In [1]:

```
###1. Write a program using function USINto reverse a string.
###Sample data: "1234abcd"
###Expected Output: "dcba4321"
def USINto(x):
  return x[::-1]

mytxt = USINto("1234abcd")

print('New Text:',mytxt)
```

```
New Text: dcba4321
```

In [2]:

```
######Write a function that accepts a string and calculate the number of uppercase letters and low
ercase letters.
s =input('Enter the String:')
l,u = 0,0
for i in s:
    if (i>='a'and i<='z'):
        l=l+1                   #counting lower case
    if (i>='A'and i<='Z'):
        u=u+1                   #counting upper case
print('Lower case characters: ',l)
print('Upper case characters: ',u)
```

```
Enter the String:Omsai
Lower case characters:  4
Upper case characters:  1
```

In [4]:

```
###Create a function that takes a list and returns a new list with unique elements of the first li
st.
def unique(list1):
    list_set = set(list1)
    unique_list = (list(list_set))
    for x in unique_list:
        print(x)
list1 = input(list())
print("the unique values from 1st list is")
unique(list1)
```

```
[]1 2 2 3
the unique values from 1st list is
3
1
2
```

In [8]:

```
### Write a program that accepts a sequence of lines as input and prints the lines after making al
l characters in the sentence capitalized.
lines = []
while True:
    l = input()
    if l:
        lines.append(l.upper())
    else:
        break;

for l in lines:
    print(l)
```

```
Omsai
Chaitali
```

```
Chaitali

OMSAI
CHAITALI
```

In [9]:

```
### Write a program that accepts a hyphen-separated sequence of words as input and prints the word
s in a hyphen-separated sequence after sorting them alphabetically.
items=[n for n in input().split('-')]
items.sort()
print('-'.join(items))
```

```
Omsai-Chaitali
Chaitali-Omsai
```

In [10]:

```
### Define a function that can receive two integral numbers in string form and compute their sum a
nd print it in console.
def addition (a,b):
    s = int(a) + int(b)
    return s
num1 = "10"
num2 = "20"
sum = addition (num1, num2)
print ("Sum = ", sum)
```

```
Sum =  30
```

In [11]:

```
######Define a function that can accept two strings as input and print the string with maximum len
gth in console. If two strings have the same length, then the function should print all strings li
ne by line.
def length_of_string(str1, str2):
    if len(str1) == len(str2):
        print(str1)
        print(str2)
    elif len(str1)<len(str2):
            print(str2)
    else:
        print(str1)
string1=input(('Enter the first string:'))
string2=input("Enter the second string:")

length_of_string(string1,string2)
```

```
Enter the first string:Omsai
Enter the second string:Chaitali
Chaitali
```

In [20]:

```
###  Define a function which can generate and print a tuple where the value are square of numbers
between 1 and 20.
```

In [32]:

```
###Write a function called showNumbers that takes a parameter called limit. It should print all th
e numbers between 0 and limit with a label to identify the even and odd numbers.
```

In [33]:

```
###Write a program which can filter() to make a list whose elements are even number between 1 and
20 ( both included)
lis = [1,2,3,4,5]
lis2 = [i for i in lis if i%2 == 0]
print(lis2)
```

```
[2, 4]
```

In [34]:

```
###Write a program which can map() and filter() to make a list whose elements are square of even n
umber in [1,2,3,4,5,6,7,8,9,10]
li = [1,2,3,4,5,6,7,8,9,10]

even_num = map(lambda x: x**2, filter(lambda   x: x%2==0, li))

print(even_num)
```

```
<map object at 0x000001EBC42D3B00>
```

In [38]:

```
###Write a program which can map() and filter() to make a list whose elements are square of even n
umber in [1,2,3,4,5,6,7,8,9,10]
squares = map(lambda x: x**2, range(10))
squares1 = filter(lambda x: x%2==0, squares)
print (squares1)
```

```
<filter object at 0x000001EBC42DF208>
```

In [39]:

```
###Write a function to compute 5/0 and use try/except to catch the exceptions
try:
    x = 5 / 0
except:
    print("Error dividing by zero")
```

```
Error dividing by zero
```

In [50]:

```
###Flatten the list [[1,2,3].,[4,5],[6,7,8]] into [1,2,3,4,5,6,7,8] using reduce
import functools
import operator

Lists = [[1,2,3],[4,5],[6,7,8,]]

List_flat = functools.reduce(operator.iconcat, Lists, [])

print("Original List:",Lists)
print("Flattened List:",List_flat)
```

```
Original List: [[1, 2, 3], [4, 5], [6, 7, 8]]
Flattened List: [1, 2, 3, 4, 5, 6, 7, 8]
```

In [51]:

```
def foo():

    try:

        return 1

    finally:

        return 2

k = foo()

print(k)
```

```
2
```

In [52]:

```
def a():

    try:

        f(x, 4)

    finally:

        print('after f')

    print('after f?')

a()
```

```
after f


---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-52-5016ed6553fe> in <module>
     11     print('after f?')
     12
---> 13 a()

<ipython-input-52-5016ed6553fe> in a()
      3     try:
      4
----> 5         f(x, 4)
      6
      7     finally:

NameError: name 'f' is not defined
```

In [ ]:

```

```