Hi! In this video, that has been divided in two parts, you will learn about 4 additional aspects you will need to work on the first assessment of the module.

[SLIDE 2] They are: how to add comments in Python

[SLIDE 3] how to work with lists

[SLIDE 4] how to control the execution of the programme, and

[SLIDE 5] how to enter arguments using the command line.

[SLIDE 6] Let's start with comments. To comment a single line in Python you must use the symbol hashtag.

[SLIDE 7] You can comment a complete line, either made of text or code, by placing the hashtag at the beginning of it, as shown in lines 1 and 6 of this example code.

[SLIDE 8] You can also comment a part of a line by inserting a hashtag (#) in the position where you want your comment to start, as done in line 3 of this example code.

[SLIDE 9] If you need to comment several lines, you can either write a hashgtag at the beginning of each line or

[SLIDE 10] write 3 quote symbols at the start and end of the block of lines you would like to comment.

[SLIDE 11] Ok, that's all for comments. Let's now move to a brief review of lists.

[SLIDE 12] Do you remember when you worked with linked lists in C?

[SLIDE 13] They were allocated one element of memory at a time and because of that, they were dynamic. That is, you could add and remove elements in a way that you could not do with arrays.

[SLIDE 14] Well, lists in Python behave exactly like that. You can add and remove elements as you wish.

[SLIDE 15] But, unlike linked lists in C, lists in Python are easier to use because the elements can be accessed by using indices, as arrays.

[SLIDE 16] And, unlike arrays and linked lists in C, they can have elements of different types in each position.

[SLIDE 17] We will study lists in details during week 3, but in the next slides I will show you the basics you need to know to get started.

[SLIDE 18] To create a list, you simply

[SLIDE 19] give it a name and then

[SLIDE 20] assign the list of elements you want to it.

[SLIDE 21] To print all the elements of a list you don't need any for loop, the function print receiving the name of the list as an input argument is enough.

[SLIDE 22] To access a specific element in a list, you use its index, in the same way you do it with arrays in C.

[SLIDE 23] To add an element to the list, you use the function insert as shown here. The first input argument is the index where you want the new value to be located and the second input argument is the value you want to insert. You can see in the comment next to this instruction the new elements in the list after the insertion operation. Much easier than when you worked with linked lists in C!

[SLIDE 24] To remove an element, you simply use the function remove and enter as input argument the value of the element you want to remove.

[SLIDE 25] Finally, if you want to visit every element of the list, you use the “for in” loop, that traverse the list starting from the element in index 0. In each iteration, the value of the next element visited is stored in variable x.

[SLIDE 26] With this I finish this brief review of lists. Please, remember that we will study them in much more detail during Week 3.

Let’s move now to the control of programme execution, that will be explained in 3 parts in the second part of this video.