



# Python para principiantes



Capítulo 1: Instalación y Entorno  
Thonny. Generalidades



# Características

- Python es un lenguaje de programación interpretado
- Se encuentra instalado en muchos sistemas operativos
- Lenguaje de software libre y está disponible de manera gratuita para todos los sistemas operativos.

<http://www.python.org>



# Realizar un programa

---

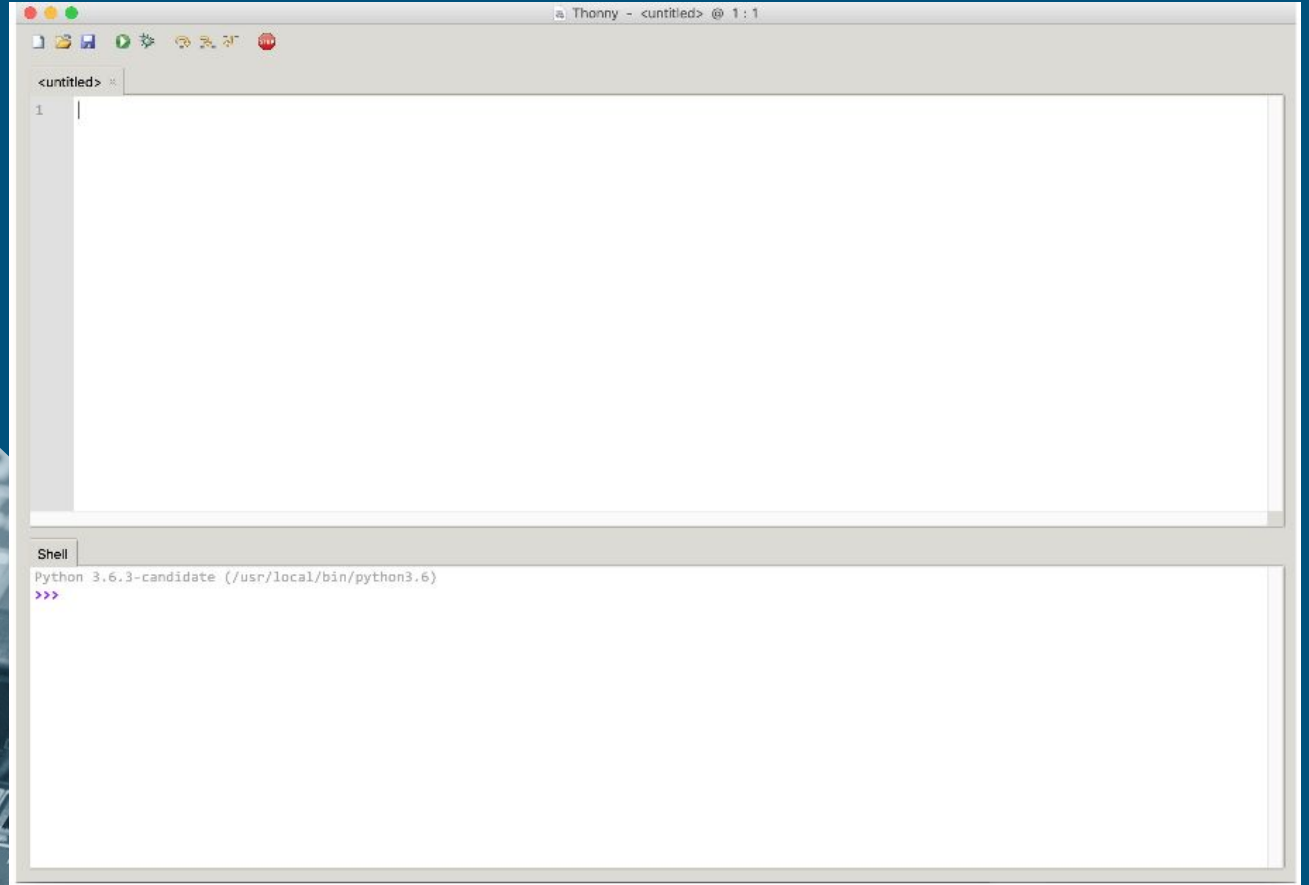
- Sólo necesitamos un editor de textos (Bloc de Notas, gEdit, etc...)
  - el intérprete que será el que haga que las instrucciones escritas en el editor de textos se ejecuten
- Hay muchos (programas) entornos de desarrollo integrados (IDE's) de gran calidad para desarrollar aplicaciones de manera profesional.
  - usar el entorno de programación Thonny,
    - Diseñado y pensado para las personas que desean iniciarse en Python.

# Instalación Thonny

---

- 01 Acceder la web oficial de Thonny: <https://thonny.org/>
- 02 Copiar la instrucción:  
`bash <(wget -O - https://thonny.org/installer-for-linux)`
- 03 Abrir un terminal(ctrl+alt+t), y pegar la instrucción(ctrl+shift+v), ENTER

# Thonny



# Entorno Thonny

---

El programa inicialmente divide la ventana en 2 partes bien diferenciadas:

## El editor de código

- Como el Bloc de Notas, gEdit..
- El contenido como texto plano, sin ningún tipo de formato.
- Nos resaltará la sintaxis y nos ayudará con la escritura de instrucciones: tabulador o ctrl+espacio
- Guardar el trabajo en archivos que deben tener la extensión .py
- Pulsar el botón Play (o pulsar F5) y nuestro programa se ejecutará
- La posibilidad de ejecutar el programa paso a paso e ir comprobando en cada momento el resultado de cada una de las operaciones

# Entorno Thonny

---

## El intérprete de Python

- Se trata de una línea de comando donde podemos introducir instrucciones y comprobar el resultado que produce su ejecución.
- Es ideal para hacer pruebas de pequeñas porciones de código y para aprender el funcionamiento de determinadas instrucciones.
- Dispone de una ayuda, ejecutar con la función `help()`
- En el nuevo prompt podemos escribir cualquiera de las palabras reservadas de Python y obtener la ayuda que está disponible
- Para salir de modo `help>`, podemos dar la instrucción **quit** y volver al intérprete

```
>>> help()  
help>
```

Observa que al ejecutar el función `help()`, se obtiene un nuevo prompt en el intérprete.

El aspecto de la ayuda es: `help>`

# Palabras reservadas de Python (palabras claves):

---

and	as	assert	break	class
continue	def	del	elif	else
except	finally	for	from	global
if	import	in	is	lambda
nonlocal	not	or	pass	raise
return	try	while	with	yield
False	None	True		



# Generalidades

---

## Sensible a mayúsculas y minúsculas

Ejemplo: For es diferente de for,

## Bloques de código

Las instrucciones que definen el comienzo del bloque de código, deben terminar en el carácter de los 2 puntos (:)

## Indentación

Es la forma que tiene Python de realizar bloques de código. Se llama así una especie de sangría antes del texto que se establece para clarificar la escritura del código.

```
for n in (1,2,3,4,5,6):  
    print(n, n*10, n*100)  
    print('-----', n, '-----')  
    print()
```

# Generalidades

---

## Comentarios

- Texto incluido dentro del código pero que no tiene ningún efecto sobre la ejecución del código
- 2 formas de escribir comentarios:
  - Con la almohadilla #, podemos escribir comentarios de una sola línea: # Comentario
  - Con las triples comillas podemos escribir código de varias líneas. Son válidas las comillasdobles y las comillas simples:  
""" Este texto aunque sea de varias líneas contiene un comentario """