

```
1 # Question 1
2 # Python code to reverse a string
3 # using loop
4
5 def reverse(s):
6     str = ""
7     for i in s:
8         str = i + str
9     return str
10
11 s = "PunjabEnggCollege"
12
13 print ("The original string is : ",end="")
14 print (s)
15
16 print ("The reversed string(using loops) is : ",end="")
17 print (reverse(s))
18
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

```
In [1]: runcell(0, 'C:/Users/Nikhil Saini/Q1.py')
The original string is : PunjabEnggCollege
The reversed string(using loops) is : egelloCggnEbajnuP

In [2]:
```

```
1 # Question 2
2 a=int(input("Enter first number of range: "))
3 b=int(input("Enter last number of range: "))
4 c=int(input("Enter the divisor : "))
5
6 for i in range(a,b+1):
7     if i==0:
8         pass
9     elif i%c==0:
10    print(i)
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

```
In [2]: runcell(0, 'C:/Users/Nikhil Saini/Q2.py')
```

```
Enter first number of range: 2
```

```
Enter last number of range: 13
```

```
Enter the divisor : 2
```

```
2
4
6
8
10
12
```

```
In [3]:
```

```
1 # Question 3
2 from math import sqrt
3 a=float(input("Enter the value of first side: "))
4 b=float(input("Enter the value of second side: "))
5 c=float(input("Enter the value of third side: "))
6 s = (a+b+c)/2
7 #Herons formula = sqrt((s)(s-a)(s-b)(s-c))
8 area_sq = s*(s-a)(s-b)(s-c)
9 if area_sq> 0:
10     print("Area : ", sqrt(area_sq))
11 else:
12     print("Triangle will not form")
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

Variable explorer Help Plots Files

Console 1/A

In [5]: runcell(0, 'C:/Users/Nikhil Saini/Q3.py')

Enter the value of first side: 3

Enter the value of second side: 4

Enter the value of third side: 5

Traceback (most recent call last):

File "C:\Users\Nikhil Saini\Q3.py", line 8, in <module>  
 area\_sq = s\*(s-a)(s-b)(s-c)

**TypeError:** 'float' object is not callable

In [6]:

IPython console History

LSP Python: ready

conda (Python 3.8.8)

Line 12, Col 37

ASCII

CRLF

RW

Mem 46%

```
1 # Question 4
2
3 n=5;
4 for i in range(n):
5     for j in range(i):
6         print('*', end='')
7     print('')
8
9 for i in range(n,0,-1):
10     for j in range(i):
11         print('*', end='')
12 print('')
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

In [6]: runcell(0, 'C:/Users/Nikhil Saini/Q4.py')

```
*
* *
* * *
* * * *
* * * * *
* * * * *
* * *
* *
*
```

In [7]:



```
1 # Question 5
2 row=int(input("Enter the number of rows : "))
3 n=0
4 for i in range(0,row+1):
5
6     for j in range(i):
7         if n==26:
8             n=0
9         else:
10             pass
11         y=chr(65+n)
12         print(y,end="")
13         n+=1
14
15 print("")
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

In [1]: runcell(0, 'C:/Users/Nikhil Saini/Q5.py')

Enter the number of rows : 2

A  
BC

In [2]: runcell(0, 'C:/Users/Nikhil Saini/Q5.py')

Enter the number of rows : 10

A  
BC  
DEF  
GHIJ  
KLMNO  
PQRSTU  
VWXYZAB  
CDEFGHIJ  
KLMNOPQRS  
TUVWXYZABC

In [3]: |

```
1 # Question 6
2
3 # First, we will take the input:
4 lower_value = int(input("Please, Enter the Lowest Range Value: "))
5 upper_value = int(input("Please, Enter the Upper Range Value: "))
6
7 print("The Prime Numbers in the input range are: ")
8 for number in range(lower_value, upper_value + 1):
9     if number > 1:
10         for i in range(2, number):
11             if (number % i) == 0:
12                 break
13         else:
14             print(number)
15
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

```
In [3]: runcell(0, 'C:/Users/Nikhil Saini/Q6.py')
```

```
Please, Enter the Lowest Range Value: 20
```

```
Please, Enter the Upper Range Value: 80
```

```
The Prime Numbers in the input range are:
```

```
23
29
31
37
41
43
47
53
59
61
67
71
73
79
```

```
In [4]: |
```

```
1 # Question 7
2 a=[]
3 for i in range(1,501):
4     if i%7==0:
5         if i%11==0:
6             a.append(i)
7
8 print(a) |
```

## Usage

Here you can get help of any object by pressing **Ctrl+I** in front of it, either on the Editor or the Console.

Help can also be shown automatically after writing a left parenthesis next to an object. You can activate this behavior in *Preferences > Help*.

```
In [4]: runcell(0, 'C:/Users/Nikhil Saini/Q7.py')
[77, 154, 231, 308, 385, 462]
```

```
In [5]:
```

```
1 # Question 8
2 num1=int(input("Enter 1st number :"))
3 num2=int(input("Enter 2nd number :"))
4 num3=int(input("Enter 3rd number :"))
5 num4=int(input("Enter 4th number :"))
6 num5=int(input("Enter 5th number :"))
7 num6=int(input("Enter 6th number :"))
8 num7=int(input("Enter 7th number :"))
9 num8=int(input("Enter 8th number :"))
10 num9=int(input("Enter 9th number :"))
11 num10=int(input("Enter 10th number :"))
12
13 a=[num1,num2,num3,num4,num5,num6,num7,num8,num9,num10]
14
15 # part 1
16 print("positive number are")
17 for i in a:
18
19     if i>0:
20         print( i)
21 # part 2
22 print("negative numbers are")
23 for i in a:
24
25     if i <0:
26         print( i)
27 # part 3
28
29 print("odd numbers are")
30 for i in a:
31     if i%2!=0:
32         print(i)
33 # part 4
34 print("even numbers are")
35 for i in a:
36
37     if i%2==0:
38         print(i)
39 # part 5
40 for i in a:
41     print(f"{i}--> occured {a.count(i)} times")
```

```
Enter 8th number :5
Enter 9th number :5
Enter 10th number :7
positive number are
2
3
4
1
3
3
4
5
5
7
negative numbers are
odd numbers are
3
1
3
3
5
5
7
even numbers are
2
4
4
2--> occured 1 times
3--> occured 3 times
4--> occured 2 times
1--> occured 1 times
3--> occured 3 times
3--> occured 3 times
4--> occured 2 times
5--> occured 2 times
5--> occured 2 times
7--> occured 1 times
```

In [6]:



```
1 # Question 9
2
3 l=[]
4
5 n=int(input("Enter the number of elements in list:"))
6 for i in range(0,n):
7     element=input()
8     l.append(element)
9
10 print(l)
11 for i in l:
12     print(f"{i} --> occured {l.count(i)} times")
13
```

```
In [9]: runcell(0, 'C:/Users/Nikhil Saini/Q9.py')
```

```
Enter the number of elements in list: 1
```

```
3
['3']
3 --> occured 1 times
```

```
In [10]: runcell(0, 'C:/Users/Nikhil Saini/Q9.py')
```

```
Enter the number of elements in list: 2
```

```
2
3
['2', '3']
2 --> occured 1 times
3 --> occured 1 times
```

```
In [11]:
```