Question1

Create a function that takes a list of strings and integers, and filters out the list so that it returns a list of integers only.

**Examples**

filter\_list([1, 2, 3, "a", "b", 4]) ➞ [1, 2, 3, 4]

filter\_list(["A", 0, "Edabit", 1729, "Python", "1729"]) ➞ [0, 1729]

filter\_list(["Nothing", "here"]) ➞ []

# Create a function that takes a list of strings and integers,

# and filters out the list so that it returns a list of integers only.

# Define a list to test the function, function1

input = [2,5,'g','h',8]

# Define a function, function1

def function1(l1):

l2 = []

for i in l1:

if type(i)==int:

l2.append(i)

return l2

# Call the function function1 with the input, input1

x = function1(input)

# Print the list which has only integers from the list given

print(x)

Question2

Given a list of numbers, create a function which returns the list but with **each element's index in the list added to itself**. This means you add 0 to the number at index 0, add 1 to the number at index 1, etc...

### Examples

add\_indexes([0, 0, 0, 0, 0]) ➞ [0, 1, 2, 3, 4]

add\_indexes([1, 2, 3, 4, 5]) ➞ [1, 3, 5, 7, 9]

add\_indexes([5, 4, 3, 2, 1]) ➞ [5, 5, 5, 5, 5]

# Given a list of numbers, create a function which returns the list

# but with each element's index in the list added to itself.

# This means you add 0 to the number at index 0, add 1 to the number at index 1, etc...

# Examples ([0, 0, 0, 0, 0]) ➞ [0, 1, 2, 3, 4]

# add\_indexes([1, 2, 3, 4, 5]) ➞ [1, 3, 5, 7, 9]

# add\_indexes([5, 4, 3, 2, 1]) ➞ [5, 5, 5, 5, 5]

# Define a function, function1

def function1(l1):

for i in range(0,len(l1)):

l1[i] = l1[i]+i

return l1

# Check with multiple inputs and verify the result

l2 = function1([0, 0, 0, 0, 0])

print(l2)

l2 = function1([1, 2, 3, 4, 5])

print(l2)

l2 = function1([5, 4, 3, 2, 1])

print(l2)

Question3

Create a function that takes the height and radius of a cone as arguments and returns the volume of the cone rounded to the nearest hundredth. See the resources tab for the formula.



### Examples

cone\_volume(3, 2) ➞ 12.57

cone\_volume(15, 6) ➞ 565.49

cone\_volume(18, 0) ➞ 0

# Create a function that takes the height and radius of a cone as arguments and

# returns the volume of the cone rounded to the nearest hundredth.

# cone\_volume(3, 2) ➞ 12.57

# cone\_volume(15, 6) ➞ 565.49

# cone\_volume(18, 0) ➞ 0

# Define a constant pi

pi = 22/7

# Define a function, cone\_volume

def cone\_volume(height, radius):

v = pi\*height\*radius\*radius/3

return v, round(v/100)\*100 # The volume is rounded off to the nearest hundredth

# call function with multiple inputs and print the cone volume

volume, volume\_roundoff = cone\_volume(3,2)

print("Volume got is ",volume)

print("Volume rounded of to the nearest hundedth is ",volume\_roundoff)

volume, volume\_roundoff = cone\_volume(15,6)

print("Volume got is ",volume)

print("Volume rounded of to the nearest hundedth is ",volume\_roundoff)

volume, volume\_roundoff = cone\_volume(18,0)

print("Volume got is ",volume)

print("Volume rounded of to the nearest hundedth is ",volume\_roundoff)

Question4

This Triangular Number Sequence is generated from a pattern of dots that form a triangle. The first 5 numbers of the sequence, or dots, are:

1, 3, 6, 10, 15

This means that the first triangle has just one dot, the second one has three dots, the third one has 6 dots and so on.

Write a function that gives the number of dots with its corresponding triangle number of the sequence.

### Examples

triangle(1) ➞ 1

triangle(6) ➞ 21

triangle(215) ➞ 23220

# Write a function that gives the number of dots with its corresponding triangle number of the sequence.

# The triangular number sequence is : 1, 3, 6, 10, 15

# Define a function to calculate the Triangular Number Sequence

def function1(n):

num = 0

if n == 1:

num = 1

elif n > 1:

num = n+function1(n-1)

return num

# Test with multiple inputs

print(function1(1))

print(function1(2))

print(function1(3))

print(function1(4))

print(function1(5))

Question5

Create a function that takes a list of numbers between 1 and 10 (excluding one number) and returns the missing number.

### Examples

missing\_num([1, 2, 3, 4, 6, 7, 8, 9, 10]) ➞ 5

missing\_num([7, 2, 3, 6, 5, 9, 1, 4, 8]) ➞ 10

missing\_num([10, 5, 1, 2, 4, 6, 8, 3, 9]) ➞ 7

# Create a function that takes a list of numbers between 1 and 10 (excluding one number)

# and returns the missing number.

# Define a sample list

l1 = [1,2,3,5,6,7,8,9,10]

# Define a function, function1

def function1(list1):

for i in range(1,11):

x = False

for j in list1:

if i==j:

x = True

if x==False:

return i

# call the function, function1

num = function1(l1)

# Print the number

print("The missing number is ",num)