

# Introducción a Python

**Manuel Kaufmann**

humitos@gmail.com

<http://elblogdehumitos.com.ar/>

Universidad Autónoma  
de Encarnación

Mayo 2015

# argentina



argentina

# *Disclaimer*

*Nuevo en Jupyter<sup>1</sup> y Pysenteishon<sup>2</sup>*

[1] <https://jupyter.org/>

[2] <https://github.com/edvm/pysenteishon/>

argentina

“  
*Python es un lenguaje de programación  
poderoso y fácil de aprender...*

– <http://tutorial.python.org.ar>

en  
python  
.com.ar

“... *y está buenísimo*”

argentina

– Python Argentina

en  
python  
.com.ar

# Agenda

- ¿Qué es **Python**?
- Intérprete interactivo
- Manipulando datos
- Tomando decisiones
- **Organizando** el código

argentina





# *¿Qué es Python?*

*Propiedades y características*

# Características básicas

- Gratis y **Libre**
- Maduro e inquieto (+24 años)
- **Fácil** de aprender
- Sintaxis **limpia** y simple
- “Demasiado” portable  
(Windows, Linux, Mac, Android, ...)
- Enorme **comunidad**

# Propiedades del lenguaje

- Compila a bytecode interpretado
  - ✓ La compilación es implícita y automática
  - ✓ Tipado dinámico, pero fuerte
- Multi-paradigma
  - ✓ Todo es un objeto
  - ✓ Pero puede usarse de manera procedural
- Manejo moderno de errores

# Características “pulenta”

- Baterías incluídas
  - ✓ Extensa biblioteca estándar
- Tipos de datos de **alto nivel**
  - ✓ Enteros sin límites, strings, **diccionarios**, listas
  - ✓ Pero puede usarse de manera procedural
- Intérprete interactivo
  - ✓ Permite **explorar, probar** e incluso ver la **doc**

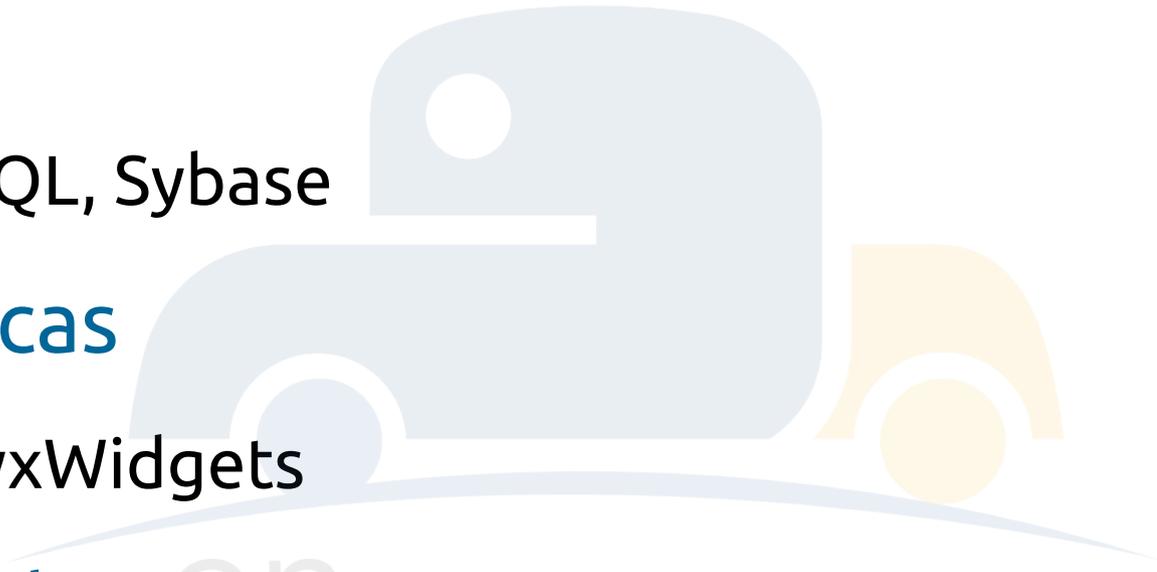
# Baterías incluidas

datetime, os, threading, urllib, unittest, sys, zipfile, csv, re, random, Tkinter, tarfile, mimetypes, logging, socket, json, math, glob, urlparse, optparser, sqlite3, subprocess, hashlib, collections, **antigravity**, decimal, pdb, gettext, md5, user, smtplib, shutil, webbrowser, xml, turtle, encodings, time, doctest, mutex, email...

# Baterías añadidas

- **Base** de datos
  - ✓ MySQL, PostgreSQL, Sybase
- Interfaces **gráficas**
  - ✓ Qt, GTK, win32, wxWidgets
- Frameworks **Web**
  - ✓ Django, webpy, Flask
- Imágenes, Juegos, **Ciencia...**

argentina



en  
python  
.com.ar

# Python Argentina

- ¿Quiénes somos?
  - ✓ Grupo de **entusiastas** del lenguaje
- ¿Cómo participar?
  - ✓ Suscribiéndote a la **Lista de Correo** (~ 1300)
  - ✓ Canal de IRC: **#pyar** en Freenode
- **PyAr es federal**
  - ✓ Eventos en **todas** las provincias

<http://python.org.ar>

argentina

# *Intérprete interactivo*

*Acción y diversión*

python  
.com.ar

The Python logo is a large, light blue silhouette of a snake, positioned behind the main text. It is partially obscured by the text and the 'en python' text below.

