

# PYTHON\_PROGRAMMING

## 리스트



# 리스트

- ❖ 리스트(List)는 데이터의 목록을 다루는 자료형
- ❖ []대괄호로 명명 한다.
- ❖ 리스트 안에는 어떠한 자료형도 포함시킬 수 있다

❖ 예)

- `Ls = []`
- `Ls = ["서울", "경기도"...]`
- `Ls = ['서울', 100, 1.111...]`
- `Ls = [10, 20, 30...]`

0	
1	
2	
3	
4	
5	
6	...

# 리스트

- ❖ 리스트는 박스(변수)를 한 줄로 붙인 뒤에 박스 전체의 이름(aa)을 지정.
  - ❖ 각각은 aa[0], aa[1], aa[2], aa[3]과 같이 번호(첨자)를 붙여서 사용.
- 

## 리스트 사용

---

ls = [10 , 20 , 30 , 40]

ls[0] => 10 ( 0번째 인덱스)

ls[1] => 20 ( 1번째 인덱스)

ls[2] => 30 ( 2번째 인덱스)

ls[3] => 40 ( 3번째 인덱스)

---

# 예제

## ❖ 리스트를 사용하는 이유

예)

a,b,c,d = 0,0,0,0

Sum = 0

a = int(input("첫번째 숫자 입력 : "))

b = int(input("두번째 숫자 입력 : "))

c = int(input("세번째 숫자 입력 : "))

d = int(input("네번째 숫자 입력 : "))

Sum = a + b + c + d

print(" 합 계 : ", Sum)

## 예제

```
ls = [500 , 200 , 300 , 400]; Sum = 0
```

```
print("ls : ",ls)
```

```
print("ls[0] :", ls[0])
```

```
print("ls[1] :", ls[1])
```

```
print("ls[2] :", ls[2])
```

```
print("ls[3] :", ls[3])
```

## 예제

```
ls = [0 , 0 , 0 , 0]; Sum = 0
```

```
ls[0]=int(input("첫번째 숫자 입력 : "))
```

```
ls[1]=int(input("두번째 숫자 입력 : "))
```

```
ls[2]=int(input("세번째 숫자 입력 : "))
```

```
ls[3]=int(input("네번째 숫자 입력 : "))
```

```
Sum = ls[0] + ls[1] + ls[2] + ls[3]
```

```
print("ls[0] :", ls[0])
```

```
print("ls[1] :", ls[1])
```

```
print("ls[2] :", ls[2])
```

```
print("ls[3] :", ls[3])
```

```
print("리스트의 합 : %d" % Sum)
```

## 예제

```
ls = [0 , 0 , 0 , 0]; Sum = 0
```

```
print("len(ls) : ",len(ls))
```

```
for i in range(len(ls)):
```

```
    ls[i]=int(input(str(i)+"째 숫자 입력 : "))
```

```
    Sum += ls[i]
```

```
for i in range(len(ls)):
```

```
    print("ls[%d] :", (i,ls[i]))
```

```
print("리스트의 합 :", Sum)
```

❖ while문으로 바꾸시오

## 예제

```
ls = [0 , 0 , 0 , 0]
Sum , i = 0,0
while i<len(ls):
    ls[i]=int(input(str(i)+"번째 숫자 입력 : "))
    Sum += ls[i]
    i+=1
else: i=0;
while i<len(ls):
    print("ls[%d] :", (i,ls[i]))
    i+=1
print("리스트의 합 :", Sum)
```



# 리스트 slicing

```
ls = [10 , 20 , 30 , 40]
```

```
print("ls : ",ls)
```

```
print("\nls[1:3] => ls[1] ~ [2] :",ls[1:3])
```

```
print("ls[0:3] => ls[0] ~ [2] :",ls[0:3])
```

```
print("ls[2:] => ls[2] ~ [끝까지] :",ls[2:])
```

```
print("ls[:2] => ls[0] ~ [2] :",ls[:2])
```

## 리스트 (얕은 복사)

```
ls = [10 , 20 , 30 , 40]
```

```
arr = ls
```

```
print("ls : {} ls , id : {}".format(ls,id(ls)))
```

```
print("arr : {} arr , id : {}".format(arr,id(arr)))
```

## 리스트 (얕은 복사)

```
ls = [10 , 20 , 30 , 40]
```

```
arr = ls
```

```
arr[2]=20000
```

```
print("ls : {} , ls id : {}".format(ls,id(ls)))
```

```
print("arr : {} , arr id : {}".format(arr,id(arr)))
```

