

# Fundamentos Python 2 — Estructuras de control y bucles

Las herramientas del científico de  
datos

Juan Manuel Moreno — [jmmoreno@profesorescol.imf.com](mailto:jmmoreno@profesorescol.imf.com)



# ÍNDICE

1. Objetivos unidad 1
2. Condicionales
3. Bucles



# 01

# Objetivos unidad 1

# 1.– Objetivos Unidad 1

- Conocer los principales fundamentos de Python.
- Saber instalar Jupyter Notebook, la herramienta que vamos a utilizar para trabajar con Python.
- Realizar desarrollos básicos en Python a través de Jupyter Notebook
- Saber cómo declarar, procesar y distinguir distintos tipos de variables.
- Conocer y manejar las sentencias condicionales If – Else.
- Conocer el funcionamiento de los bucles en Python, for y while.
- Trabajar con las principales estructuras de datos en Python: Tuplas, listas y diccionarios de datos.
- Comprender cómo modularizar los programas a través de funciones.
- Resolver problemas de diferente dificultad.



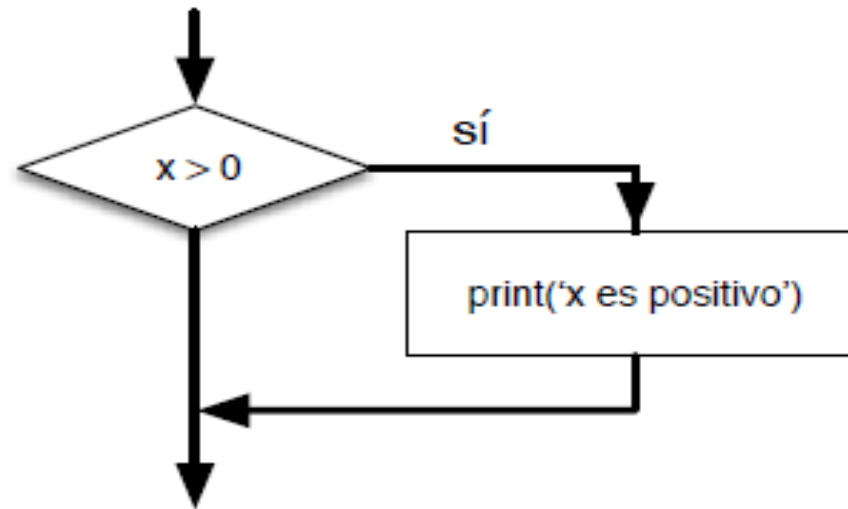
# 02

# Condicionales

## 2.- Condicionales

### 2.1.- Condicionales

- Utilizan operadores booleanos True o False.
- Se construye una expresión booleana (if), si se cumple esta instrucción se realiza una operación, en el caso contrario, si no (else), se realiza otra operación.
- El tipo más básico de condicional es el if simple

