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"]) ➞ []
```

Answer:

```
def filter_list(list1):
    int_list = [x for x in list1 if isinstance(x, int)]
    return int_list
list_int = filter_list([1,2,3,'ok','hello'])
print(list_int)
```

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]
```

Answer:

```
from operator import add


list_add=[]

def new_list(list1):
    index_list = [i for i in range(len(list1))]
    list_add = list(map(add, list1, index_list))
```

return list_add

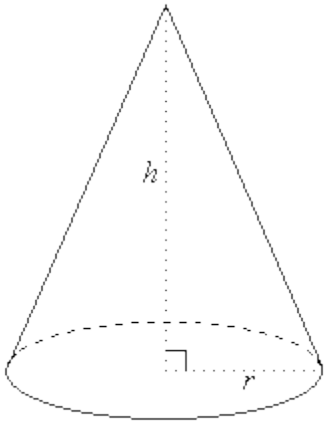
list1 = [0,23,34,43]

added_list = new_list(list1)

print(added_list)

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
```

Answer:

height = 3

radius = 2

def vol_cone(h, r):

vol = round((1/3) * 3.14 * (r ** 2) * h, 2)

print(vol)

vol_cone(height, radius)

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
```

Answer:

```
def triangle(n):
    count = 2
    if n == 1:
        return n
    else:
        for i in range(1, n + 1):
            if i == 1:
                dot = i
            else:
                dot = dot + count
                count+=1
        print(dot)
triangle(215)
```

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
```

Answer:

```
def missing_num(list1):
    new_list = list(range(list1[0], list1[-1]+1))
    missing_num = set(list1) ^ set(new_list)
    print(missing_num)
missing_num([1,2,3,4,6,7,8,9,10])
```