# *Preguntas*

*¡Decilo!*

en  
python  
.com.ar



# *Manipulando datos*

*Tipos y usos*

Jupyter Notebook

# Números

- Enteros

- ✓ Suma
- ✓ Paréntesis
- ✓ Módulo
- ✓ Potencia

- Float

- Otras bases (*0xf4*, *hex()*, *bin()*)

argentina



en  
python  
.com.ar

<http://bit.ly/1HsoOIV>

# Strings *(cadenas)*

- Comillas, apóstrofes y multilínea
- Operaciones
  - ✓ Suma
  - ✓ Multiplicación
- Funciones y métodos
  - ✓ len()
  - ✓ .encode('utf-8')

<http://bit.ly/1L6wUVp>

# Acceso a strings *(cadenas)*

- Posicionales
  - ✓ Índice (positivo / **negativo**)
- Rebanadas
  - ✓ [cerrado, **abierto**)
  - ✓ Índices (positivo / negativo / omitido)
  - De a **saltos**

# Listas

- **Diferentes** tipos de elementos
- Accedemos como cualquier **secuencia**
- Concatenamos, **reemplazamos**, borramos
- Listas **dentro** de listas
- Métodos
  - ✓ `.index("elemento")`
  - ✓ `.sort()`

<http://bit.ly/1AcVMuh>

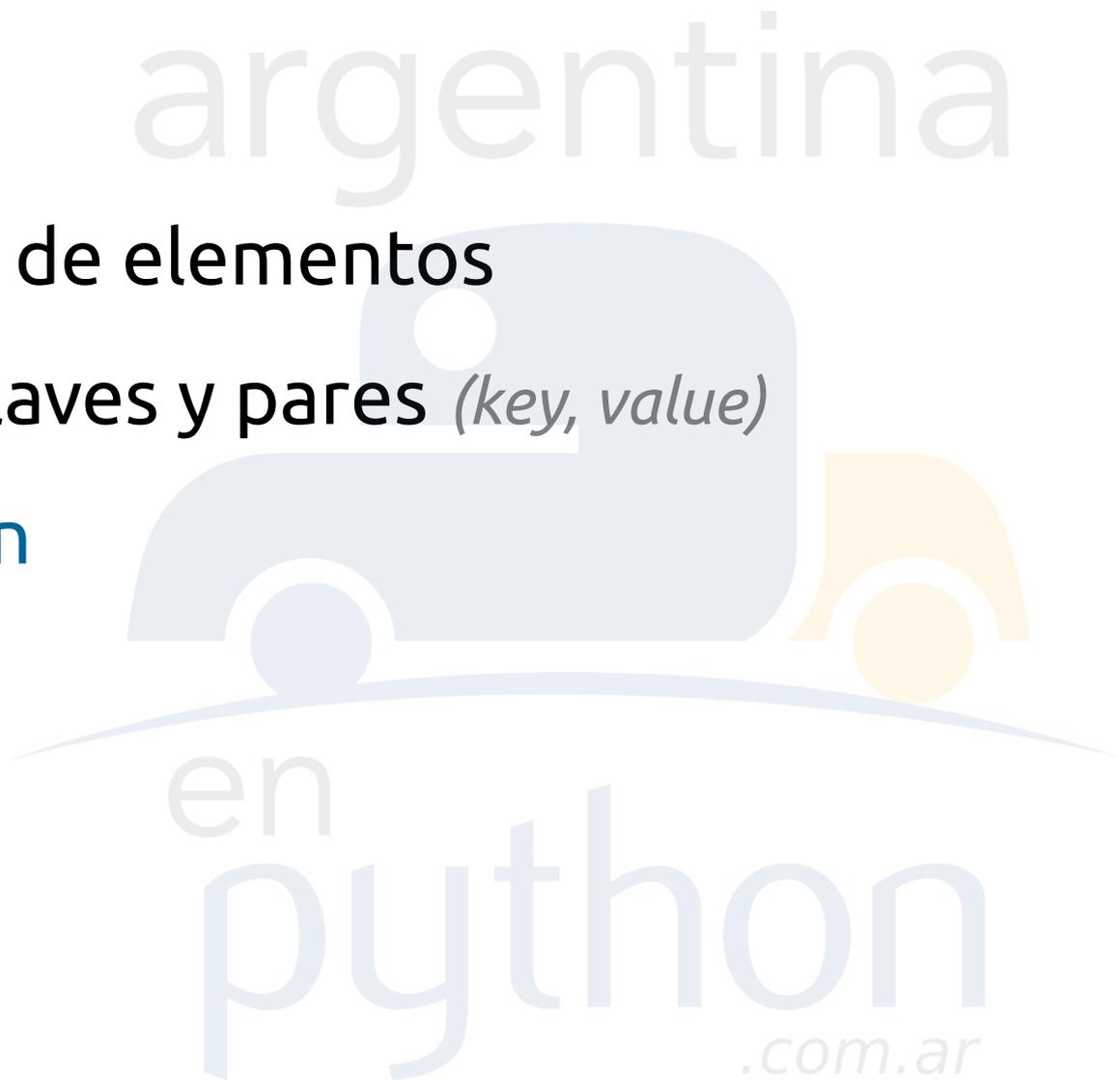
# Conjuntos

- **Diferentes** tipos de elementos
- Se definen con llaves y valores
- Propiedades **matemáticas** de conjuntos
- Operaciones (*/ & - ...*)
  - ✓ `.update([...])`
  - ✓ `.intersection({...})`
  - ✓ `.union({...})`