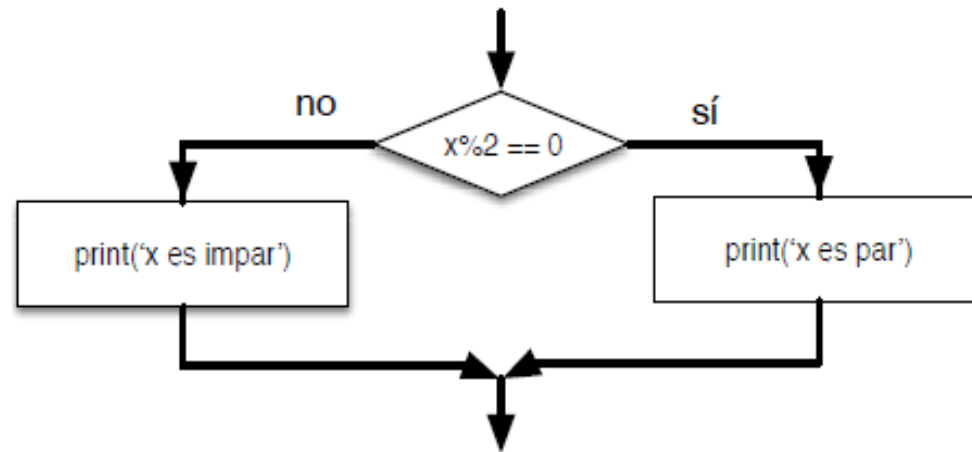




## 2.- Condicionales

### 2.2.- If simple

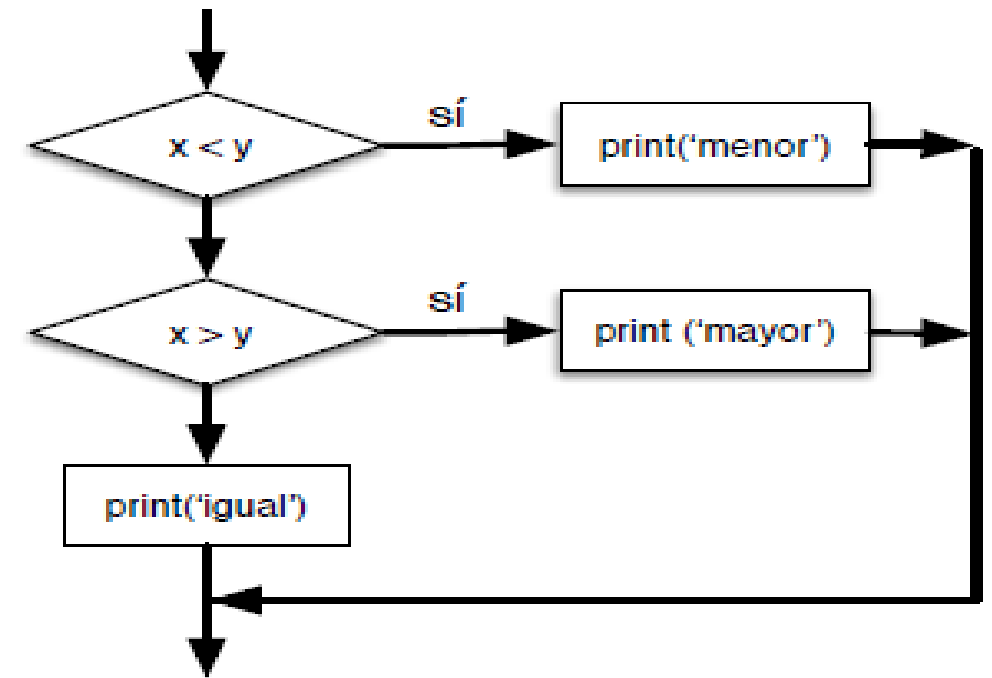
- En algunas ocasiones, en la parte del else, puede ser que no hagamos nada ya que solamente queremos controlar una orden lógica.
- En otras ocasiones en el si no, else, se incluirá otra instrucción diferente



## 2.- Condicionales

### 2.3.- If encadenado

- Otro tipo de condicional es el if encadenado,
- Si la condición booleana no se cumple, se abre una nueva expresión booleana con otro if (toma forma de elif).
- Pueden abrirse n nuevos elif si es necesario, se finaliza con un else.
- if —elif — else

