# PYTHON

## **Diferencias entre Python y IPython**

Existen pequeñas diferencias entre ambos lenguajes. En ipython hay comandos ‘’mágicos’’ que no funcionan en Python normal. Son excepciones.

## **Jupyter notebook**

Para la ayuda es shift+tab y clicas sobre el comando que quieres tener información.

Un set es un diccionario en el que solo hay keys, y estas keys son únicas, es decir, no se pueden repetir. Además son también aleatorias.

Es importante tener en cuenta cuando pones que b=a. Si son objetos si que se va a copiar la misma ‘’referencia’’ por lo que si cambias uno cambias otro, pero si son números, por ejemplo a=6, a=b,a=7, en este caso b no cambia ya que lo has referenciado a un solo número, no se guarda un objeto en memoria.

## **Comandos para la Shell con Jupyter Notebook:**

Por definición, para ejecutar los comandos con bash se emplea **!** y se escribe el comando. CADA COMANDO DEBE IR CON ! por lo que solo se ejecuta 1 comando por cada !.

Es muy importante saber movernos por la Shell desde jupyter. Para ellos tenemos los comandos mágicos, se pueden ver con **%lsmagic.** Con **%automagic** quitamos y ponemos el modo automático. Es mejor dejarlo en off:

* Con % ejecutamos el comando
* Con %% podemos definir una célula con %%! Y nos ejecutará todo el código que escribamos.

Comandos útiles para listas de IPython:

* <var>.n : une los elementos de la lista mediante ‘’\n’’
* <var>.s : une los elementos de la lista mediante espacios

## **Estructuras en Python**

Las principales estructuras que hay en Python son:

* List: [ ] Se puede meter todos los datos que quieras. Son modificables.
* Tupla: ( ) Se puede meter los datos que quieras. Es inmutable.
* Dictionary:{ } Sirve para introducir parejas de key:value.
* Set: ( ) Es un diccionario de keys. Tienen que ser valores únicos y además no tienen ningún orden. Si haces un set de una lista obtienes todos los valores únicos de esa lista.

Funciones útiles

* isinstance (variable, tupla con las opciones (float,dict….)): Nos permite comprobar si la variable es de un tipo en concreto.
* hasattr (variable, atributo) : nos dice si el objeto tiene el atributo (Split por ejemplo).
* dir (variable) : te da los atributos del objeto. Si pones dir() sin nada te da como el ‘’historial’’ de todos tus comandos.
* id(variable) : te da el id de la variable
* <variable>.copy() : copia el contenido, permitiendo que no sea la misma referencia y manteniendo independientes las variables
* \_\_\_.find(valor a buscar)
* \_\_\_\_\_\_\_\_\_.strip(): te permite eliminar los espacios en blanco del principio y del final