



Flow Control , Functions & Objects:

Loops & Conditional Execution.

+ Functions

+ Objects

Session 15

Programación Estadística con Python

Alberto Sanz, Ph.D

asanz@edem.es

MASTER EN DATA ANALYTICS PARA LA EMPRESA

- Looping
 - For loops
 - While loops

- Conditional execution
 - If / else :

- User Defined Functions
 - def
 - (Anonymous / Lambda functions)

Iteration: *for* loops I

- When we want to repeat a process several times:
 - Iteration
 - Repetition of a process with **control** over the **number (or order)** of current repetition
 - **for** loops:

```
# Basic loop
for i in range(0,11,1):
    print("i:",i)
```



```
i: 0
i: 1
i: 2
i: 3
i: 4
i: 5
i: 6
i: 7
i: 8
i: 9
i: 10
```

Iteration: *for* loops I

- When we want to repeat a process several times:
 - ▣ Iteration
 - Repetition of a process with **control** over the **number (or order)** of current repetition
 - ▣ *for* loops:

```
# Basic loop
for i in range(0,11,1):
    print("i:",i)
```

Indentation
matters!!

From

Up to

Step



```
i: 0
i: 1
i: 2
i: 3
i: 4
i: 5
i: 6
i: 7
i: 8
i: 9
i: 10
```

Iteration: *for* loops II

5

```
# Basic loop 2  
for i in [1,2,3,4]:  
    print("i:",i)
```



```
i: 1  
i: 2  
i: 3  
i: 4
```

```
# Basic loop 3  
for i in ["red", "blue", "yellow"]:  
    print(i)
```



```
Red  
blue  
yellow
```

Tip: The execution flow can be forced to escape a loop with a *conditional break*

Iteration: *for* loops EXAMPLE

- Basic example in analytics:
Chosing the optimal number of bins in specific histogram.
- ▣ Create a series of histograms with increasing number of bins.

```
for i in range(1, 51, 1):  
    x=wbr['cnt']  
    plt.hist(x, bins=i)  
    plt.show()
```

Conditional Iteration : *while* loops

- When we want to repeat a process while some condition holds:
 - *while* loops:

```
#### While loops
count=1                                # Initianlize counter
while (count <4) :                      # Control
    print (count, "Calidad")           # Action
    count=count+1                      # Counter increase
```



```
1 Calidad
2 Calidad
3 Calidad
```

Tip: You can further explore *while* loops at: <https://www.geeksforgeeks.org/loops-in-python/>

Conditional execution: *If* / *else* (I)

- Execute part of your code conditional to some variable:
 - ▣ *If* / *else* statements

```
for i in range(0,11,1):  
    if i < 5:  
        grade="Fail"  
    else:  
        grade="Pass"  
    print(" My grade is ", i,":", grade)
```


Conditional execution: *If* / *else* (I)

- Execute part of your code conditional to some variable:
 - ▣ *If* / *else* statements

```
for i in range(0,11,1):  
    if i < 5:  
        grade="Fail"  
    else:  
        grade="Pass"  
    print(" My grade is ", i,":", grade)
```



```
My grade is 0 : Fail  
My grade is 1 : Fail  
My grade is 2 : Fail  
My grade is 3 : Fail  
My grade is 4 : Fail  
My grade is 5 : Pass  
My grade is 6 : Pass  
My grade is 7 : Pass  
My grade is 8 : Pass  
My grade is 9 : Pass  
My grade is 10 : Pass
```

Conditional execution: *If* / *else* (I)

10

- Execute part of your code conditional to some variable:
 - ▣ *If* / *else* statements

```
for i in range(0,11,1):  
    if i < 5:  
        grade="Fail"  
    else:  
        grade="Pass"  
    print(" My grade is ", i,":", grade)
```



```
My grade is 0 : Fail  
My grade is 1 : Fail  
My grade is 2 : Fail  
My grade is 3 : Fail  
My grade is 4 : Fail  
My grade is 5 : Pass  
My grade is 6 : Pass  
My grade is 7 : Pass  
My grade is 8 : Pass  
My grade is 9 : Pass  
My grade is 10 : Pass
```