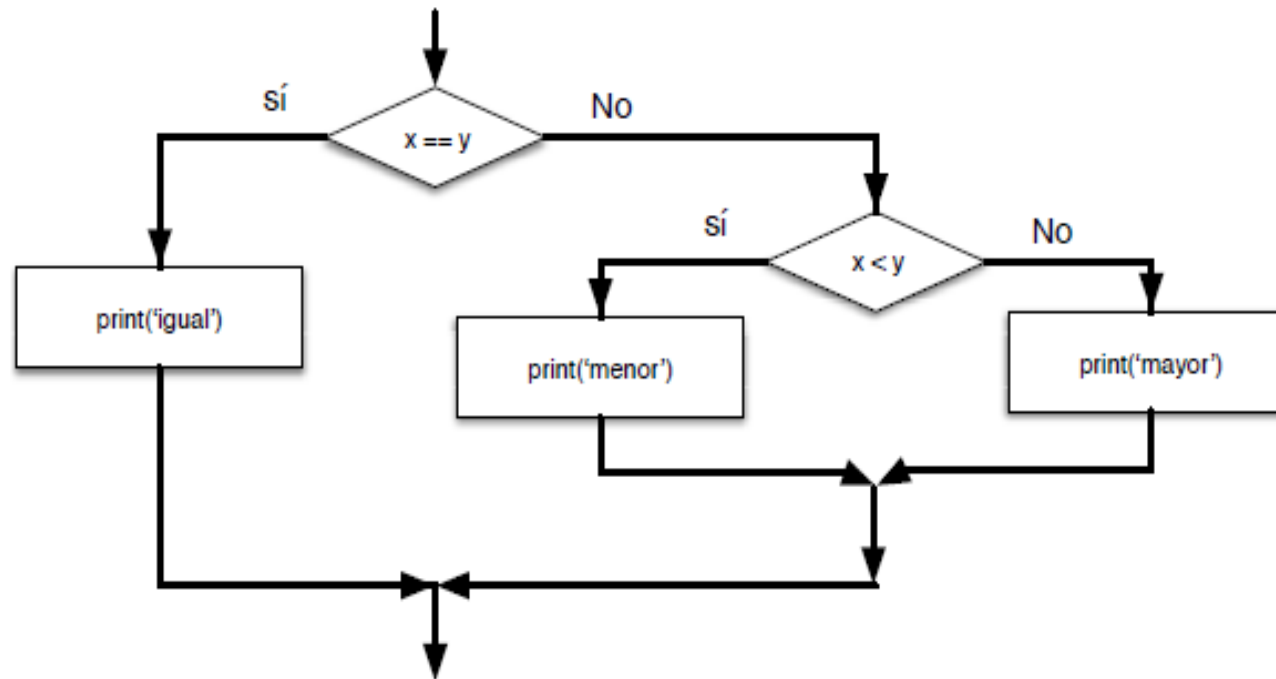




## 2.- Condicionales

### 2.4.- If anidado

- El último tipo de condicional que nos vamos a encontrar en Python será el condicional anidado.
- Se incluye un nuevo condicional dentro de la estructura del propio condicional, pudiendo ser tanto en el if como en el else





