The “if” statement in Python

Nick Szirbik

Sept. 2020

Week 2 of the DAPOM course (4)

The general Python **if-else** syntax

`if` *<condition>* **`:`**

<indented Statement Block For True Condition>

[**`else:`**

<indented Statement Block For False Condition>]

- The statement blocks can contain one or more statements, or **`pass`**
- The else “branch” is optional
- The condition is a Boolean constant (**`True`**, **`False`**) , a Boolean variable, or a Boolean expression

Boolean expressions

```
2 < 5      # obviously True
3 > 7           # patently False
x == 11    # True only if x got the value 11 before this
check
x > 10     # depends on the value of variable x
2 * x < x # cannot be True, no?
#
```


Meaning	Math Symbol	Python Symbols
Less than	<	<
Greater than	>	>
Less than or equal	≤	<=
Greater than or equal	≥	>=
Equals	=	==
Not equal	≠	!=

More Boolean operators: Logical and Membership operators

- Logical operators are used to combine conditional statements:

Operator	Description	Example
and	Returns True if both statements are true	x < 5 and x < 10
or	Returns True if one of the statements is true	x < 5 or x < 4
not	Reverse the result, returns False if the result is true	not(x < 5 and x < 10)

- Membership operators are used to test if an item or sequence of items is presented in an collection (list, set, dictionary, tuple):

Operator	Description	Example
in	Returns True if a sequence with the specified value is present in the object	x in y
not in	Returns True if a sequence with the specified value is not present in the object	x not in y

Nested **ifs** and the use of **elif** (to make it nicer looking)

```
# nasty looking code, without elif

def letterGrade(score):
    if score >= 90:
        letter = 'A'
    else:   # grade must be B, C, D or F
        if score >= 80:
            letter = 'B'
        else:  # grade must be C, D or F
            if score >= 70:
                letter = 'C'
            else:    # grade must D or F
                if score >= 60:
                    letter = 'D'
                else:
                    letter = 'F'
    return letter
```



```
# nicer looking code with elif

def letterGrade(score):
    if score >= 90:
        letter = 'A'
    elif score >= 80:
        letter = 'B'
    elif score >= 70:
        letter = 'C'
    elif score >= 60:
        letter = 'D'
    else:
        letter = 'F'
    return letter
```

Finally: Importing, or getting access to existing code in Python

- We can **import** (basically is an automatic copy/paste from another chunk of code) various other programs that we have access to

- These are called in our **jargon** as: modules, packages, or libraries (in the old jargon: "subroutines")

In the next practical, you will need some imports, for example:

```
import csv # importing a whole library, or
from csv import reader # importing only a single function
```

If you have questions

- Please prepare these on paper, and ask them during the tutorial

- You can ask in the break-out room of your team

- Or you can ask during the plenary openings

- If urgent, send an email to me (n.b.szirbik@rug.nl), but keep it short. If I have time, I try to respond immediately…