

<div>2022 년 2 학기</div> <div>파이썬프로그래밍심화</div> <div>조영준 교수님</div>	제출일	2022-10-30
	학 과	인공지능학부
	학 번	214499
	성 명	나유경

[실습 1] 모든 리스트 원소의 합계 출력하기

[코드] # 주석 작성 필수

```
#1.
### 변수 선언
list1 = [] #리스트 선언
sum1 = 0 # 합계 변수 선언

### 입력(시작값, 끝값, 증가값)
for i in range(10,41,10): #10~40 번째 인덱스까지 10씩 증가
    list1.append(i) #리스트에 원소 추가

### for문을 이용해 처리
for i in list1:
    sum1 +=i #합계에 더함

### 출력
for i in range(len(list1)):
    print(f'{i+1}번째 숫자: {list1[i]}') #list 원소 출력

print('합계==>',sum1) #합계 출력
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
y
1번 째 숫자 : 10
2번 째 숫자 : 20
3번 째 숫자 : 30
4번 째 숫자 : 40
합 계 ==> 100
```

[실습 2] N의 배수 출력하기(2차원 리스트)

[코드] # 주석 작성 필수

```
#2.
#numpy 간절하게 쓰고싶습니다...
### 입력
n = int(input('N을 입력하세요: '))
ls = [[0,1,2,3,4],[5,6,7,8,9],[10,11,12,13,14],[15,16,17,18,19]] #4행 5열 2차원
배열 선언
result = [[n]*5]*4 #n으로 구성된 행렬 생성

### 처리 및 출력
for i in range(len(ls)): #행렬의 곱
    result[i] = [a*b for a,b in zip(ls[i], result[i])] #list comprehension
    사용

for i in result: #출력 코드
    for j in i:
        print f'{j:3d}', end='') #공백 생성
    print()
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
N을 입력하세요: 3
  0  3  6  9 12
15 18 21 24 27
30 33 36 39 42
45 48 51 54 57
```

[실습 3] 습도 구하기

[코드] # 주석 작성 필수

```
#3.
### list 생성
temp_ls = [0,10,20,30] #온도 리스트
full_h_ls = [4.8, 9.4, 17.3, 30.4] #포화수증기량 리스트(앞 온도리스트의 인덱스를 따르는)

### 입력
h = float(input('현재 수증기량 입력: '))
temp = int(input('현재 온도 입력: '))

### 처리 및 출력
a = temp_ls.index(temp) #사용자가 입력한 온도가 몇 번 인덱스인지 확인하기
real_h = (h / full_h_ls[a]) *100 #공식 사용
print(f'현재 습도는 {real_h}% 입니다.') #출력 문구
### 프로그램 종료문구 출력
print('프로그램을 종료합니다.')
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
현재 수증기량 입력 : 9.4
현재 온도 입력 : 20
현재 습도는 54.33526011560693% 입니다 .
프로그램을 종료합니다 .
```

[실습 4] 튜플 활용하기 → 시험점수가 가장 높은 학생은?

[코드] # 주석 작성 필수

```
#4.  
### 튜플  
names = ('choi', 'han', 'jung', 'kang', 'kim', 'lee', 'moon', 'na', 'park',  
         'son')  
nums = (93, 50, 92, 68, 80, 90, 65, 100, 75, 75)  
  
### 처리  
a = max(nums) #max() 함수 활용  
b = (nums.index(a)) #a가 있는 해당 index를 찾음  
  
### 출력  
print(f'시험 점수가 가장 높은 학생은 {names[b]}, 점수는 {a} 입니다.') #출력
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
시험 점수가 가장 높은 학생은 na, 점수는 100 입니다.
```

