Lab: Lists Basics

Problems for in-class lab for the Python Fundamentals Course @SoftUni.
Submit your solutions in the SoftUni judge system at https://judge.softuni.bg/Contests/Practice/Index/1724

1. Strange Zoo

You are at the zoo and the **meerkats** look strange**.** You will receive **3 strings**: (tail, body, head). You have to **re-arrange** the elements in a list, so that the animal looks normal again: (head, body, tail)

Example

Input	Output
my tail my body seems on place my head is on the wrong end!	['my head is on the wrong end!','my body seems on place','my tail']
tail body head	['head', 'body', 'tail']

Hints

We start by reading the three parts of the body:

```
1    tail = input()
2    body = input()
3    head = input()
```

Then, we create a list containing those three:

```
4    meerkat = [tail, body, head]
```

We swap the elements and print the list

```
5    meerkat[0], meerkat[2] = meerkat[2], meerkat[0]
6    print(meerkat)
```

2. Courses

You will receive a single number **n**. On the next **n** lines you will receive **names** of courses. You have to create a **list of them and print it**

Example

Input	Output
2 PB Python PF Python	['PB Python', 'PF Python']

4 Front-End C# Web JS Core Programming Fundamentals	['Front-End', 'C# Web', 'JS Core', 'Programming Fundamentals']

Hints

We read the number **n** and we create an **empty list**

courses.py

```
n = int(input())
courses = []
```

We create a loop, read each course and add it in the list

```
for i in range(n):
    current_course = input()
    courses.append(current_course)
```

Finally, we print the list

```
print(courses)
```

3. List Statistics

You will be given a number **n**. On the next **n** lines you will receive integers. You have to create and print two lists:

- One with all the positives (including 0) numbers
- One with all the negatives numbers

Finally print the following message: **"Count of positives: {count_positives}. Sum of negatives: {sum_of_negatives}"**

Example

Input	Output
5 10 3 2 -15 -4	[10, 3, 2] [-15, -4] Count of positives: 3. Sum of negatives: -19

Hints

We start by reading the number n:

03-list-statistics.py

```
n = int(input())
positives = []
negatives = []
```

Then, we create **a loop**, we read the **current number** and check if it **is greater than zero or not**

```
for n in range(n):
    current_number = int(input())
    if current_number >= 0:
        positives.append(current_number)
    else:
        negatives.append(current_number)
```

- If it is, we add it to the positives list
- If it is not, we add it to the negatives list

Then we print the three lines:

```
print(positives)
print(negatives)
print(f"Count of positives: {len(positives)}. Sum of negatives: {sum(negatives)}")
```

- To get the count of the positives we can use the **len** function
- To get the sum of the negatives we can use the **sum** function

4. Search

You will receive a number **n** and a **word**. On the next **n lines** you will be given some **strings**. You have to **add** them in a **list and print** them. After that you have **to filter out** only the strings that **include** the given **word** and **print** that list too.

Example

Input	Output
3 SoftUni I study at SoftUni I walk to work I learn Python at SoftUni	['I study at SoftUni', 'I walk to work', 'I learn Python at SoftUni'] ['I study at SoftUni', 'I learn Python at SoftUni']
4 tomatoes I love tomatoes I can eat tomatoes forever I don't like apples Yesterday I ate two tomatoes	['I love tomatoes', 'I can eat tomatoes forever', "I don't like apples", 'Yesterday I ate two tomatoes'] ['I love tomatoes', 'I can eat tomatoes forever', 'Yesterday I ate two tomatoes']

Hints

We start by reading the number n, the word we will search for and create our empty list

04-search.py

```
1  n = int(input())
2  word = input()
3  strings = []
```

We create a loop and add all the strings in our list. After that, we print it

```
4  for i in range(n):
5      current_string = input()
6      strings.append(current_string)
7  print(strings)
```

Finally, we create another loop to remove the strings we don't need by iterating through the strings reversed (so we don't skip elements by removing) and print the list again

```
8  for i in range(len(strings) - 1, -1, -1):
9      element = strings[i]
10     if word not in element:
11         strings.remove(element)
12 print(strings)
```

5. Numbers Filter

You will receive a single number **n**. On the next **n** lines you will receive integers. After that you will be given one of the following commands:

- even
- odd
- negative
- positive

Filter all the numbers that fit in the category (0 counts as a positive and even). Finally, print the result.

Example

Input	Output
5 33 19 -2 18 998 even	[-2, 18, 998]
3 111 -4 0 negative	[-4]

Hints

First, we read the number n and we create the numbers list and the filtered list

05-numbers_filter.py

```
n = int(input())
numbers = []
filtered = []
```

We create a loop to read all the numbers and to add them to the list

```
for i in range(n):
    current_number = int(input())
    numbers.append(current_number)
```

Then we read the command and check for all the cases

```
if command == "even":
    for number in numbers:
        if number % 2 == 0:
            filtered.append(number)
elif command == "odd":
    for number in numbers:
        if number % 2 != 0:
            filtered.append(number)
elif command == "negative":
    for number in numbers:
        if number < 0:
            filtered.append(number)
elif command == "positive":
    for number in numbers:
        if number >= 0:
            filtered.append(number)
print(filtered)
```

- Finally, we print the filtered list