## 리스트 (깊은 복사)

```
ls = [10 , 20 , 30 , 40]
```

```
arr = ls[:]
```

```
arr[2]=20000
```

```
print("ls : {} , ls id : {}".format(ls,id(ls)))
```

```
print("arr : {} , arr id : {}".format(arr,id(arr)))
```

## 리스트 (깊은 복사)

```
import copy
ls = [10 , 20 , 30 , 40]
#arr = ls[:]
arr = copy.deepcopy(ls)
arr[2]='deepcopy'

print("ls : {} , ls id : {}".format(ls,id(ls)))
print("arr : {} , arr id : {}".format(arr,id(arr)))
```

## 리스트 연산

```
ls = [ 10 , 20 , 30 ]  
arr = [ 40 , 50 , 60 ]
```

```
print("ls : " , ls)  
print("arr : " , arr)
```

```
Str = ls + arr  
print("ls + arr => Str : " , Str)
```

```
string = ls * 3  
print("ls * 3 => string : " , string)
```

❖ 반복문을 이용해서 원하는 연산 되도록 만드시오.

## 리스트 연산 풀이

```
ls = [ 10 , 20 , 30 ]  
arr = [ 40 , 50 , 60 ]
```

```
print("ls : " , ls)  
print("arr : " , arr)  
Str = [0,0,0]  
for i in range(len(Str)):  Str[i] = ls[i]+arr[i]
```

```
print("ls + arr => Str : " , Str)  
string=[0,0,0]  
for i in range(len(string)):  string[i] = ls[i]*3
```

```
print("ls * 3 => string : " , string)
```

# 선택정렬

## ❖ 정의

- 첫째 자리에 원하는 값을 위치하는 것으로 오름차순과 내림차순에 따라 값이 변할 수 있다. 오름 차순을 기준으로 하였을 경우 앞에 있는 값과 그 이후의 값들을 비교하여 가장 작은 값을 그 위치에 놓는다. 다음 위치에 있는 값과 그 이후의 값들을 비교하여 그 중 가장 작은 값을 놓는다. 정렬이 끝날 때까지 이를 반복하면 전체적으로 값이 정렬이 이루어 진다.

## ❖ 오름차순

- 수치가 점점 올라가는 수
- 예) 1, 2, 3, 4, 5
- 예) 가, 나, 다, 라, 마

## ❖ 내림차순

- 수치가 점점 내려가는 수
- 예) 5, 4, 3, 2, 1
- 예) 마, 라, 다, 나, 가



# 선택정렬 알고리즘

## ❖ 오름차순

정렬전	4	8	2	7	6	비교
1차	4	8				$4 > 8$
2차	4		2			$4 > 2$
3차	2		4			swap
4차	2			7		$2 > 7$
5차	2				6	$2 > 6$
	2	8	4	7	6	

# 선택정렬 알고리즘

## ❖ 오름차순

정렬전	2	8	4	7	6	비교
1차		8	4			$8 > 4$
2차		4	8			swap
3차		4		7		$4 > 7$
4차		4			6	$4 > 6$
		4	8	7	6	

# 선택정렬 알고리즘

## ❖ 오름차순

정렬전	2	4	8	7	6	비교
1차			8	7		$8 > 7$
2차			7	8		swap
3차			7		6	$7 > 6$
4차			6		7	swap
			6	8	7	

# 선택정렬 알고리즘

## ❖ 오름차순

정렬전	2	4	6	8	7	비교
1차				8	7	$8 > 7$
2차				7	8	swap
완료	2	4	6	7	8	

## 선택정렬 알고리즘 예제

```
ls = [4,8,2,7,6]
```

```
i,j=0,0
```

```
for i in range(4):
```

```
    for j in range(i+1,5):
```

```
        if ls[i]>ls[j]:
```

```
            ls[i],ls[j]=ls[j],ls[i]
```

```
print(ls)
```

