Introduction à la programmation, lesson 2: Input / Output

Most programs interact with a user. A program might need to say something to a user, that is, print something on a screen (output). To this end, we can use the function print that we saw last week. Also, the user might need to say something to the program (input). To this end, we use the function input():

```
variable = input() #the program stops and waits for a user to type something on the keyboard
print variable
```

Let us consider a more interesting example. A program will ask a user their name, and then will print it out.

```
print "What is your name? "
var_name = input()
print "Your name is: ", var_name
```

When Python executes the function input(), it

- stops the execution of the program
- shows a message on the screen
- waits for the user to type something and to press enter
- restarts the execution

NB! In Python 3 the message that the user sends to the program transforms into a string (note that an integer can also be considered as a string). For example, the following code will give an error:

```
print("I will show you that I can multiply by two.")
var = input("Enter an integer: ")
var_double = var * 2
```

We can fix this error by converting the input into an integer using the function **int()** :

```
print("I will show you that I can multiply by two.")
var = int(input("Enter an integer"))
var_double = int(var) * 2
```

If the input is a float, you can use the function float().

You can check the type of a variable using function **type()** .

```
var_1 = "tata yoyo"
print type(var_1)
var_2 = 3
var_3 = 3.0
var_4 = input("Enter an integer: ")
var_5 = int(var_4)
var_6 = float(var_4)
```

Exercise 0. What are the types of the variables var_2, var_3, var_4, var_5 and var_6 in the example above? Check your answer using the function type().

Exercise 1. Write a program that asks a user their name, their age, and then prints “Next year [name] will be [age] years old”.

New type: Booleans

We have already seen three types of variables: strings, integers, floats. We will now introduce Booleans. A Boolean variable can take only two values: True and False. A Boolean expression is an expression that has value True or False. Here are some examples of Boolean expressions that use operators == (“is equal to”), != (“not equal to”), >, <, <= (“less or equal”), >= (“greater or equal”).

```
print 3 == 3
print 3 == 5
print 3 + 2 == 5
print 3 != 3
print 3 != 4
print 3 > 4
print 3 <= 4
print 3 <= 3
```

We can of course define a variable of Boolean type:

```
var_bool_1 = True
var_bool_2 = False
print var_bool_1, var_bool_2
```

Conditional expressions : if / else

1. if statement

```
if <condition>:
    #Four indented spaces!
    Steps to be executed if the condition is True
```

Try the following program first with a positive number, and then with a negative.

```
var = input("Enter an integer: ")
if var > 0:
    print "The integer is positive"
```

Try to enter a float. What happens? How to fix it?

2. if … else statement

```
if <condition>:
      Steps to be executed if the condition is True
else:
      Steps to be executed if the condition is False
```

Note that else is not followed by a condition. For example:

```
var = input("Enter an integer: ")
if var > 0:
    print "The integer is positive"
else:
    print "The integer is negative or zero"
```

3. elif statement

```
var_age = input("Your age: ")
if var_age < 0:
        print "You lie!"
```

```
elif var_age < 3:
        print "You are a baby."
elif var_age < 12:
      print "Go play outside!"
else:
        print "You have reasonable age."
```

Exercise 2. Write a program that asks a user to enter two numbers a, b, and then prints out “The sum equals a+b” if a+b is smaller than 100 and otherwise prints out “The sum is too large”.

Boolean operators

The or operator

(expr_1 **or** expr_2) equals True if at least one of the two expressions expr_1, expr_2 is equal to True. It equals False if both expr_1 and expr_2 are equal to False.

```
var_age = input("What is your age? ")
if var_age <= 26 or var_age >= 60:
      print "You have a discount"
else:
      print "You pay the full price"
```

Question: for which value of var_age the expression (var_age <= 26 ou var_age >= 60) is equal to True?

The «and» operator

(expr_1 **and** expr_2) is equal to True if both expr_1 and expr_2 are equal to True.

```
var_age = input("What is your age? ")
if var_age > 26 and var_age < 60:
      print "You pay the full price"
else:
      print "You have a discount"
```

Question: for which value of var_age the expression (var_age <= 26 and var_age >= 60) is equal to True?

Some more exercises

Exercise 1. Write a program that asks a user to enter two integers, then prints out their sum and their difference. For example:

```
First integer: 3
Second integer: 5
Sum: 8
Difference: -2
```

Exercise 2. Write a program that asks a user to enter three integers, and outputs their maximum.

Exercise 3. Write a program that asks a user to give a sequence of students as a string (e.g. “Alvaro Arenas, Roberto Bruno, Alexis Krauthgamer”), and creates a list of all students, a list of students whose family names start with letters A-M, and a list of students whose family names start with letters N-Z, and prints out the three lists.