<http://bit.ly/1AcVrrK>

# Diccionarios

- **Diferentes** tipos de elementos
- Se definen con llaves y pares (*key, value*)
- **No poseen orden**
- Operaciones
  - ✓ `.keys()`
  - ✓ `.get(...)`
  - ✓ `.copy()`



<http://bit.ly/1JXn3kv>

The Python logo is a large, light blue silhouette of a snake, positioned behind the main text. It is partially obscured by the text and the 'en python' text below.

# *Preguntas*

*¡Decilo!*

en  
python  
.com.ar



argentina

# *Tomando decisiones*

*Si pasa esto, voy por acá*



# Si pasa, esto... si no, aquello...

- Estructura

- ✓ `if`, `elif`, `else`

- Comparadores

- ✓ `or`, `and`, `not`

- ✓ `<` `>` `==` `!=` `<=` `>=` `in` `is`

- ✓ **Todo** evalúa a `True` o `False`

<http://bit.ly/1Fybhit>

# Por cada elemento...

- Estructura
  - ✓ `for`, `in`
  - `continue`, `break`, `else`
- ¿Se terminó la secuencia?
  - ✓ Sigo
  - ✓ Paro
- ¿Y si necesitamos **sólo** números?

<http://bit.ly/1IOXWBz>

# Mientras suceda... argentina

- Estructura
  - ✓ `while`
  - `continue`, `break`, `else`
- ¿Se cumple la condición?
  - ✓ Sigo
  - ✓ Paro



<http://bit.ly/1FsNdMg>

# Excepciones

- Estructura
  - ✓ try, **except**
  - else, finally
- Algo se escapa de **lo normal**
  - ✓ Podemos capturarlas
  - ✓ Nos aparece un error
- Podemos generar excepciones

<http://bit.ly/1dii43O>

The Python logo is a large, light blue silhouette of a snake, positioned behind the main text. It is partially obscured by the text and the 'en python' text below.

