

# Introducción a Python

JAVIER CÓZAR

# Recursos de un lenguaje de programación

## CONTROL

TIPADO, VARIABLES Y  
OPERADORES

ESTRUCTURAS DE  
CONTROL

FUNCIONES

PAQUETES

## ESTRUCTURAS DE DATOS

LISTAS

DICCIONARIOS

TUPLAS Y CONJUNTOS

I/O Y EXCEPCIONES

# Python

- Fácil, **intuitivo** y potente
- **Agilidad** en desarrollo
- Interpretado y portable
- Libre (Python Software Foundation)
- **Propósito general**

*Guido Van Rossum  
(Benevolent  
Dictator for Life)*



# Python



- <https://www.python.org>
- Soporte oficial
- Para usuarios técnicos
- Flexible
- Instalación básica



- <https://www.anaconda.com>
- Privada, distribución enterprise
- Cómoda para usuarios no técnicos
- Gestionada
- Instalación stack científico

# Modelo de ejecución

```
python myscript.py
```

- Modo batch (scripts o programas)
- Modo interactivo (REPL)

READ  
EVAL  
PRINT  
LOOP

```
Python 3.6.7 (v3.6.7:6ec5cf24b7, Oct 20  
2018, 03:02:14)
```

```
[GCC 4.2.1 Compatible Apple LLVM 6.0  
(clang-600.0.57)] on darwin
```

```
Type "help", "copyright", "credits" or  
"license" for more information.
```

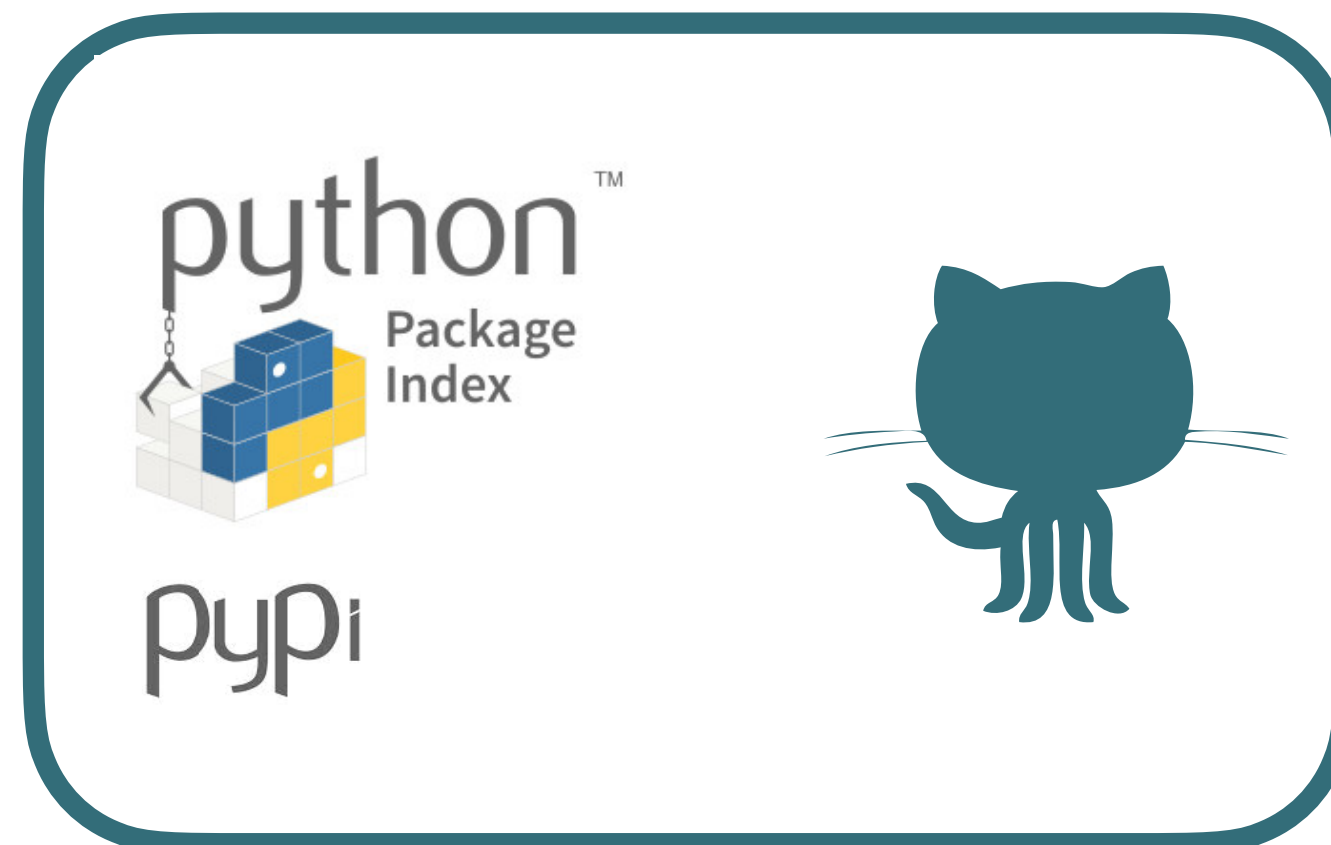
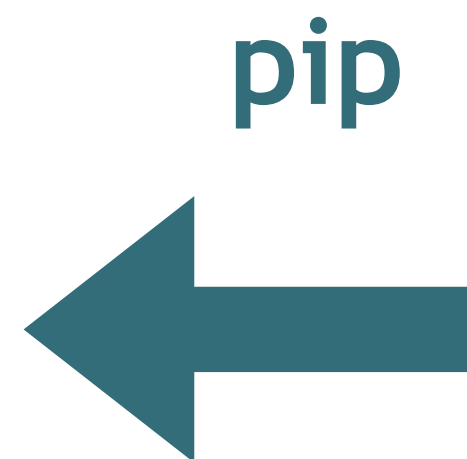
```
>>> print("Hello")
```

```
Hello
```

```
>>>
```

# Módulos y dependencias

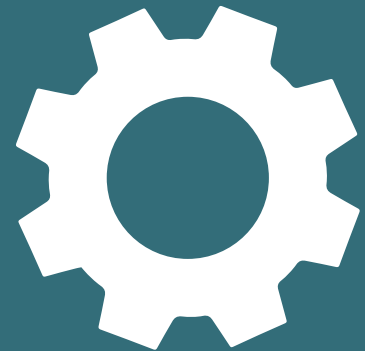
- Python divide su funcionalidad en **paquetes** y **módulos**
- Python incluye funcionalidad básica (**core packages**)
- Podemos ampliar la funcionalidad mediante nuevos paquetes y módulos
- La **instalación** de estas dependencias supone descargar el código fuente de repositorios
- El módulo **pip** se encarga de gestionar estas descargas



# Objetivos



Hemos visto los elementos básicos de python y aquellos recursos que nos permiten **manejar la información** en Python



Hemos trabajado con un **conjunto de datos real**, cargado a partir de un fichero



¡Hay que practicar para aprender! **Kaggle** es un recurso muy bueno

# Introducción a Python

JAVIER CÓZAR