First
Indentation
for loop
action

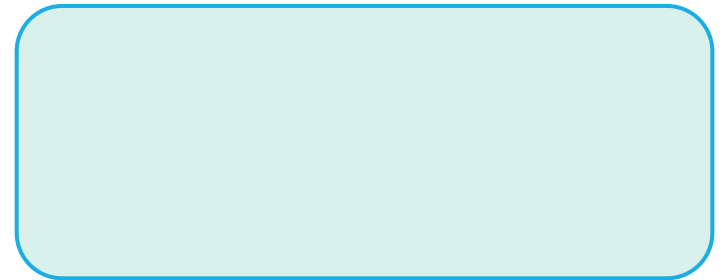
Second
Indentation
for conditional
action

Functions in python(I)

11

- We can define our own functions in Python!
 - ▣ Can be used also as “encapsulated routines” that make our code cleaner

```
# Define a function `plus()`  
def plus(a,b):  
    print("let's add these two numbers")  
    return a + b
```



```
plus (2,3)
```

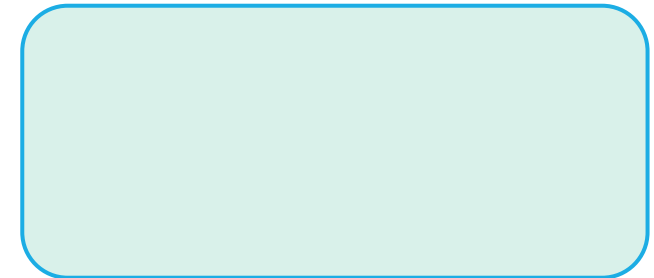


```
let's add these two numbers  
5
```

Functions in python(II)

12

```
def divide (a,b):  
    """Explain here what my function is doing  
    I can use several lines if triple quotation"""  
    print("let's divide these two numbers...")  
    return a / b
```



```
divide (3,2)
```



```
Let's divide these two numbers...  
1.5
```

```
divide (2,3)
```



```
Let's divide these two numbers...  
0.6666666666666666
```

```
divide (b=2,a=3)
```



```
Let's divide these two numbers...  
1.5
```

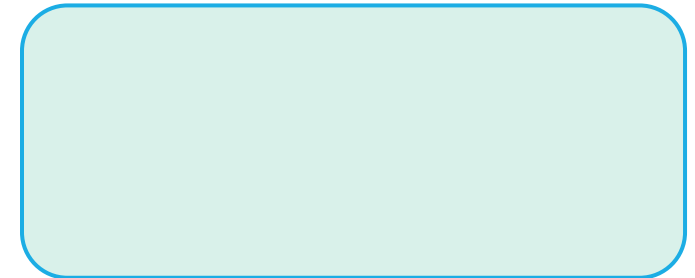
Functions in python(III)

13

- We can define *default* values for some arguments in a function

```
# Define a function `plus()`  
def plus(a, b=2):  
    print("let's add these two numbers")  
    return a + b
```

default value



```
plus (2, 3)
```



```
let's add these two numbers  
5
```

```
plus (2)
```

Argument b is absent, then it
takes the default value b=2



```
let's add these two numbers  
4
```

Tip: You can explore anonymous `lambda` functions at:
<https://www.programiz.com/python-programming/anonymous-function>

Objects in python(I)

Objects in python(I)

■ Data analysis perspective:

- Objects are data containers that can self analyse themselves
- Objects have
 - Attributes: (Data)
 - Methods: (Functions)

```
#Extract an attribute (age) from a Pandas  
#Dataframe Object (my data)  
print (my_data.age) #Age is an attribute
```



Out: [23,34,45]

```
#Execute a Method(count) on a Pandas  
#Dataframe attribute (age) which is part of an #  
Object (my data)  
print (my_data.age.count()) #count is a method  
print (my_data.age.sum()) #sum is a method
```



Out: 3

Out: 102

Objects in python(II)

16

- We can **define** our own **classes** of objects in Python!

```
#Define the class of objects  
class lista_pro:  
    datos = []  
    def media(self):  
        print("The average is:...." )  
        return sum(self.datos)/len(self.datos)
```



Out:

- We can **create objects** of a specific **class**

```
#Create an object from a specific class  
edad= lista_pro()
```



Out:

Out:

- We can use **objects**!

```
#Use the object !  
edad.datos=[2,3,4] # Fill with data  
print(edad.media()) # Apply the method "media"  
print(edad.datos)   # Extract the attribute "datos"
```



Out: 3

Out: [2,3,4]

Questions?

Thank you !

Alberto Sanz
asanz@edem.es