# *Preguntas*

*¡Decilo!*

en  
python  
.com.ar

argentina

# *Organizando el código*

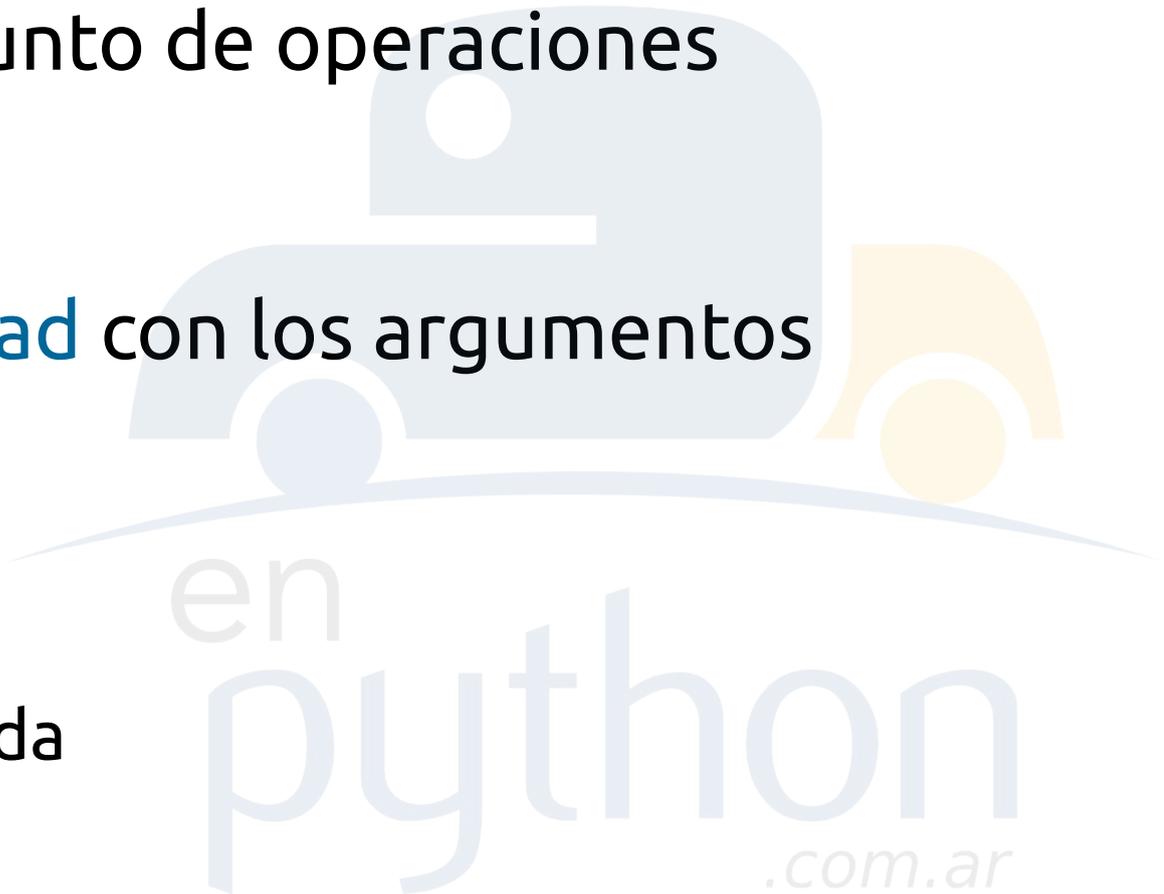
*Funciones, clases, módulos y paquetes*

python  
.com.ar

# Funciones

- Definen un conjunto de operaciones
- Son **objetos**
- Amplia **flexibilidad** con los argumentos
  - ✓ Opcionales
  - ✓ **Nombrados**
  - ✓ Cantidad indefinida

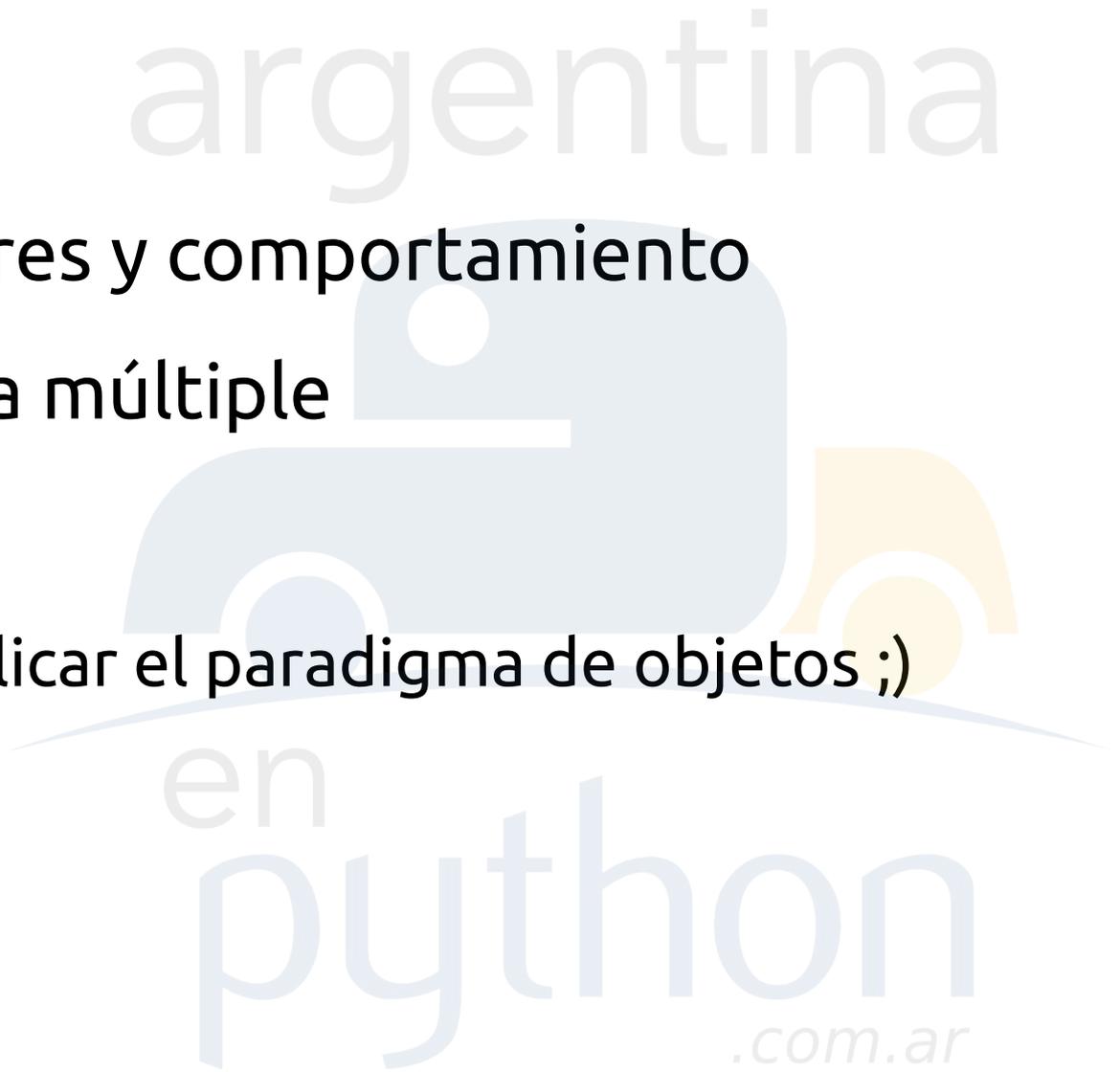
argentina



# Clases

- Encapsulan valores y comportamiento
- Soporta herencia múltiple

... no voy a explicar el paradigma de objetos ;)



