : #main함수(항상실행)
    for tmpTup in trainTupleList : #tmpTup에 trainTupleList가 하나씩 들어가며 반복문을 실행한다
        tName = tmpTup[0] #tName에tmpTup[0]값을 저장한다
        tWeight = tmpTup[1] #tWeight에tmpTup[1]값을 저장한다
        if tName in trainDic :#tName에 trainDic가 있을 경우 실행
            trainDic[tName] += tWeight #trainDic[tName]에 tWeight의 값을 누적
        else : #if조건문이 성립되지 않을때
            trainDic[tName] = tWeight #trainDic[tName]에 tweight의 값을 저장

    print('기차 수송량 목록 ==> ', trainTupleList) #출력
    print('-----') #출력
    trainList = sorted(trainDic.items(), key = operator.itemgetter(1), reverse = True) #기차딕셔너리값을 기준으로 정렬
    print('기차Wt총수송량Wt순위') #출력
    print('-----') #출력
    print(trainList[0][0], 'Wt', trainList[0][1], 'Wt', currentRank) #출력
    for i in range(1, len(trainList)) : #i가 1부터 trainList의 list의 개수-1까지 증가하며 반복문 실행
        totalRank += 1 #total에 +1누적
        if trainList[i][1] == trainList[i-1][1] : #if조건문 trainList[i][1]이 trainList[i-1][1] 와 같을때 실행
            pass #넘어감
        else : #if조건문이 성립하지 않을때
            currentRank = totalRank #currentRank에 totalRank저장
    print(trainList[i][0], 'Wt', trainList[i][1], 'Wt', currentRank) #출력

```

```

Python 3.5.3 (v3.5.3:1880cb95a742, Jan 16 2017, 16:02:32) [MSC v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/영섭/Desktop/2학년 1학기/고급프로그래밍/출석과제6/응
용예제2.py =====
기차 수송량 목록 ==> [('토마스', 5), ('헨리', 8), ('에드워드', 9), ('에밀리', 5),
('토마스', 4), ('헨리', 7), ('토마스', 3), ('에밀리', 8), ('퍼시', 5), ('고든', 13)]
=====
기차    총수송량    순위
=====
헨리      15      1
고든      13      2
에밀리    13      2
토마스    12      4
에드워드   9       5
퍼시      5       6
>>> |

```


우재남, 파이썬 of beginner, 한빛아카데미,
2017