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]

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

# eg : 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]

import re

# Define a function extractintegers

def extractintegers(l1):

l2 = [i for i in l1 if isinstance(i,int)]

return l2

# check with multiple inputs if the function is working correctly

newList = extractintegers([1, 2, "a", "b"])

print(newList)

newList = extractintegers([1, "a", "b", 0, 15])

print(newList)

newList = extractintegers([1, 2, "aasf", "1", "123", 123])

print(newList)

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"

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

str1 = "Hello World"

# swap case of a string

str2 = str1.swapcase()

# Define a function reverse() to reverse the string

def reverse(s):

str = " "

for i in s:

str = i + str

return str

# Call the function with str2

str3 = reverse(str2)

# Print the string which is reversed and has opposite case

print(str3)

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.

# 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.

# Declare a sample list lst

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

# create 3 variables first, middle, last and assign values from the list lst

first = lst[0]

middle = list(lst[1:-1])

last = lst[-1]

# Print all the three variables

print(first)

print(middle)

print(last)

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

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

# Examples # factorial(5) ➞ 120

# factorial(3) ➞ 6#

# factorial(1) ➞ 1#

# factorial(0) ➞ 1

# Define a function to calculate the factorial of a number

def fact\_recursion(num):

if num == 1:

return num

else:

return num\*fact\_recursion(num-1)

# get the integer number from the user

n = int(input("Enter a positive integer number :"))

# use if condition to check the number

if n < 0:

print("factorial cannot be calculated for negative number")

elif n == 0:

print("factorial of 0 is 1")

elif n >= 1:

print("The factorial of a number ",n,"is",fact\_recursion(n))

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"]

# 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\_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"]

# Declare a variable to store the values

collection1 = (["a", "a", "a", "b"], "a")

# Declare list and variable to store values from the collection1 variable

list1 = collection1[0]

var1 = collection1[1]

# print the variable values to check if it stored the values correctly

print(list1)

print(var1)

# use for loop to iterate through the list list1

for i in list1:

if i == var1:

# remove the item and move it to the end of the list

list1.remove(i)

list1.append(i)

# print the new list

print(list1)