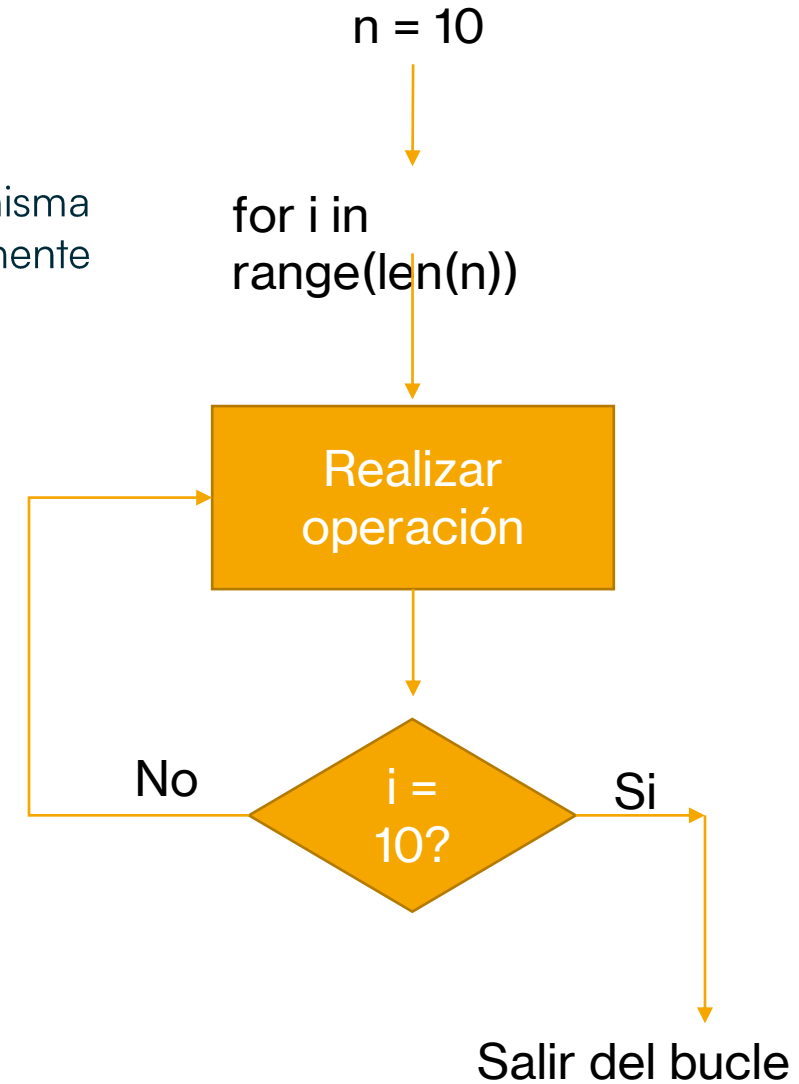
# 03

# Bucles

## 3.- Bucles

### 3.1.- Bucle For

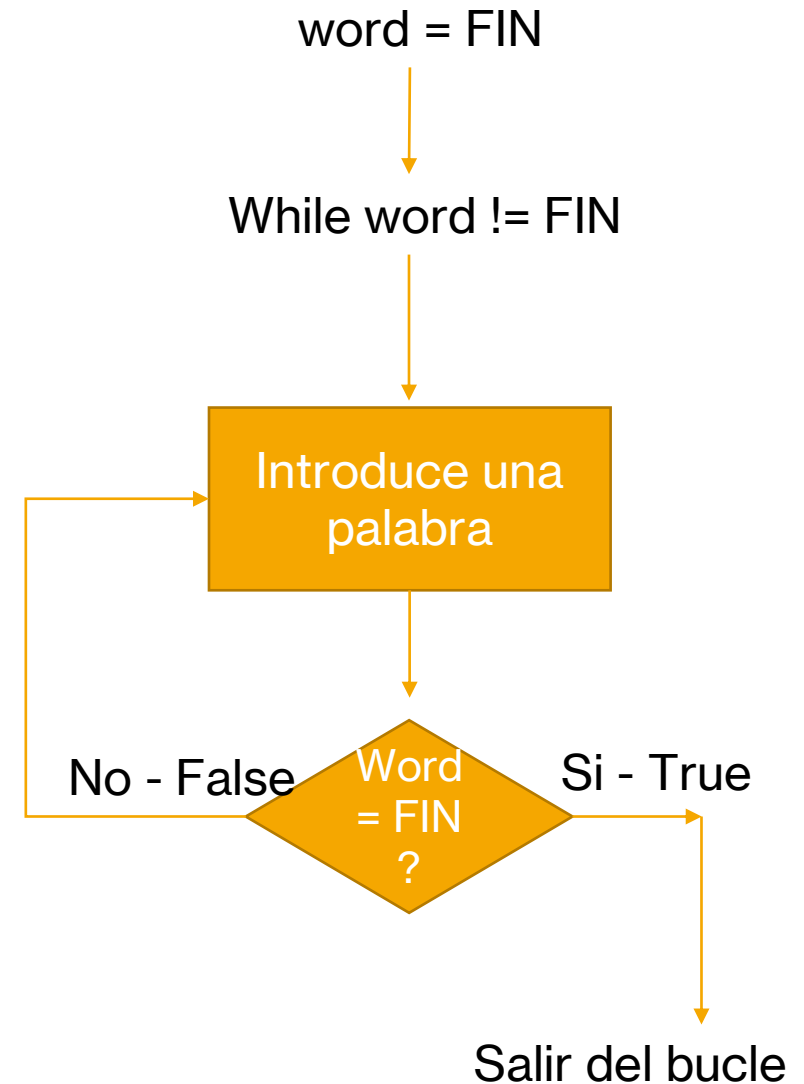
- Algo muy habitual cuando estamos desarrollando es tener que realizar una misma operación varias veces, para ello existen los bucles, en donde podemos iterativamente realizar una operación.
- El bucle **for**:
  - Secuencial, se realiza la tarea tantas veces como el tamaño de la secuencia.
  - Se utiliza una variable a modo de índice para ir recorriendo la secuencia.