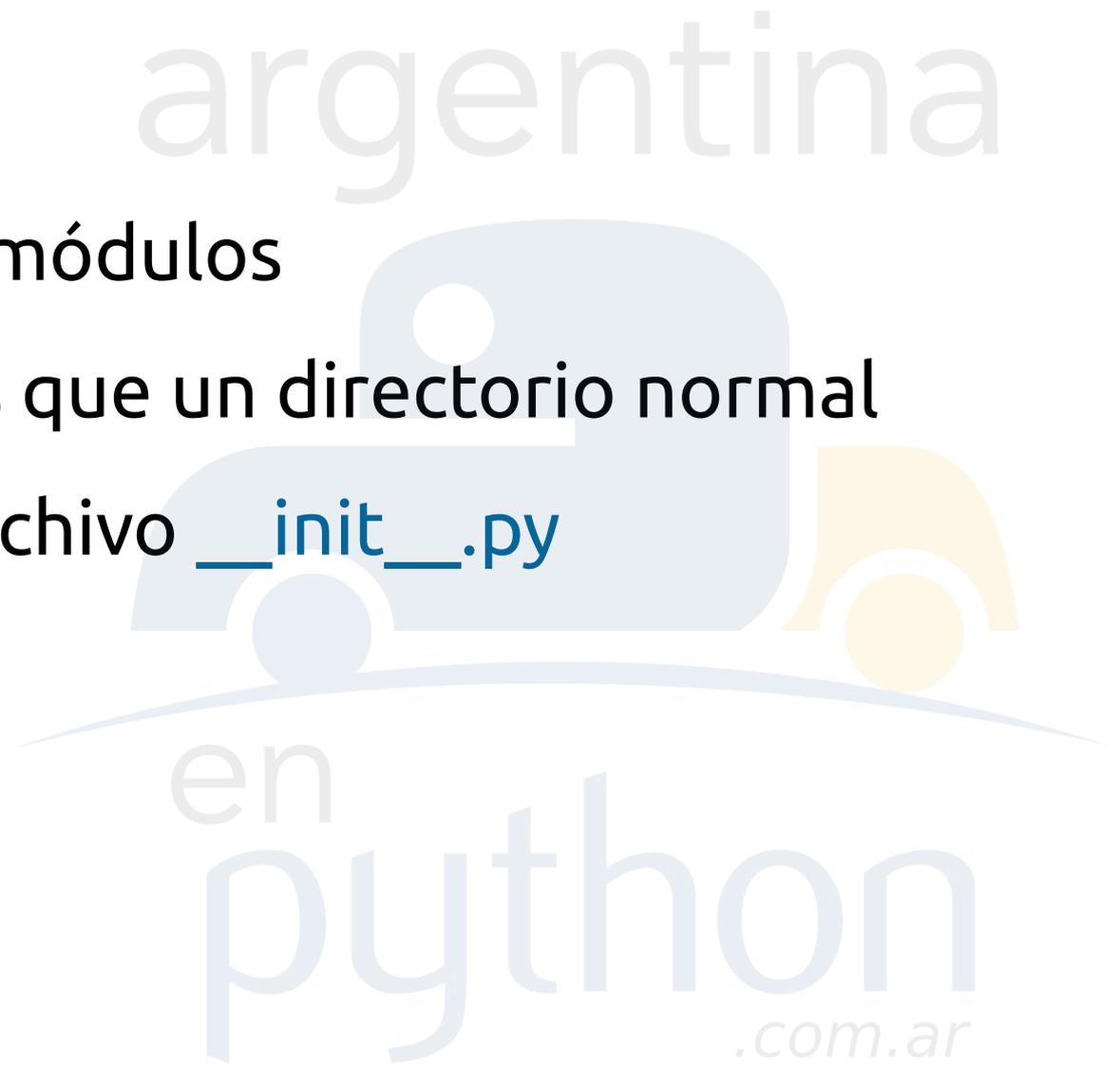
# Módulos

- Funciones, clases, constantes en un archivo
- Es un `.py` normal que lo importo



# Paquetes

- Un conjunto de módulos
- Ni más ni menos que un directorio normal
- Utilizamos un archivo `__init__.py`



The Python logo is a large, light blue silhouette of a snake, positioned behind the main text. It is partially obscured by the text and the 'en python' text below.

