In [4]:
```
# DICTIONARY
d={'key':'value'}
print(d)
print()
d.update({'name':'Meher'})
d.update({'class':'cse'})
d.update({'section':'A'})
print("After updating i.e, adding elements")
print(d)
print()
print("Accessing Keys : ")
for i in d:
    print(i)
print()
print("Accessing values :")
for i in d:
    print(d[i])
#can have diff combo of data types of keys
```

```
{'key': 'value'}

After updating i.e, adding elements
{'key': 'value', 'name': 'Meher', 'class': 'cse', 'section': 'A'}

Accessing Keys :
key
name
class
section

Accessing values :
value
Meher
cse
A
```

In [5]:
```
# List of dictionaries
d1={'Name':'Meher','Rollno':512}
d2={'Name':'Tejaswini','Rollno':526}
d3={'Name':'Asishwarya','Rollno':516}
l=[d1,d2,d3]
print(l)
```

```
[{'Name': 'Meher', 'Rollno': 512}, {'Name': 'Tejaswini', 'Rollno': 526}, {'Name': 'Asishwarya', 'Rollno': 516}]
```

In [10]:
```
# Taking key and value inputs
n=int(input())
d={}
for i in range(n):
    a,b=map(str,input().split())
    d[a]=b
print(d)
print(a,b)
```

```
2
jack cse
myke ece
{'jack': 'cse', 'myke': 'ece'}
myke ece
```

In [3]:
```
# create user input dictionary
l=[]
l1=['Name','Rollno']
for i in range(2):
    d={}
    for j in l1:
        a=input()
        d[j]=a
    l.append(d)
print(l)
```

```
Meher
512
Keerthana
20a31a0512
[{'Name': 'Meher', 'Rollno': '512'}, {'Name': 'Keerthana', 'Rollno': '20a31a0512'}]
```

In [ ]:
```
a=[1,2,3]
b=a
b[0]=100
print(a)
print(b)
# a also gets updated coz both r poiting to same memory
```

In [6]:
```
l=[]
d={}
for i in range(2):
    d.update({
        'Name':input(),
        'Class':input()
    })
    l.append(d)
print(l)
```

```
dfghj]
ghjkl
fdgui
vbnm
[{'Name': 'fdgui', 'Class': 'vbnm'}, {'Name': 'fdgui', 'Class': 'vbnm'}]
```

In [5]:
```
db=[
    {'abc@gmail.com':'abc'},
    {'def@gmail.com':'def'},
    {'ghi@gmail.com':'ghi'},
]
username=input()
password=input()
temp={username:password}
if temp in db:
    print("Found")
else:
    print("Not found")
```

```
abc@gmail.com
abc
Found
```

In [9]:
```
#arr=[[123],
 #   [4,5,6],
  #   [7,8,9]]

row=2
col=2
arr=[]
for i in range(row):
    ele=[]
    for j in range(col):
        ele.append(int(input("Enter : ")))
    arr.append(ele)
print(arr)
```

```
Enter : 1
Enter : 2
Enter : 3
Enter : 4
[[1, 2], [3, 4]]
```

In [1]:
```
row=3
col=3
arr1=[]
for i in range(row):
    temp=input("Enter elements in a row : ").split(' ')
    ele=list(map(int,temp))
    arr1.append(ele)
print(arr1)

arr2=[]
for i in range(row):
    temp=input("Enter elements in a row : ").split(' ')
    ele=list(map(int,temp))
    arr2.append(ele)
```

```
print(arr2)

res=[[0 for i in range(col)] for i in range(row)]

print("1st array : ",arr1)
print("2nd array : ",arr2)
for i in range(row):
    for j in range(col):
        res[i][j]=arr1[i][j]+arr2[i][j]
print("Sum is : ",res)
```

```
Enter elements in a row : 1 2 3
Enter elements in a row : 1 2 3
Enter elements in a row : 1 2 3
[[1, 2, 3], [1, 2, 3], [1, 2, 3]]
Enter elements in a row : 1 2 3
Enter elements in a row : 1 2 3
Enter elements in a row : 1 2 3
[[1, 2, 3], [1, 2, 3], [1, 2, 3]]
1st array :  [[1, 2, 3], [1, 2, 3], [1, 2, 3]]
2nd array :  [[1, 2, 3], [1, 2, 3], [1, 2, 3]]
Sum is :  [[2, 4, 6], [2, 4, 6], [2, 4, 6]]
```