[실습 5] 딕셔너리 활용하기 → 프로그래밍 언어 개발연도 안내 챗봇

[코드] # 주석 작성 필수

```
#5.
### 딕셔너리 선언
langs = {'c':1972, 'java':1995, 'python':1991, 'go': 2009, 'pascal':1969}

### 챗봇 프로그램 만들기!(while, if문 등을 활용해보세요!)
while(True): #while 무한루프
    n = input('프로그래밍 언어 입력(0입력시 종료): ')
    if(n == 'C' or n == 'c'):
        print(n, "언어는 ", langs['c'], "년에 태어났어요.")
    elif(n == 'JAVA' or n == 'java'):
        print(n, "언어는 ", langs['java'], "년에 태어났어요.")
    elif(n == 'PYTHON' or n == 'python'):
        print(n, "언어는 ", langs['python'], "년에 태어났어요.")
    elif(n == 'GO' or n == 'go'):
        print(n, "언어는 ", langs['go'], "년에 태어났어요.")
    elif(n == 'PASCAL' or n == 'pascal'):
        print(n, "언어는 ", langs['pascal'], "년에 태어났어요.")

    elif(n == '0'):
        print('챗봇 프로그램 종료됩니다.')
        break
    else:
        print("등록되지 않은 언어입니다.")
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
프로그래밍 언어 입력 (0입력시 종료): c
c 언어는 1972년에 태어났어요.
프로그래밍 언어 입력 (0입력시 종료): java
java 언어는 1995년에 태어났어요.
프로그래밍 언어 입력 (0입력시 종료): go
go 언어는 2009년에 태어났어요.
프로그래밍 언어 입력 (0입력시 종료): haskell
등록되지 않은 언어입니다.
프로그래밍 언어 입력 (0입력시 종료): 0
챗봇 프로그램 종료됩니다.
```

[과제 1] 버킷리스트 작성하기

[코드] # 주석 작성 필수

```
#6. 버킷리스트 작성
### 입력

ls = []

### 출력

while(True): #무한루프
    n = int(input('버킷리스트 기능 선택(1.추가 2.삭제 0.종료): '))
    if(n == 1): #입력값이 1이면 append 이용해서 추가
        a = input('추가할 내용: ')
        ls.append(a)
        print('*****')
        for i in ls: #리스트 출력
            print(i)
        print('*****')
        continue

    elif(n==2): #입력값이 2이면 remove 이용해서 삭제
        b = input('삭제할 내용: ')
        ls.remove(b)
        print('*****')
        for i in ls: #리스트 출력
            print(i)
        print('*****')
        continue

    elif(n==0): #입력값이 0이면 종료
        print('프로그램을 종료합니다.')
        break;

    else: #입력값이 0,1,2가 아니면 아래와 같이 출력
        print('기능 선택 오류. 다시 선택하세요')
```

[실행 결과]

```
버킷리스트 기능 선택(1.추가 2.삭제 0.종료): 1
추가할 내용: 파이썬 마스터
*****
파이썬 마스터
*****
버킷리스트 기능 선택(1.추가 2.삭제 0.종료): 1
추가할 내용: 혼자 여행가기
*****
파이썬 마스터
혼자 여행가기
*****
버킷리스트 기능 선택(1.추가 2.삭제 0.종료): 1
추가할 내용: 고양이 키우기
*****
파이썬 마스터
혼자 여행가기
고양이 키우기
*****
버킷리스트 기능 선택(1.추가 2.삭제 0.종료): 2
삭제할 내용: 고양이 키우기
*****
파이썬 마스터
혼자 여행가기
*****
버킷리스트 기능 선택(1.추가 2.삭제 0.종료): 9
기능 선택 오류. 다시 선택하세요
버킷리스트 기능 선택(1.추가 2.삭제 0.종료): 0
프로그램을 종료합니다.
```

[과제 2] 키와 표준 몸무게를 리스트에 저장하기

[코드] # 주석 작성 필수

```
#7. 키와 표준 몸무게
###list 생성
height = []
weight = []

###리스트에 원소 추가
for i in range(150,181,3):
    height.append(i)    #height 원소 추가

for i in range(len(height)):
    a = (height[i]-100)*0.9    #a 변수에 계산값 저장
    weight.append(a)    #Weight 원소 추가

### 출력
for i in range(len(height)):
    print(f'{i+1:3d}    {height[i]}cm    {weight[i]:.1f}kg')
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

1	150cm	45.0kg
2	153cm	47.7kg
3	156cm	50.4kg
4	159cm	53.1kg
5	162cm	55.8kg
6	165cm	58.5kg
7	168cm	61.2kg
8	171cm	63.9kg
9	174cm	66.6kg
10	177cm	69.3kg
11	180cm	72.0kg

[과제 3] 바둑판

[코드] # 주석 작성 필수

```
#8. (과제 3) 바둑판
### 입력 및 변수 선언
#바둑판
go = [[0, 0, 0, 0, 0, 0, 0, 0, 0],
      [0, 1, 0, 1, 2, 1, 2, 1, 0],
      [0, 2, 1, 1, 1, 2, 2, 0, 0],
      [0, 0, 2, 2, 2, 1, 0, 2, 0],
      [0, 0, 0, 0, 0, 1, 0, 2, 1],
      [0, 0, 0, 2, 0, 1, 2, 1, 0],
      [0, 0, 0, 2, 1, 0, 1, 1, 0],
      [0, 0, 0, 1, 1, 0, 0, 0, 0],
      [0, 0, 0, 0, 2, 2, 2, 0, 0]]

black = 0 # 흑돌 갯수 변수
white = 0 # 백돌 갯수 변수

### 1) 돌 개수 세기
for i in range(len(go)):
    black += go[i].count(1) #흑돌 개수 카운트
    white += go[i].count(2) #백돌 개수 카운트
print('*****') #돌 개수 출력
print(f'흑돌의 개수: {black}')
print(f'백돌의 개수: {white}')
print('*****')