# *Preguntas*

*¡Decilo!*

en  
python  
.com.ar

argentina

# *Algunas perlitas*

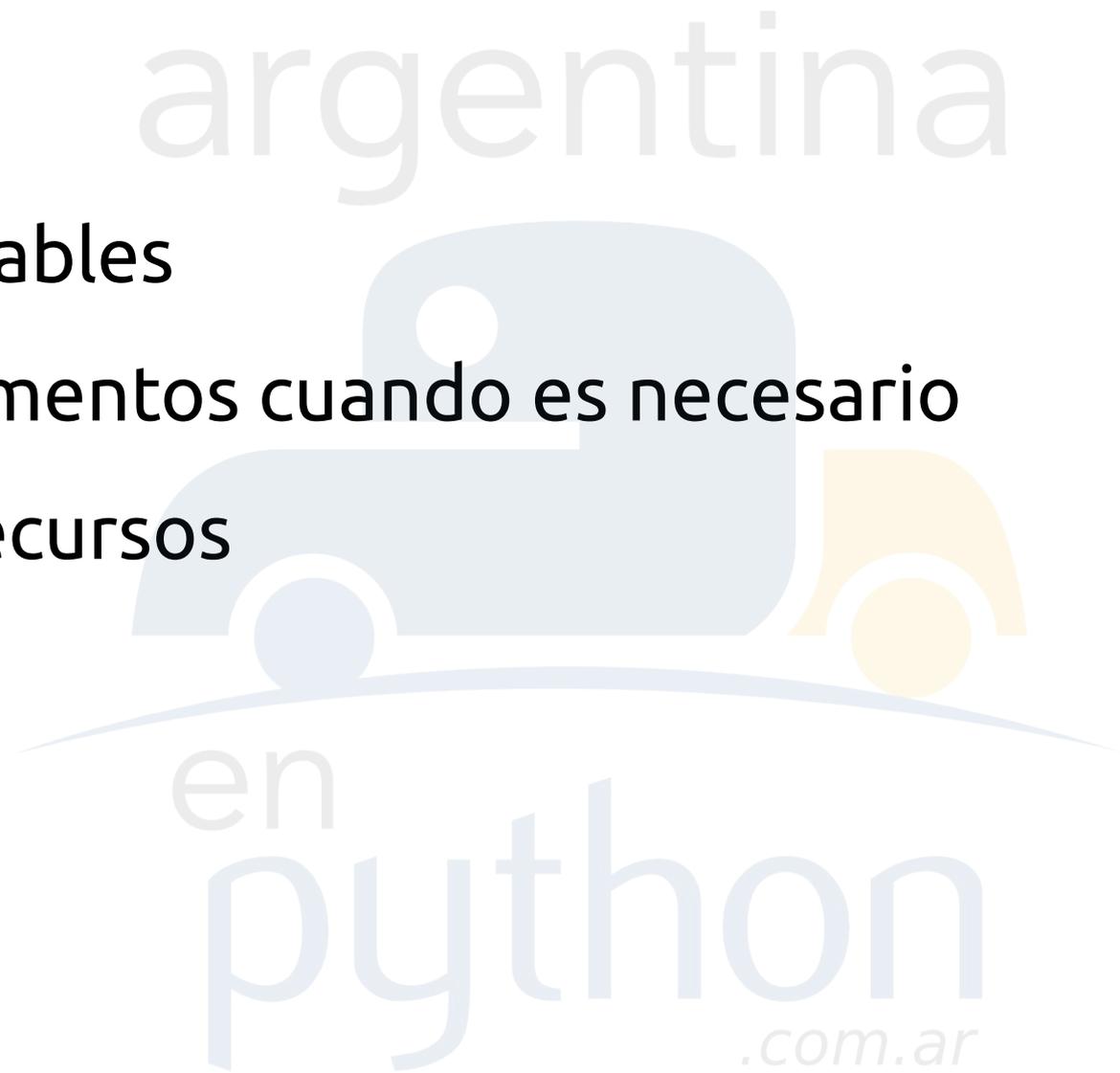
*Syntactic sugar, tipos 'copados', decoradores*

