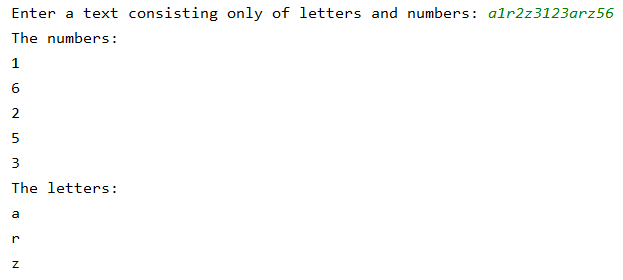
**Exercises Set and Dictionary**

**Exercise 1**

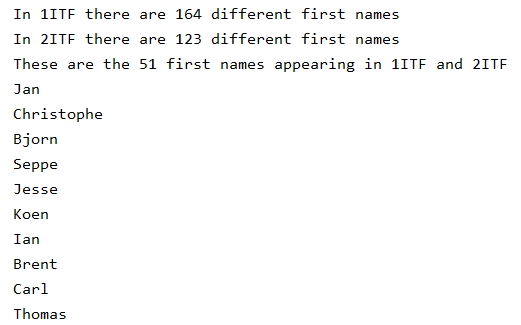
Write a program that asks for a string consisting of letters and numbers. When you press Enter, the program will first display all numbers, and afterwards all letters. All duplicates are removed.



**Exercise 2**

For this exercise you use the files *first\_names1ITF.txt* and *first\_names2ITF.txt*.

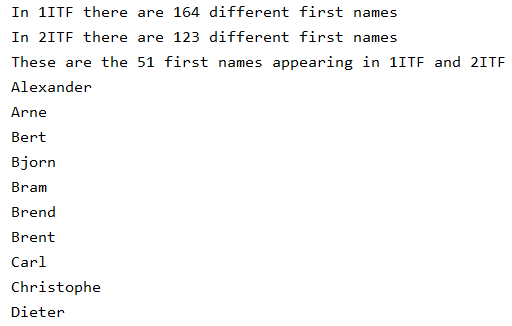
1. Write a program that generates the following output.



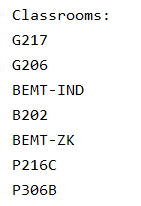
….

Take note: the order of the first names will be different every time you run this program.

1. Now adjust your program so that the list of first names is printed alphabetically.



….

**Exercise 3**

For this exercise use the file *local\_booking.txt.*

Write a program that prints all the classrooms that are used for the lessons in the input file.

Remark: if you run your program, the printed result might be in a different order

**Exercise 4**

In this exercise you work with the files *games.txt* and *games.xml.*

In both files you will find information about board games. What interests us is how many types of games there are.

Read both files and generate the screenshot below.

In the txt-file: 19 types of games

In the xml-file: 4 types of games

The types that occur in both files:

{'educational', 'financial', 'strategic'}

The types that only appear in the txt-file:

{'horror', 'skills', 'running game', 'language', 'toddlers', 'puzzle', 'historical', 'cards', 'tactical', 'simulation',

'construction skills', 'dice', 'company', 'sport', 'memory', 'combination'}

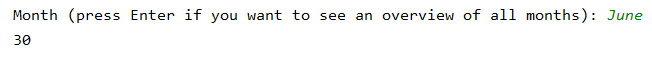
The types that only appear in the xml-file:

{'gambling'}

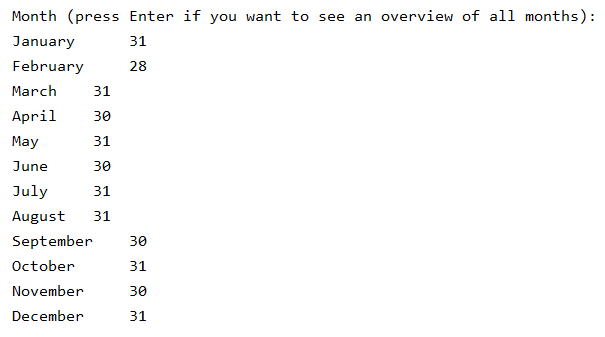
**Exercise 5**

Create a Dictionary with the months and the number of days in that month.

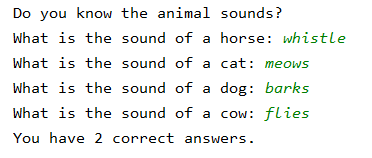
Write a program to show the number of days of a chosen month. If the user presses Enter immediately on the first question, the complete overview is printed.







**Exercise 6**

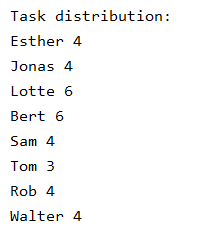
1. Create a Dictionary with animals and their accompanying sound.

Use this Dictionary to let a student practice with the animal sounds as shown in the screenshot below. (green text = input from the student). The number of correct answers is printed at the bottom. Since the sound of the cow and the horse is wrong, the student scored 2 correct answers.

1. Expand the exercise so that the program reads the animals and the accompanying sound from a file. (you have to create this txt-file yourself)

**Exercise 7**

For this exercise you will use the file *tasks.csv* that contains a table showing who performs a specific task at a certain time. Write a program that allows you to check how many tasks everyone is supposed to do.

The output is as follows: