

# Fundamentos de programación

## Sesión 03

Ing. Gómez Marín, Jaime<sup>1</sup>

<sup>1</sup>Fundamentos de Python para Ciencia de Datos  
Departamento de TdG

August 2019



- Introducción
- Condicionales
- Bucles
- Funciones
- Conclusiones
- Bibliografía



En esta sesión se aprenderá a usar los conceptos básicos de la programación estructural con Python.



Python maneja estructuras condicionales que nos permiten evaluar expresiones ógicas y poder realizar acciones dependiendo de sus resultados, las condicionales que usa Python son:

- if ..
- if .. else ..
- if .. elif .. else ..



# Diagrama de flujo de : if ..

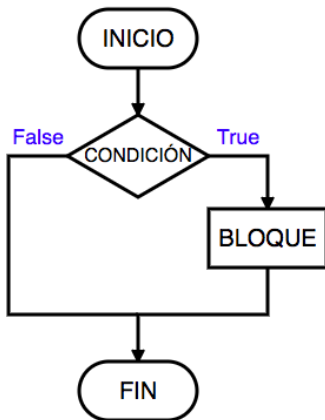


Figure: 01 : if

```
26 print("-----")
27
28 pbipc = 20567
29
30 # expression that can be true or false
31 if pbipc > 17971:
32     print("you HAVE pbi per capital over "
33           "the average world")
34
35 print("Thank you")
```

# Diagrama de flujo de : if .. else ..

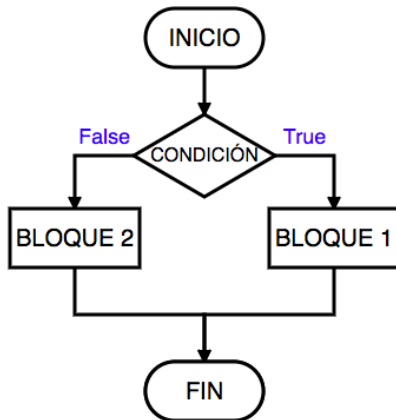


Figure: 02 : if .. else ..

## Code - if .. else ..

```
38 print("-----")
39 pbipc = 20567
40
41 # expression that can be true or false
42 if pbipc > 17971:
43     # within an indent, we have the expression
44     # that is run if the condition is true
45     print("you HAVE pbi per capital over the "
46           "average world")
47 else:
48     print("you DONT HAVE pbi per capital over "
49           "the average world")
50
51 print("Thank you")
```





# Diagrama de flujo de : if .. elif .. else ..

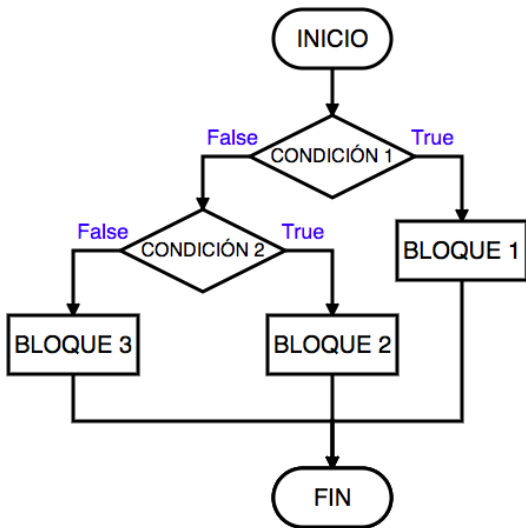


Figure: 03 : if .. elif .. else ..

## Code - if .. elif .. else ..

```
53 print("-----")
54 pbipc = 14393
55
56 # expression that can be true or false
57 if pbipc > 17971:
58     print("you HAVE pbi per capital upper the "
59           "average world")
60 elif pbipc == 17971:
61     print("you HAVE pbi per capital equals the "
62           "average world")
63 else:
64     print("you DONT HAVE pbi per capital less "
65           "the average world")
66
67 print("Thank you")
```



Python maneja como todo lenguaje de programación el control de bucles. Principalmente tiene 2 tipo de bucles :

- bucle for
- bucle while



```
30 print("-----")
31 pbipc_list = [14393, 14999, 16068, 25530, 25283]
32 n = len(pbipc_list)
33 print(n)
34
35 for i in range(n):
36     print(pbipc_list[i])
```

```
38 print("-----")
39 pbipc_list = [14393, 14999, 16068, 25530, 25283]
40
41 for pbipc in pbipc_list:
42     print(pbipc)
```

```
81 | dates = [2010, 2011, 2012, 2013]
82 |
83 | i = 0;
84 | year = 0
85 | while (year != 2012):
86 |     year = dates[i]
87 |     i = i + 1
88 |     print(year)
89 |
90 | print("it took ", i, "repetitions to get out of loop")
```

Se pueden definir funciones como un fragmento de código repetido que debe cumplir un propósito de ser rehusado en un programa.

- Los bloques de función comienzan con la palabra clave `def` , seguido por el nombre de la función y los paréntesis ( ).
- Cualquier parámetro de entrada o argumentos deben ser colocados dentro de estos paréntesis. También puede definir los parámetros dentro de los paréntesis.
- El bloque de código dentro de cada función se inicia con dos puntos (:) y es con sangría.
- La sentencia `return [expresión]` sale de una función y es opcional



# Code - Function

```
1  '''
2  Programa : Ciencia de Datos con Pthon
3  Modulo 01 : Fundamentos de Python para Ciencia de Da
4  Sesion 03 : Fundamentos de programación - Function
5  Fecha : 21/07/2019
6  Version : 1
7  Author :
8  '''
9
10 def inc(a):
11     '''
12     inc 1 to a
13     '''
14     b = a + 1
15     return(b)
16
17 ret = inc(1)
18 print(ret)
```





En esta sesión se ha visto el uso de condicionales, bucles y funciones dentro del entorno de Python





Naomi Ceder. The Quick Python Book - Manning Publications, 2018.