### 2) 바둑알 특수문자로 변경하기
for i in range(len(go)):
    for j in range(len(go[i])):
        if(go[i][j]==0): #0 이면..
            go[i][j]= '⬜'
        elif(go[i][j]==1): #1 이면..
            go[i][j]= '●'
        elif(go[i][j]==2): #2 이면..
            go[i][j]= '○'
        go[i][j]= '○'
print('바둑알을 특수문자로 바꿔보자!(흑돌:●, 백돌:○, 돌 없음:⬜)')
for i in go: #출력 코드
    for j in i:
        print(f'{j}', end='') #공백 생성
    print()
print('*****')

### 3) 어떤 좌표에 돌이 있는지 출력하기
while(True): #무한루프
    x = int(input('X 축 좌표값을 입력하세요(1~9, 종료시 -1 입력): '))
```


[실행 결과]

페이지 9 / 11

[과제 4] 행성까지의 여행 시간은?(딕셔너리)

[코드] # 주석 작성 필수

```
#9. (과제 4 번) 행성까지의 시간
### 입력
#주어진 딕셔너리
dic = {'수성': 91_700_000, '금성': 41_400_000, '화성': 78_400_000,
       '목성': 628_700_000, '토성': 1_277_400_000, '천왕성': 2_750_400_000,
       '해왕성': 4_347_400_000 }

#행성 이름 및 이동속도 입력받기
name = input('행성이름: ')
speed = int(input('이동속도(km/h): '))
key = dic[name] #행성간의 거리는 key 값으로 받아옴

time = key / speed #행성간의 이동시간 계산 방법
print(f'이동 시간: 약 {time:2f}시간')

#년, 월, 일, 시간 계산 방법
year = int(time // 8760) #년
time -= (year*8760)
mon = int(time // 720) #월
time -= (mon*720)
days = int(time // 24) #일
hour = int(time-(days *24)) #시간
print(f'시간 변환: 약{year}년 {mon}월 {days}일 {hour}시간')
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
행성이름 : 화성
이동 속도 (km/h): 300
이동 시간 : 약 261333.333333시간
시간 변환 : 약 29년 10월 3일 21시간
(base) youkyoung-na@nayugyeong-ui-Mac
.py"
행성이름 : 해왕성
이동 속도 (km/h): 700
이동 시간 : 약 6210571.428571시간
시간 변환 : 약 708년 11월 23일 19시간
(base) youkyoung-na@nayugyeong-ui-Mac
```

[과제 5] 특정 학생의 점수 검색하기(튜플 + 이진탐색)

[코드] # 주석 작성 필수

```
#10. 과제 5 번
### 입력값 (알파벳순으로 정렬되어있음)
members = (('choi', 93), ('han', 50), ('jung', 92), ('kang', 68), ('kim', 80),
           ('lee', 90), ('moon', 65), ('na', 100), ('park', 75), ('song', 75))

search = input("학생의 이름(성) 입력: ")
number = -1 # 점수 초기화

start = 0
end = (len(members)-1) #인덱스 끝값이기 때문에 len 에 -1 을 해준다.

for n, m in members: ### 이 부분을 이진 탐색에 맞게 수정해주세요.
    mid = (start + end) // 2 #중앙값
    if n == search: # 찾는 값이 있으면 점수를 number 에 저장하고 반복 종료
        number = m
        break
    elif n < search: #만약 값을 못찾고 search 가 오른쪽에 있다면..
        start = mid + 1 #start 는 이전의 mid 값에서 한칸 오른쪽으로
    else: #search 가 왼쪽에 있는 경우
        end = mid -1 #end 가 mid 값의 한칸 왼쪽으로

if number > -1: # 값이 업데이트가 됐다면..
    print(search, number)
else: #값이 업데이트가 안됐다면...
    print("찾는 학생이 없습니다.")
```

[실행 결과] # 제공된 실행 결과와 동일하게 나와야 함

```
학생의 이름(성) 입력: na      학생의 이름(성) 입력: kim
na 100                        kim 80
```