Question 1

Create a function that takes a list of non-negative **integers** and **strings** and return a new list without the strings.

**Examples**

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

filter\_list([1, "a", "b", 0, 15]) ➞ [1, 0, 15]

filter\_list([1, 2, "aasf", "1", "123", 123]) ➞ [1, 2, 123]

def filter\_list(n):

l= []

for i in n:

if isinstance(i, int): # Check if the element is an integer

l.append(i)

print(l)

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

filter\_list([1, "a", "b", 0, 15])

filter\_list([1, 2, "aasf", "1", "123", 123])

Question 2

The "Reverser" takes a string as input and returns that string in reverse order, with the opposite case.

### Examples

reverse("Hello World") ➞ "DLROw OLLEh"

reverse("ReVeRsE") ➞ "eSrEvEr"

reverse("Radar") ➞ "RADAr"

def reverse(n):

l = []

res:str = n[::-1]

print(res.swapcase()) # Convert lowercase to uppercase and vice versa

l.append(res)

reverse("Hello World")

reverse("ReVeRsE")

reverse("Radar")

Question 3

You can assign variables from lists like this:

lst = [1, 2, 3, 4, 5, 6]

first = lst[0]

middle = lst[1:-1]

last = lst[-1]

print(first) ➞ outputs 1

print(middle) ➞ outputs [2, 3, 4, 5]

print(last) ➞ outputs 6

With Python 3, you can assign variables from lists in a much more succinct way. Create variables first, middle and last from the given list using **destructuring assignment** (check the **Resources** tab for some examples), where:

first ➞ 1

middle ➞ [2, 3, 4, 5]

last ➞ 6

Your task is to unpack the list writeyourcodehere into three variables, being first, middle, and last, with middle being everything in between the first and last element. Then print all three variables.

lst = [1, 2, 3, 4, 5, 6]

first, \*middle, last = lst # Unpacking the list

print(first) # Outputs 1

print(middle) # Outputs [2, 3, 4, 5]

print(last) # Outputs 6

Question 4

Write a function that calculates the **factorial** of a number **recursively**.

### Examples

factorial(5) ➞ 120

factorial(3) ➞ 6

factorial(1) ➞ 1

factorial(0) ➞ 1

def factorial(n):

res=1

for i in range(1,n+1):

res \*= i

print(res)

factorial(5)

factorial(3)

factorial(1)

factorial(0)

Question 5

Write a function that moves all elements of one type to the **end** of the list.

### Examples

move\_to\_end([1, 3, 2, 4, 4, 1], 1) ➞ [3, 2, 4, 4, 1, 1]

# Move all the 1s to the end of the array.

move\_to\_end([7, 8, 9, 1, 2, 3, 4], 9) ➞ [7, 8, 1, 2, 3, 4, 9]

move\_to\_end(["a", "a", "a", "b"], "a") ➞ ["b", "a", "a", "a"]

def move\_to\_end(lst, target):

# Create two lists: one for non-target elements and one for target elements

non\_target = [x for x in lst if x != target]

target\_elements = [x for x in lst if x == target]

# Combine non-target elements followed by target elements

return non\_target + target\_elements

# Test cases

print(move\_to\_end([1, 3, 2, 4, 4, 1], 1)) # ➞ [3, 2, 4, 4, 1, 1]

print(move\_to\_end([7, 8, 9, 1, 2, 3, 4], 9)) # ➞ [7, 8, 1, 2, 3, 4, 9]

print(move\_to\_end(["a", "a", "a", "b"], "a")) # ➞ ["b", "a", "a", "a"]