

# Taller Python

## CENPAT 2019



# Temario



# Temario - Lunes

Presentación del lenguaje

¿Dónde y cómo hablamos Python? (consola, scripts, Jupyter Lab).

Librerías (presentación e instalación)

Instalación de herramientas, presentación de Jupyter Lab.



# Temario - Martes

Tipos de datos básicos.

Estructuras de control, decisión y repetición.

Funciones.



# Temario - Miércoles

Estadísticas descriptivas con Python.

Estructuras de datos complejas:

Arreglos Numpy.

Estructuras de datos tabulares (Dataframes Pandas):



# Temario - Jueves

Lecturas de archivos de texto, csv, excel, etc.

Operaciones (descripción, limpieza, combinación, etc)

Visualizaciones básicas con matplotlib.

Scipy para computación científica.



# Temario - Viernes

Introducción a GIS con geopandas.

¿Cómo compartimos nuestro código?

Introducción básica a Github.



# Día 1



# Presentación del lenguaje

“Python es un lenguaje de programación interpretado cuya filosofía hace hincapié en la legibilidad de su código.

Se trata de un lenguaje de programación multiparadigma, ya que soporta orientación a objetos, programación imperativa y, en menor medida, programación funcional. Es un lenguaje interpretado, dinámico y **multiplataforma**”.



# ¿Dónde y cómo hablamos Python?

```
(base) λ ipython
Python 3.6.5 |Anaconda, Inc.| (default, Mar 29 2018, 1
3:32:41) [MSC v.1900 64 bit (AMD64)]
Type 'copyright', 'credits' or 'license' for more info
rmation
IPython 7.2.0 -- An enhanced Interactive Python. Type
'?' for help.
```

```
In [1]: mensaje = "Hola CENPAT!"
```

```
In [2]: print(mensaje)
Hola CENPAT!
```

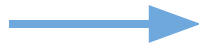
```
In [3]: print(mensaje.upper())
HOLA CENPAT!
```

```
In [4]: |
```

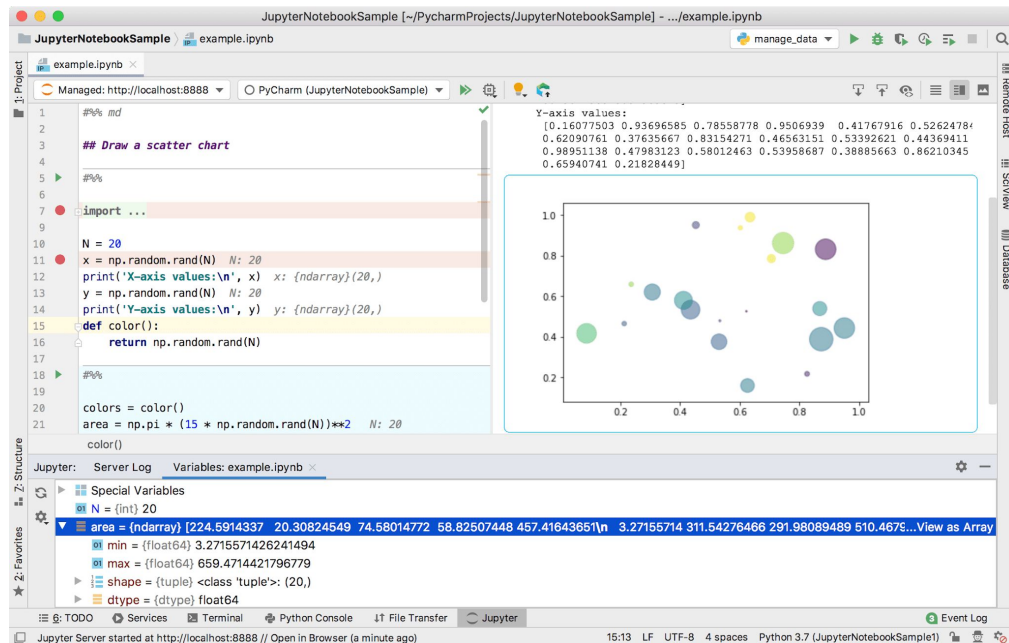


# ¿Dónde y cómo hablamos Python?

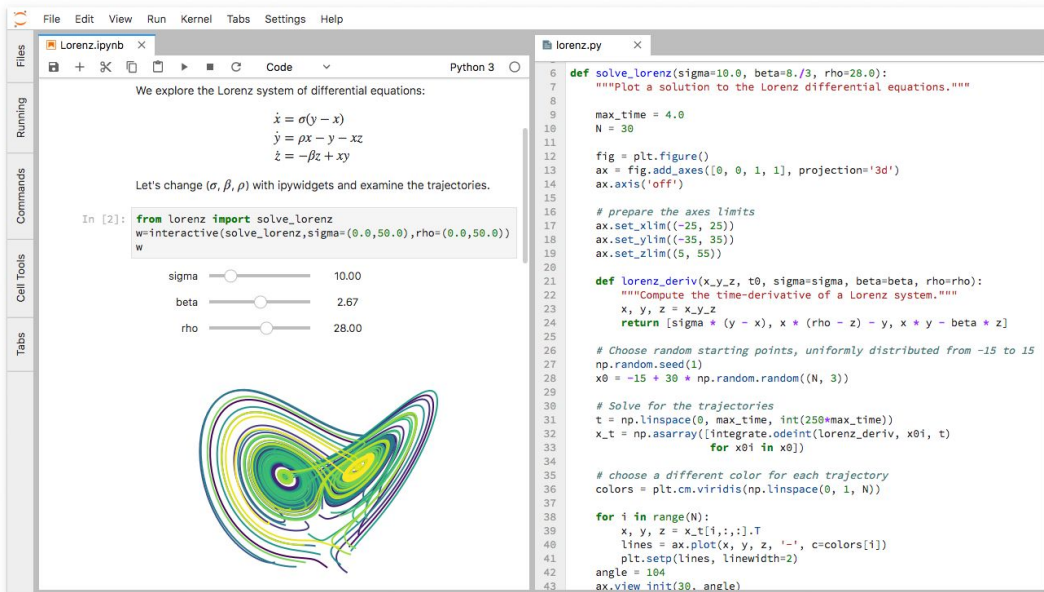
```
def add5(x):  
    return x+5  
  
def dotwrite(ast):  
    nodename = getNodeName()  
    label=symbol.sym_name.get(int(ast[0]),ast[0])  
    print '    %s [label="%s' % (nodename, label),  
    if isinstance(ast[1], str):  
        if ast[1].strip():  
            print '= %s"' % ast[1]  
        else:  
            print '"]'  
    else:  
        print '"]';'  
    children = []  
    for n, child in enumerate(ast[1:]):  
        children.append(dotwrite(child))  
    print '    %s -> {' % nodename,  
    for name in children:  
        print '%s' % name,
```



# ¿Dónde y cómo hablamos Python?



# ¿Dónde y cómo hablamos Python?



# Librerías



# PYTORCH

