

# Documentación

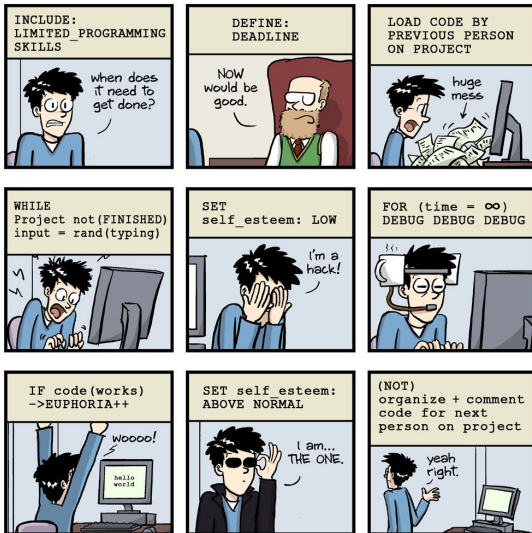
C. Jarne

cecilia.jarne@unq.edu.ar



# Ideas básicas para empezar:

## PROGRAMMING FOR NON-PROGRAMMERS



JORGE CHAM © 2014

[WWW.PHDCOMICS.COM](http://WWW.PHDCOMICS.COM)

# Documentar?

## Pros:

- Acelera la comunicación entre miembros del equipo que programa
- reduce el tiempo para entender un proyecto ya en curso
- Permite Organizar grandes proyectos
- Mejora la velocidad del desarrollo.

## Contras:

- Consume tiempo (y dinero)
- Se desactualiza rápidamente
- A los desarrolladores no les gusta
- Si no se conocen las herramientas correctas puede ser difícil

# Ideas básicas para empezar:

Documentar significa comunicar y es necesario en todos los niveles

- Anotaciones en el código: de formato, comentarios, de estructuración de funciones de clases.
- Manual de usuario o de referencia
- Introducción para nuevos usuarios o desarrolladores.

El código que escribimos puede ser reutilizado y leído varias veces.

- Es importante la claridad, mas que la astucia al escribir.
- Elegir un estilo y respetarlo: estructura de bloques, indentación, ser cuidadoso con el nombre de las variables, la longitud de linea.
- Respetar las convenciones de lenguaje la comunidad en la que uno esta trabajando (en Python, ver PEP 8)
- Ser consistente (no cambiar a la mitad!!)

# Comentarios sobre el código

- Explicar la intención más que el trabajo hecho.
- También cualquier desviación de respecto de los estándares.
- Opciones de implementación inesperadas.

```
1 def Tripletupla(x):
2     # assign x to y
3     y=x
4     # assign x to z
5     z=x
6     # double y
7     y*=2
8     #triple z
9     z*=3
10    #create tupla
11    t=(x,y,z)
12    #return tupla
13    return t
```

```
1 def Tripletupla(x):
2
3     y = x = z
4     y*= 2
5     z*= 3
6     #Applied for scaling, see [34]
7     eq. (2.3)
8     t=(x,y,z)
9     #return tupla
10    return t
```

## Comentarios en C

```
1
2 // Comento asi
3
4 /* Comento asi
5 mas de una linea*/
```

## Comentarios en python

```
1 #Comento asi
2
3 '''
4 Comento asi mas
5 de una linea
6
7 '''
```

# Docstrings and PEP 257:one-line

- String literal: la primera instrucción en un módulo, función, etc.
- Todos los módulos, funciones y clases normalmente debería tener cadenas de documentación
- Se usa `""" " asi " """`
- Debe ser un comando ( “Haz esto”, “Devolver eso” ),
- La naturaleza del valor de retorno debe ser mencionada

```
def function(a, b):  
    """Do X and return a list."""
```



# Docstrings and PEP 257: Multi-line

- Una línea de resumen (como una línea unifilar) + una linea en blanco.
- Más descripción
- Todo tiene la misma sangría que las comillas

```
def complex_number(real=0.0, imag=0.0):  
    """Form a complex number.  
  
    Keyword arguments:  
    real -- the real part (default 0.0)  
    imag -- the imaginary part (default 0.0)  
    """  
    if imag == 0.0 and real == 0.0:  
        return complex_zero  
    ...
```


Existen algunas herramientas que automáticamente extraen la documentación escrita en el código


- Pydoc
- Doxygen
- Sphinx

- Soporte para C++, C, ObjC,
- C#, PHP, Java, Python, IDL, Fortran, VHDL, Tcl
- Puede ser estructurada desde files sin documentar.
- Visualización gráfica de las dependencias
- Permite escribir también paginas generales

*[https : //www.stack.nl/ dimitri/doxygen/manual/starting.html](https://www.stack.nl/~dimitri/doxygen/manual/starting.html)*

- Soporte muy bueno para los principiantes.
- Especial para generar documentación en python.

 SciPy.org

 ENTHOUGHT

Scipy.org

Docs

NumPy v1.12 Manual

NumPy Reference

Routines

Statistics

index

next

previous

## numpy.correlate

**numpy.correlate**(a, v, mode='valid')

Cross-correlation of two 1-dimensional sequences.

This function computes the correlation as generally defined in signal processing texts:

```
c_[av][k] = sum_n a[n+k] * conj(v[n])
```

with a and v sequences being zero-padded where necessary and conj being the conjugate.

**Parameters:**

- a, v** : array\_like  
Input sequences.
- mode** : {'valid', 'same', 'full'}, optional  
Refer to the [convolve](#) docstring. Note that the default is 'valid', unlike [convolve](#), which uses 'full'.
- old\_behavior** : bool  
old\_behavior was removed in NumPy 1.10. If you need the old behavior, use [multirarray.correlate](#).

**Returns:**

- out** : ndarray  
Discrete cross-correlation of a and v.

**See also:**

- [convolve](#) Discrete, linear convolution of two one-dimensional sequences.
- [multirarray.correlate](#) Old, no conjugate, version of correlate.

**Notes**

The definition of correlation above is not unique and sometimes correlation may be defined differently. Another common definition is:

Previous topic  
[numpy.corrccoef](#)Next topic  
[numpy.cov](#)

- Siempre es buena idea leer el manual de usuario un poco al menos(ahorra tiempo!!)
- También sirve hacer los tutoriales (se aprende mas rápido)

Gracias Totales!