Jupyter Notebook

.com.ar

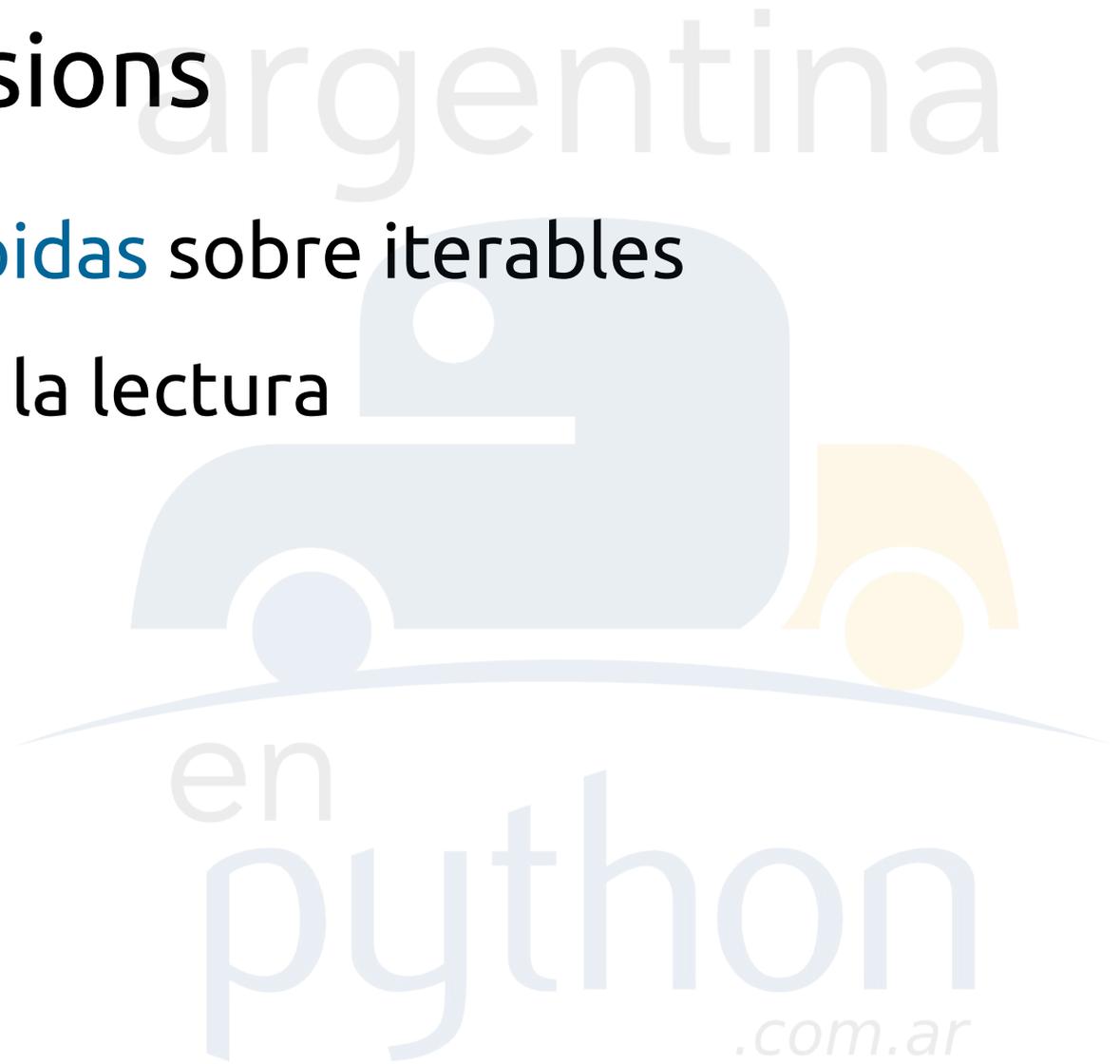
# Generadores

- Similar a los iterables
- **Generan** los elementos cuando es necesario
- Optimizan los recursos



# List Comprehensions

- Operaciones **rápidas** sobre iterables
- Puede dificultar la lectura



# Context Managers

- Estructura
  - ✓ `with`
- Dentro de un `contexto`
- Ejecutar “algo” al ingresar y al salir



# Namespaces

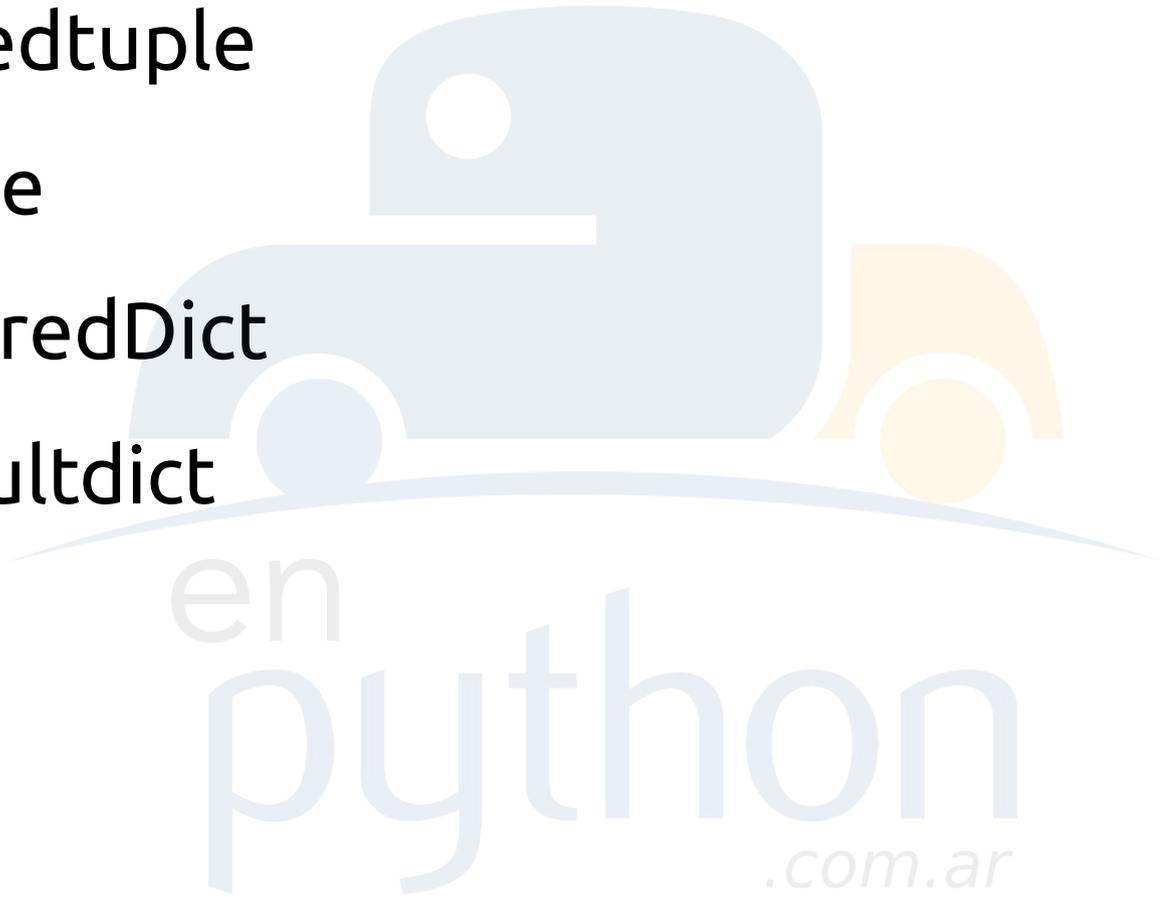
- Muy **útiles** para organizar el código
- Aislan diferentes **espacios** de variables
- Local y **global**