# 순위구하기

점수	등수
82	3

1차
등수

2차
등수

3차
등수

4차
등수

5차
등수

82	85	76	79	96
----	----	----	----	----

82				
1				

82	85			
2				

82		76		
2				

82			79	
2				

82				96
3				

비교
----

82<82	동점 변경
-------	----------

82<85	순위 변경
-------	----------

82<76	동점 변경
-------	----------

82<79	동점 변경
-------	----------

82<96	순위 변경
-------	----------

# 순위구하기

점수	등수
82	3
85	2

1차
등수

2차
등수

3차
등수

4차
등수

5차
등수

82	85	76	79	96
----	----	----	----	----

82	85			
	1			

	85			
	1			

	85	76		
	1			

	85		79	
	1			

	85			96
	2			

비교
----

82<85	동점 비교
-------	----------

85<85	동점 비교
-------	----------

85<76	동점 비교
-------	----------

85<79	동점 비교
-------	----------

85<96	순위 변경
-------	----------

# 순위구하기

점수	등수
82	3
85	2
76	5

1차
등수

2차
등수

3차
등수

4차
등수

5차
등수

82	85	76	79	96
----	----	----	----	----

82		76		
		2		

	85	76		
		3		

		76		
		3		

		76	79	
		4		

		76		96
		5		

비교
----

76<82	동점 변경
-------	----------

76<85	동점 순위변경
-------	------------

76<76	동점 변경
-------	----------

76<79	동점 변경
-------	----------

76<96	동점 순위변경
-------	------------



# 순위구하기

점수	등수
82	3
85	2
76	5
79	4

1차
등수

2차
등수

3차
등수

4차
등수

5차
등수

82	85	76	79	96
----	----	----	----	----

82			79	
			2	

	85		79	
			3	

		76	79	
			3	

			79	
			3	

			79	96
			4	

비교
----

79<82	동점 변경 없음
-------	-------------

79<85	동점 순위 변경
-------	-------------

79<76	동점 변경 없음
-------	-------------

79<79	동점 변경 없음
-------	-------------

79<96	동점 순위 변경
-------	-------------

# 순위구하기

점수	등수
82	3
85	2
76	5
79	4
96	1

1차
등수

2차
등수

3차
등수

4차
등수

5차
등수

82	85	76	79	96
----	----	----	----	----

82				96
				1

	85			96
				1

		76		96
				1

			79	96
				1

				96
				1

비교
----

96<82	변동 없음
-------	----------

96<85	순위 변동
-------	----------

96<76	변동 없음
-------	----------

96<79	변동 없음
-------	----------

96<96	순위 변동
-------	----------

# 풀이

```
jumSu = [82,85,76,79,96]
rank = [1,1,1,1,1]
i,j=0,0
for i in range(5):
    for j in range(5):
        if jumSu[i] < jumSu[j]:
            rank[i]+=1
i=0
print("점수\t등수")
while i<5:
    print(jumSu[i],"\t",rank[i])
    i+=1
```

# 리스트 조작 함수

함 수	설 명	사용법
append()	제일 뒤에 값 추가한다	LM.append(값)
pop()	제일 뒤의 값을 빼고 빼낸 값 삭제	LM.pop()
sort()	항목 정렬	LM.sort()
reverse()	항목 순서를 역순으로 변경	LM.reverse()
index()	지정한 값을 찾아서 그 위치를 반환	LM.index(찾을 값)
insert()	지정된 위치에 값을 삽입한다	LM.insert(위치, 값)

# 리스트 조작 함수

함 수	설 명	사용법
<code>remove()</code>	리스트에서 지정한 값을 제거. 단 지정한 값이 여러 개일 경우 첫 번째 값만 지운다	<code>LM.remove(지울값)</code>
<code>extend()</code>	리스트 뒤에 리스트를 추가한다. 리스트의 더하기(+)연산과 동일한 기능을 한다	<code>LM.extend(LM)</code>
<code>count()</code>	리스트에서 찾을 값의 개수를 센다	<code>LM.count(찾을 값)</code>
<code>del()</code>	리스트에서 해당 위치의 항목을 삭제	<code>del(LM[위치])</code>
<code>len()</code>	리스트에 포함된 전체 항목의 개수를 센다	<code>len(LM)</code>

## 리스트 추가 ( append, len )

```
ls = [10,20,30]
```

```
ls.append(1000)
```

```
for i in range(len(ls)):
```

```
    print("ls[{}] : {}".format(i,ls[i]))
```

```
print("리스트의 총 개수 : ",len(ls))
```

```
print("ls : ",ls)
```

```
ls=[]
```

```
print("ls초기화 후 : ",ls)
```

## 예제

```
ls = []  
for i in range(0 , 4) :  
    ls.append(0)  
Sum = 0  
  
for i in range(0 , len(ls)) :  
    ls[i] = int(input(str(i+1) + "번째 숫자 : "))  
    Sum += ls[i]  
  
for i in range(0 , len(ls)) :  
    print("입력 받은 값 ls[{}] : {}".format( i , ls[i] ))  
print("합 계 : %d " % Sum)
```

## 예제

```
num = int(input('몇개의 공간 만들겠습니까? :'))
ls = []
Sum = 0
for i in range(num) :
    ls.append(int(input(str(i+1) + "번째 숫자 : ")))
    Sum += ls[i]

for i in range(0 , len(ls)) :
    print("입력 받은 값 ls[{}] : {}".format( i , ls[i] ))
print("합 계 :", Sum)
```



## 리스트 조작 함수( pop, sort, reverse )

```
List = [ 30 , 20 , 10 ]  
print("현재 리스트 : " , List)
```

```
List.append(40)  
print("append(40) 후 리스트 : " , List)
```

```
print("pop() 으로 추출한 값 : " , List.pop())  
print("pop() 후 리스트 : " , List)
```

```
List.sort()  
print("sort() 후 리스트 : " , List)
```

```
List.reverse()  
print("reverse() 후 리스트 : " , List)  
del(List[2])  
print("del() 후 리스트 : " , List)
```

## 리스트 조작 함수( index, insert, remove, extend )

```
List = [ 30 , 20 , 10 ]
```

```
print("현재 리스트 : " , List)
```

```
print(" 10 값의 위치 : " , List.index(10))
```

```
List.insert(2,200)
```

```
print("insert(2,200) 후 리스트 : " , List)
```

```
List.remove(200)
```

```
print("remove(200) 후 리스트 : " , List)
```

```
List.extend( [ 555 , 666 , 555 ] )
```

```
print("extend( [ 555 , 666 , 555 ] ) 후의 리스트 : " , List)
```

```
print("555 값의 개수 : " , List.count(555))
```

# Quiz

- ❖ List = [ "김개똥" , "2002년입사" , "잘못된 사항" ,"등급B" ]
- ❖ 아래와 같이 기본 저장 값을 출력 후 사용자로 부터 지울값을 입력받아 지우고,
- ❖ 3명을 같은 내용으로 추가 하시오

['김개똥' , '2002년입사' , '잘못된 사항' , '등급B']

지울 값 입력 : 잘못된 사항

추가할 이름 입력 : 홍길동

추가할 이름 입력 : 김말이

추가할 이름 입력 : 개똥이

김개똥/2002년입사/등급B/

홍길동/2002년입사/등급B/

김말이/2002년입사/등급B/

개똥이/2002년입사/등급B/

# 풀이

```
import copy
List = [ "김개똥" , "2002년입사" , "잘못된 사항", "등급B" ]
print(List)
removeName = input('지울 값 입력 : ')
List.remove(removeName)
#del(List[2])
copyList = copy.deepcopy(List)
for i in range(3):
    changeName=input('추가할 이름 입력 : ')
    copyList[0]=changeName
    List.extend(copyList)
for i in range(0,len(List)):
    if i % 3 ==0 and i != 0:
        print()
    print(List[i],end="/")
```

# 문제

- ❖  $ls = [10, 5, 20, 7, 9, 31, 12, 11, 19, 32]$
- ❖ 리스트 2개를 만들어서 홀수번째의 값, 짝수번째의 값을 따로 넣고 짝수번째와 홀수번째의 차를 또다른 리스트에 넣어놓고 출력하시오.  
(결과 :  $[5, 13, -22, 1, -13]$  )
- ❖  $ls$ 의 값 중 인덱스 홀수 번째와 짝수 번째의 합과 차를 구하시오 (짝수번째  $[0, 2, 4, 8] - [1, 3, 5, 7, 9]$  홀수번째)  
(결과 :  $-16$  )
- ❖  $ls$ 에 저장된 값을  $invertLs$ 에 거꾸로 저장하시오
- ❖  $ls$ 의 값을 오름차순으로  $sortLs$ 에 저장 후 출력
- ❖  $ls$ 의 값을 내림차순으로  $reverseLs$ 에 저장 후 출력

## 풀이 (2번 문제)

```
ls = [10,5,20,7,9,31,12,11,19,32]
print(ls)
evenSum=[0,0,0,0,0]
oddSum=[0,0,0,0,0]
result = [0,0,0,0,0]
k=0
for i in range(0,10,2):
    evenSum[k] = ls[i]
    k+=1
k=0
for i in range(1,10,2):
    oddSum[k] = ls[i]
    result[k]=evenSum[k]-oddSum[k]
    k+=1
print('even : ',evenSum)
print('odd : ',oddSum)
print(result)
```

# 풀이

#3번문제 풀이

```
ls = [10,5,20,7,9,31,12,11,19,32]
i,j,oddSum,evenSum = 0,0,0,0
for i in range(1,10,2):
    oddSum+=ls[i]
for j in range(0,10,2):
    evenSum+=ls[j]
print(evenSum-oddSum)
```

# 풀이

#4번 문제 (1번 이어서 치면 됨)

```
j=10
invertLs=[0,0,0,0,0,0,0,0,0,0]
for i in range(0,10,1):
    j-=1
    invertLs[i]=ls[j]
print("ls : ",ls)
print("invertLs : ",invertLs)
```



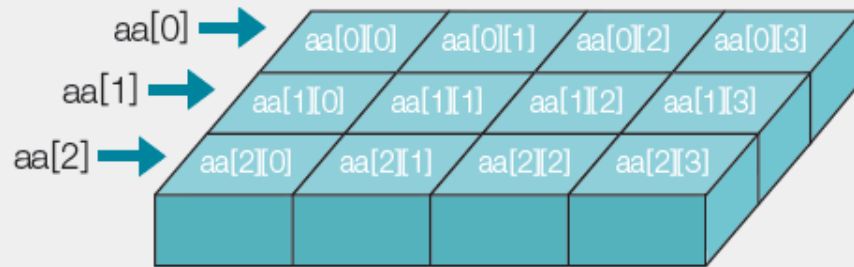
# 풀이

5번 문제 풀이 1,2번 이어서 진행

```
sortLs = [0,0,0,0,0,0,0,0,0,0]
for i in range(0,10,1):
    sortLs[i]=ls[i]
j=0
for i in range(0,10,1):
    for j in range(i+1,10,1):
        if sortLs[i]>sortLs[j]:
            sortLs[i],sortLs[j] = sortLs[j],sortLs[i]
print("sort : ",sortLs)
```

## 2차원 리스트

```
aa = [ [ 1, 2, 3, 4] ,  
       [5, 6, 7, 8] ,  
       [9, 10, 11, 12] ]
```



전체 리스트 이름 : aa

## 예제

```
aa=[[1,2,3,4],  
    [5,6,7,8],  
    [9,10,11,12]]  
print('[0][0]',aa[0][0])  
print('[0][1]',aa[0][1])  
print('[0][2]',aa[0][2])  
print('[0][3]',aa[0][3])  
print('[1][0]',aa[1][0])  
print('[1][1]',aa[1][1])
```

❖ 반복문을 이용하여 모든 값 출력

풀이

```
aa=[  
    [1,2,3,4],  
    [5,6,7,8],  
    [9,10,11,12]  
]  
for i in aa:  
    for k in i:  
        print(k,end='Wt')  
    print()  
print()  
for i in range(len(aa)):  
    for k in range(len(aa[i])):  
        print(aa[i][k],end='Wt')  
    print()
```

## 예제 (얕은 복사)

```
aa=[  
    [1,2,3,4],  
    [5,6,7,8],  
    [9,10,11,12]  
]  
a = aa[0]  
a[1]=2000000000  
  
print('[0]',aa[0])  
print(a)
```

❖ 깊은 복사로 변경 하시오

## 풀이 (깊은 복사)

```
import copy
aa=[
    [1,2,3,4],
    [5,6,7,8],
    [9,10,11,12]
]
#a = aa[0][:]
a = copy.deepcopy(aa[0])
a[1]=2000000000

print('[0]',aa[0])
print(a)
```

# Quiz

❖ 반복문을 이용하여 아래와 같이 출력 하시오.

변수 : ls\_1 = []; ls\_2 = []; value = 1

=====

1	2	3	4
5	6	7	8
9	10	11	12

## 풀이

```
ls_1 = []; ls_2 = []; value = 1
for i in range(0, 3):
    for k in range(0, 4):
        ls_1.append(value)
        value+=1
    ls_2.append(ls_1)
    ls_1 = []
for i in ls_2:
    for k in i:
        print(k, end="wt")
    print()
print("ls_1 : {}".format(ls_1))
print("ls_2 : {}".format(ls_2))
```



## 예제

```
be = ['2019','12','31']  
print(be)
```

```
af=list(map(int,be))  
print(af)
```

## 예제

```
be = [  
    ['100'],  
    ['200'],  
    ['300']  
]  
print('수정 전 : ',be)  
for i in range(len(be)):  
    be[i]=list(map(int,be[i]))  
print(be)  
for i in range(len(be)):  
    be[i][0]=str(be[i][0]*100)  
print('수정 후 : ',be)
```

# Quiz

❖ 반복문을 이용하여 아래와 같이 출력 하시오.

```
ls =  
    [['이름'], ['나이'], ['주소'], ['지울값'], ['연봉']],  
    [['홍길동'], ['20살'], ['산골자기'], ['지우세요'], ['5000']],  
    [['김개똥'], ['30살'], ['지구촌'], ['지우세요'], ['4500']],  
    ]
```

=====아래와 같이 출력 하시오=====

```
['이름']['나이']['주소']['지울값']['연봉']  
['홍길동']['20살']['산골자기']['지우세요']['5000']  
['김개똥']['30살']['지구촌']['지우세요']['4500']
```

## 풀이 (1)

```
ls = [  
    ['이름'], ['나이'], ['주소'], ['지울값'], ['연봉']],  
    [['홍길동'], ['20살'], ['산골자기'], ['지우세요'], ['5000']],  
    [['김개똥'], ['30살'], ['지구촌'], ['지우세요'], ['4500']],  
]
```

```
print(len(ls))  
for i in range(len(ls)):  
    for j in range(len(ls[i])):  
        print(ls[i][j], end="")  
    print()  
for i in ls:  
    for j in i:  
        print(j, end="")  
    print()
```

# Quiz

❖ 반복문을 이용하여 아래와 같이 출력 하시오.

```
ls =  
    [['이름'], ['나이'], ['주소'], ['지울값'], ['연봉']],  
    [['홍길동'], ['20살'], ['산골자기'], ['지우세요'], ['5000']],  
    [['김개똥'], ['30살'], ['지구촌'], ['지우세요'], ['4500']],  
    ]
```

=====아래와 같이 출력 하시오=====

```
['이름'] ['나이'] ['주소'] ['연봉']  
['홍길동'] ['20살'] ['산골자기'] ['5000']  
['김개똥'] ['30살'] ['지구촌'] ['4500']
```

## 풀이 (2)

```
ls =[
    ['이름'], ['나이'], ['주소'], ['지울값'], ['연봉']],
    [['홍길동'], ['20살'], ['산골자기'], ['지우세요'], ['5000']],
    [['김개똥'], ['30살'], ['지구촌'], ['지우세요'], ['4500']],
]
```

```
print(len(ls))
for i in range(len(ls)):
    for j in range(len(ls[i])):
        if j%3==0 and j != 0:
            del(ls[i][j])
    for j in ls[i]:
        print(j,end=" ")
    print()
```

# Quiz

❖ 반복문을 이용하여 아래와 같이 출력 하시오.(10%인상)

```
ls =  
    [['이름'], ['나이'], ['주소'], ['지울값'], ['연봉']],  
    [['홍길동'], ['20살'], ['산골자기'], ['지우세요'], ['5000']],  
    [['김개똥'], ['30살'], ['지구촌'], ['지우세요'], ['4500']],  
    ]
```

=====아래와 같이 출력 하시오=====

```
['이름'] ['나이'] ['주소'] ['연봉']  
['홍길동'] ['20살'] ['산골자기'] ['5500']  
['김개똥'] ['30살'] ['지구촌'] ['4950']
```

# 풀이

```
ls =  
    [['이름'], ['나이'], ['주소'], ['지울값'], ['연봉']],  
    [['홍길동'], ['20살'], ['산골자기'], ['지우세요'], ['5000']],  
    [['김개똥'], ['30살'], ['지구촌'], ['지우세요'], ['4500']],  
    ]
```

## #1번문제

```
for i in ls:  
    for j in i:  
        print(j, end=' ')  
    print()  
print()
```

## #2번문제

```
for i in range(len(ls)):  
    for j in range(len(ls[i])):  
        if j%3==0 and j != 0:  
            del(ls[i][j])  
    for j in ls[i]:  
        print(j, end=" ")  
    print()
```

## #3번문제

```
for i in range(len(ls)):  
    for j in range(len(ls[i])):  
        if j%3==0 and j != 0 and i != 0:  
            ls[i][j][0]=str(int((int(ls[i][j][0]) *1.1)))  
    print()  
for i in range(len(ls)):  
    for j in ls[i]:  
        print(j, end=" ")  
    print()
```