## 3.- Bucles

### 3.2.- Bucle While

- Se realiza una misma acción hasta que se cumple una instrucción de parada.
- Esta instrucción de parada es una expresión booleana.
- Por ejemplo, mientras (while), la expresión no devuelve False, realiza el proceso

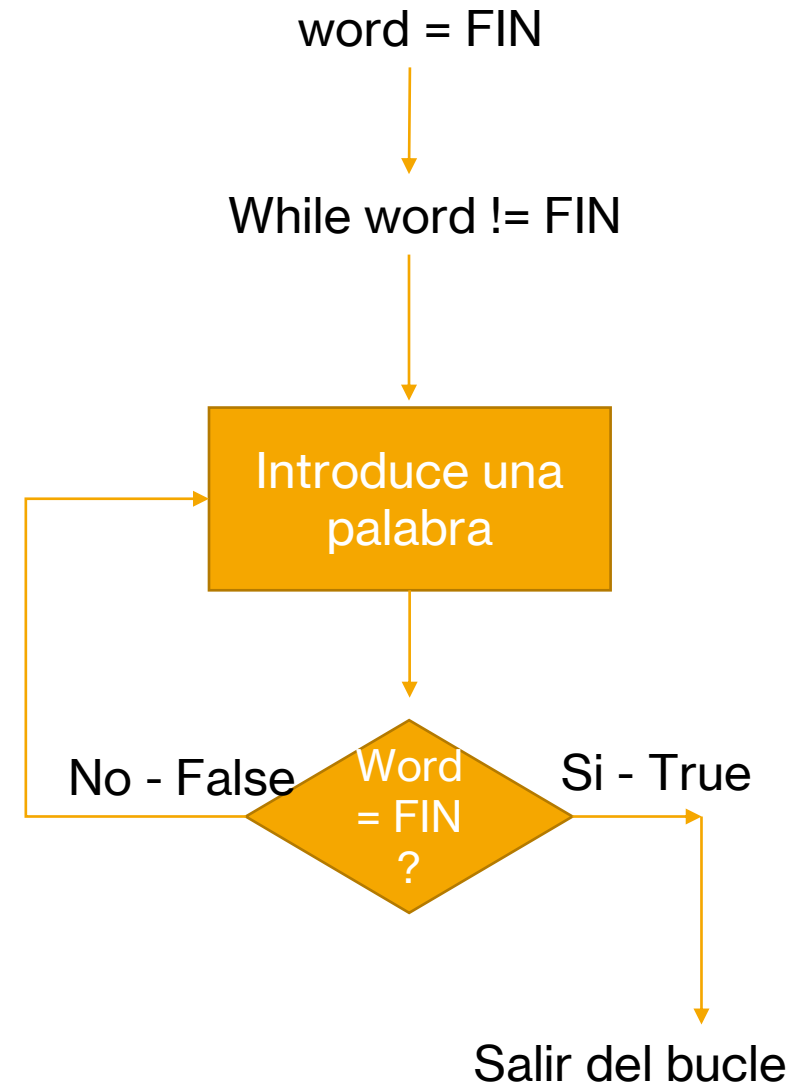




## 3.- Bucles

### 3.2.- Bucle While

- Se realiza una misma acción hasta que se cumple una instrucción de parada.
- Esta instrucción de parada es una expresión booleana.
- Por ejemplo, mientras (while), la expresión no devuelve False, realiza el proceso



# Seguimiento práctico del contenido

A partir de aquí, vamos a ver cómo funcionan los condicionales y los bucles for y while a través del siguiente notebook.

## 1\_3\_Herramientas\_control\_flujo.ipynb

IMF

Smart Education