In [4]:

```
# Transpose of a matrix
row=3
col=3
arr1=[]
for i in range(row):
    temp=input("Enter elements : ").split()
    ele=list(map(int,temp))
    arr1.append(ele)
print(arr1)

res=[[0 for i in range(col)] for i in range(row)]

for i in range(row):
    for j in range(col):
        res[i][j]=arr1[j][i]
print(res)
```

```
Enter elements : 1 2 3
Enter elements : 4 5 6
Enter elements : 7 8 9
[[1, 2, 3], [4, 5, 6], [7, 8, 9]]
[[1, 4, 7], [2, 5, 8], [3, 6, 9]]
```

In [9]:

```
#list slicing can also be applied to strings
l=[1,2,3,4,5,6,7,8,9]
#l[start,stop,step size]
print(l[0:8])
print(l[0:100:2])
print(l[1:])
print(l[::2])
#to reverse a list
print(l[::-1])
#only python supports negative indexing
```

```
[1, 2, 3, 4, 5, 6, 7, 8]
[1, 3, 5, 7, 9]
[2, 3, 4, 5, 6, 7, 8, 9]
[1, 3, 5, 7, 9]
[9, 8, 7, 6, 5, 4, 3, 2, 1]
```

In [16]:

```
# 0,1,1,2,3,5,8,13,21.........
l=[0,1]
s=0
for i in range(5):
    s=l[-1]+l[-2]
    l.append(s)
print(l)
```

```
[0, 1, 1, 2, 3, 5, 8]
```

In [12]:

```
s="hello world"
```

```
print(s)
s=s.capitalize()
print(s)
res=s.split(' ') # default is space
print(res)
#to convert string to list , use split()
#to convert list to string , use .join()
print('-'.join(res))
print(s.title())
s1="HELLO THERE"
s2="hello there"
print(s1.lower())
print(s2.upper())
print(s2.count('l'))
```

```
hello world
Hello world
['Hello', 'world']
Hello-world
Hello World
hello there
HELLO THERE
2
```

In [13]:

```
s1="HELLO world"
res=s1.swapcase() # swap case swaps lower letters to upper and upper to lower
print(s1)
print(res)
```

```
HELLO world
hello WORLD
```

In [25]:

```
# STRING FORMATTING
first="Mr.X is "
age=38
last=" ears old."
print(first+str(age)+last)
print("Mr.X is {} years old".format(age))

num=3.14
print("The square of {} is {}".format(num,num*num))
print("The square of {} is {:10.2f}".format(num,num*num))
print("The square of {:10} is {:.6f}".format(num,num*num))
```

```
Mr.X is 38 ears old.
Mr.X is 38 years old
The square of 3.14 is 9.8596
The square of 3.14 is       9.86
The square of       3.14 is 9.859600
```

In [28]:

```
# fstrings
num=10
print(f"the square of {num} is {num*num:.5f}")
```

```
the square of 10 is 100.00000
```

In [32]:

```
# Exception Handling
a=5
b=0

try:
    print(a/b)
except:
    print("b cannot be zero")
```

```
b cannot be zero
```