# Tipos “copados”

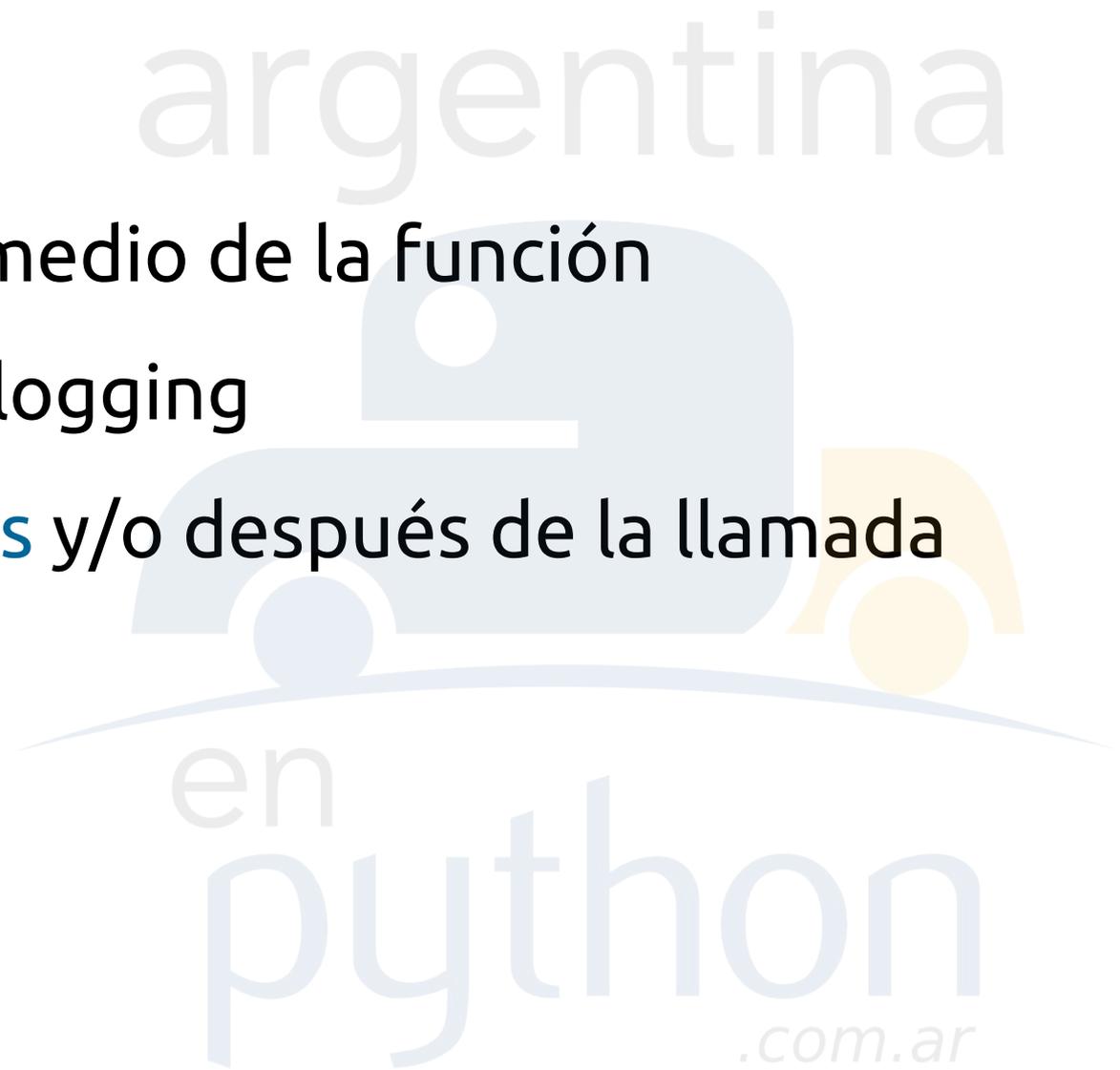
- `collections.namedtuple`
- `collections.deque`
- `collections.OrderedDict`
- `collections.defaultdict`

argentina



# Decoradores

- Se **meten** en el medio de la función
- Muy útiles para logging
- Hacen algo **antes** y/o después de la llamada



¿Preguntas? *¿Sugerencias?*

*Gracias por su tiempo ...*

*... y espero que hayan disfrutado*

**Manuel Kaufmann**

humitos@gmail.com

<http://elblogdehumitos.com.ar/>



Este obra está bajo una licencia de Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional.