clear()
This method empties the list (OR) remove all elements from list.

```
>>> list1=[10,20,30,40,50]
>>> print(list1)
[10, 20, 30, 40, 50]
>>> list1.clear()
>>> print(list1)
[]
>>> list2=[10,20,30,40,50]
>>> del list2[:]
>>> print(list2)
[]
```

pop()
This method removes and return removed element. This method removes elements using LIFO (Last In First Out Order). The element added last is removed First. This method is used to implement Stack data structure. Stack allows two operations.

1. push → Adding element
2. pop → Removing element

push operation is done using append method
pop operation is done using pop method

Example:

```
# Implementation of Stack DS using list

stack=[]

stack.append(10)
stack.append(20)
stack.append(30)
print(f'Stack {stack}')
value1=stack.pop()
value2=stack.pop()
value3=stack.pop()
print(f'Stack After Pop Operation {stack}')
print(value1,value2,value3)
```

Output:
Stack [10, 20, 30]
Stack After Pop Operation []
30 20 10

Example:
```
# Implementation of Stack DS using list


undo=[]

undo.append("Formatting")
undo.append("Font Size")
undo.append("Font Color")

print(undo.pop())
print(undo.pop())
print(undo.pop())
```

Output:
Font Color
Font Size
Formatting

reverse()
This method reverse elements within list.

```
>>> list1=[10,20,30,40,50]
>>> print(list1)
[10, 20, 30, 40, 50]
>>> list1.reverse()
>>> print(list1)
[50, 40, 30, 20, 10]
>>> list2=[10,20,30,40,50]
>>> list3=list2[::-1]
>>> print(list2)
[10, 20, 30, 40, 50]
>>> print(list3)
[50, 40, 30, 20, 10]
```

https://www.hackerrank.com/challenges/python-lists/problem?isFullScreen=true

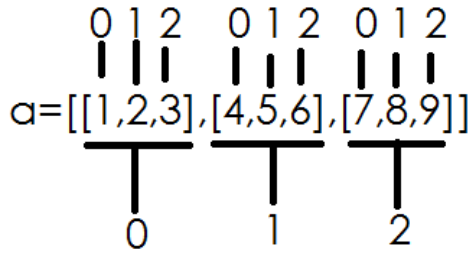
```
list1=[]
n=int(input())

for i in range(n):
    cmd=input().split()
    if cmd[0]=="insert":
        list1.insert(int(cmd[1]),int(cmd[2]))
    elif cmd[0]=="append":
       list1.append(int(cmd[1]))
    elif cmd[0]=="remove":
        list1.remove(int(cmd[1]))
    elif cmd[0]=="print":
        print(list1)
    elif cmd[0]=="pop":
        list1.pop()
    elif cmd[0]=="reverse":
        list1.reverse()
    elif cmd[0]=="sort":
        list1.sort()
```

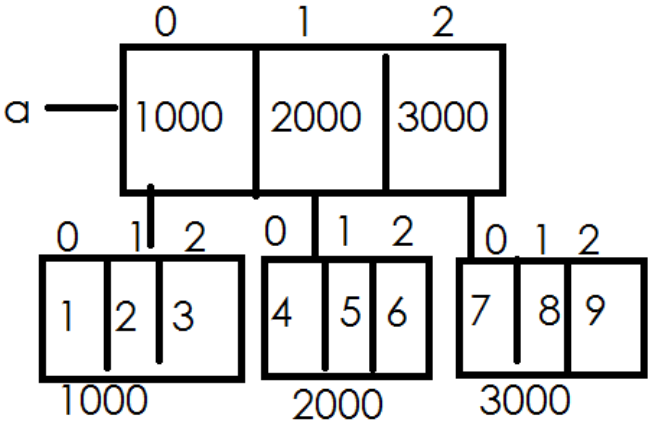
Nested list

List within list is called nested list.

Representing list as an element within list is called nested list. In nested list data is organized logically by dividing into rows and columns.

Example: Matrix






```
print(a[0]) --> [1,2,3]
print(a[1]) --> [4,5,6]
print(a[2]) --> [7,8,9]
print(a[0][0]) --> 1
print(a[0][1]) --> 2
print(a[0][2]) --> 3
print(a[1][0],a[1][1],a[1][2]) 4 5 6
print(a[2][0],a[2][1],a[2][2]) 7 8  9
```

Example:

```
# Write a program to create 2x2 matrix and display

matrix=[[10,20],[30,40]]

for i in range(2): # 0 1
    for j in range(2): # 0 1
        print(matrix[i][j],end=' ')
    print()



for a in matrix:
    for b in a:
        print(b,end=' ')
    print()
```

Output:

```
10 20
30 40
10 20
30 40
```