In [38]:

```
# CALCULATOR
number1=int(input("Enter 1st number : "))
number2=int(input("Enter 2nd number : "))
print("Choose any one of the operators : + , - , * , / , %  ")
operator=input()
```

```
if operator=='+':
    print(f"Result of addition is {number1+number2}")
elif operator=='-':
    print(f"Result of subtraction is {number1-number2}")
elif operator=='*':
    print(f"Result of multiplication is {number1*number2}")
elif operator=='/':
    try:
        print(f"Result of division is {number1/number2}")
    except:
        print("number2 cannot be zero")
elif operator=='%':
    try:
        print(f"Remainder is {number1%number2}")
    except:
        print("number2 cannot be zero")
```

```
Enter 1st number : 10
Enter 2nd number : 2
Choose any one of the operators : + , - , * , / , %
a
```

In [46]:

```
#take 5 int inputs ana print them
try:
    num1,num2,num3,num4,num5=map(int,input().split())
    print(num1,num2,num3,num4,num5)
except:
    print("Only integer input")
```

```
1 2 3 d 3
Only integer input
```

In [50]:

```
# eval()
print(eval("2+3-4*21/23"))
#any kind of calculation , do it using eval()
```

```
1.347826086956522
```

In [53]:

```
#number to remove ','
num=input()
l=list(num)
if ',' in l:
    l.remove(',')
    print(''.join(l))
```

```
45,789
45789
```

In [55]:

```
#regular functions
#default value functions
#keyword argument functions
#variable length functions

# function defines by def keyword
# def function_name(arguments):
      #function body
def addition(num1,num2):
    res=num1+num2
    return res
print(addition(10,20))
```

```
30
```

In [70]:

```
def prime(num):
    c=0
    for i in range(2,num):
        if num%i==0:
            c+=1
        if c>0:
            return "False"
    else:
```

```
        return "True"
print(prime(13))
```

```
True
```

In [ ]:

```
num=23
for i in range(1,num+1):
    pass#23
for i in range(2,num):
    pass#21
for i in range(2,num//2):
    pass#10
for i in range(2,int(num**0.5)+1):
    pass#4 iterations
```

In [74]:

```
def addition(num1,num2=0):
    res=num1+num2
    return res
num1=10
print(addition(num1))
```

```
10
```

In [79]:

```
def add(a,b,c,d):
    print(a,b,c,d)
add(10,20,d=30,c=40)
#add(10,20,a=30,b=40)
add(c=67,b=78,d=89,a=45)
```

```
10 20 40 30
45 78 67 89
```

In [82]:

```
def add(a,b,*abc):
    print(a)
    print(b)
    print(abc)
add(1,2,3,4,5,6,7,8,9)
```

```
1
2
(3, 4, 5, 6, 7, 8, 9)
```

In [83]:

```
#recursion is function calling it self
def check(n):
    print(n)
    if n>0:
        check(n-1) #base cond determine when the recursion stops
check(5)
```

```
5
4
3
2
1
0
```

In [2]:

```
n=int(input())
l=list(map(int,input().split()))
k=int(input())
if(k>n or k<0 or n<0):
    print("Invalid Input")
else:
    for i in range(k):
        a=l.pop()
        l.insert(0,a)
    s=""
    for i in l:
        s=s+str(i)+" "
    print(s)
```

```
5
```

```
32 56 34 67 12
2
67 12 32 56 34
```

In [8]:

```
N=int(input())
l=list(map(int,input().split()))
a=[]
for i in l:
    if l.count(i)==1:
        a.append(i)
print(a)
```

```
7
1 3 4 6 1 3 6
[4]
```

In [ ]:

```
# Dictionaries
# Login System
# checking username and password
keys=['Name','Password']
l=[]
for i in range(2):
    d={}
    for j in keys:
        values=input()
        d[keys]=values
    l.append(d)
print("Database : ",l)
username=input("Enter username : ")
password=input("Enter password : ")

#Method 1
di={}
di['Name']=username
di['Password']=password
if di in l:
    print("You are successfully logged in")
else:
    print("Please enter correct credentials")
```

In [ ]: