

“Python - Programación”



Rogelio Ferreira Escutia

Programación

Python - Versión

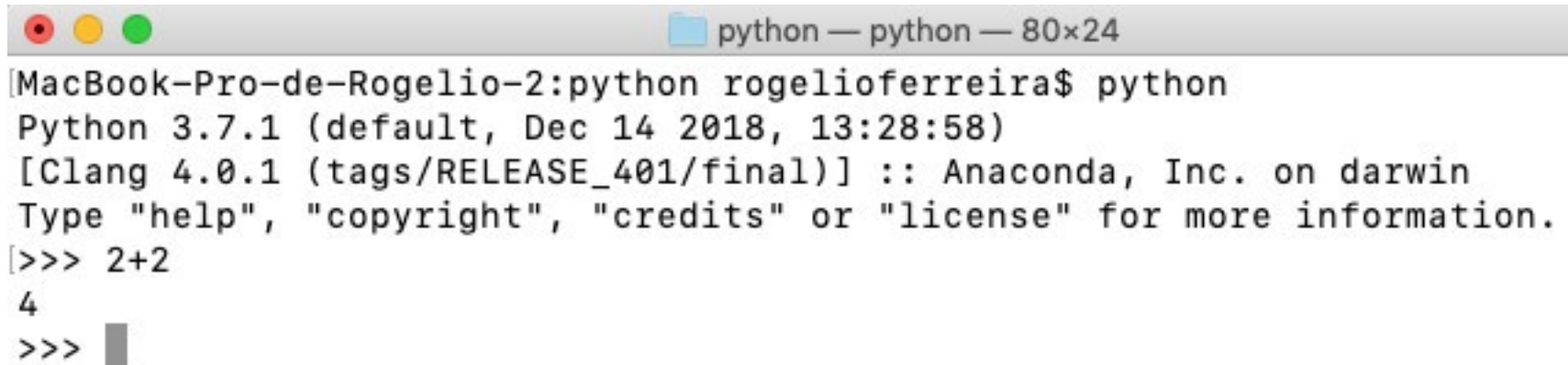
- Para ver la versión instalada (en consola):



```
python — -bash — 80x24
MacBook-Pro-de-Rogelio-2:python rogelioferreira$ python --version
Python 3.7.1
MacBook-Pro-de-Rogelio-2:python rogelioferreira$
```

Python - Intérprete

- Para entrar al intérprete de Python (en consola):



```
python — python — 80x24
[MacBook-Pro-de-Rogelio-2:python rogelioferreira$ python
Python 3.7.1 (default, Dec 14 2018, 13:28:58)
[Clang 4.0.1 (tags/RELEASE_401/final)] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
[>>> 2+2
4
>>> █
```

Python – Intérprete (Ayuda)

- Para entrar a la ayuda en el intérprete de Python (en consola):

```
>>> help()

Welcome to Python 3.7's help utility!

If this is your first time using Python, you should definitely check out
the tutorial on the Internet at https://docs.python.org/3.7/tutorial/.

Enter the name of any module, keyword, or topic to get help on writing
Python programs and using Python modules.  To quit this help utility and
return to the interpreter, just type "quit".

To get a list of available modules, keywords, symbols, or topics, type
"modules", "keywords", "symbols", or "topics".  Each module also comes
with a one-line summary of what it does; to list the modules whose name
or summary contain a given string such as "spam", type "modules spam".

help> keywords

Here is a list of the Python keywords.  Enter any keyword to get more help.

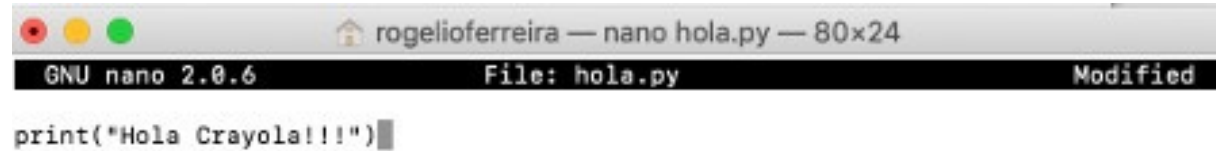
False          class           from            or
None           continue       global          pass
True           def            if              raise
and            del            import          return
as             elif           in              try
assert        else           is              while
async         except         lambda          with
await         finally        nonlocal        yield
break         for            not

help> █
```

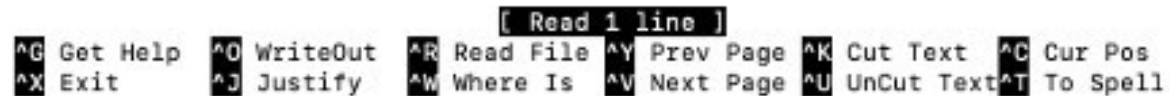


Python – Hola Mundo (consola)

- > nano hola.py



```
rogelioferreira — nano hola.py — 80x24
GNU nano 2.0.6 File: hola.py Modified
print("Hola Crayola!!!")
```



```
[ Read 1 line ]
^G Get Help  ^O WriteOut  ^R Read File  ^Y Prev Page  ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page  ^U UnCut Text ^T To Spell
```

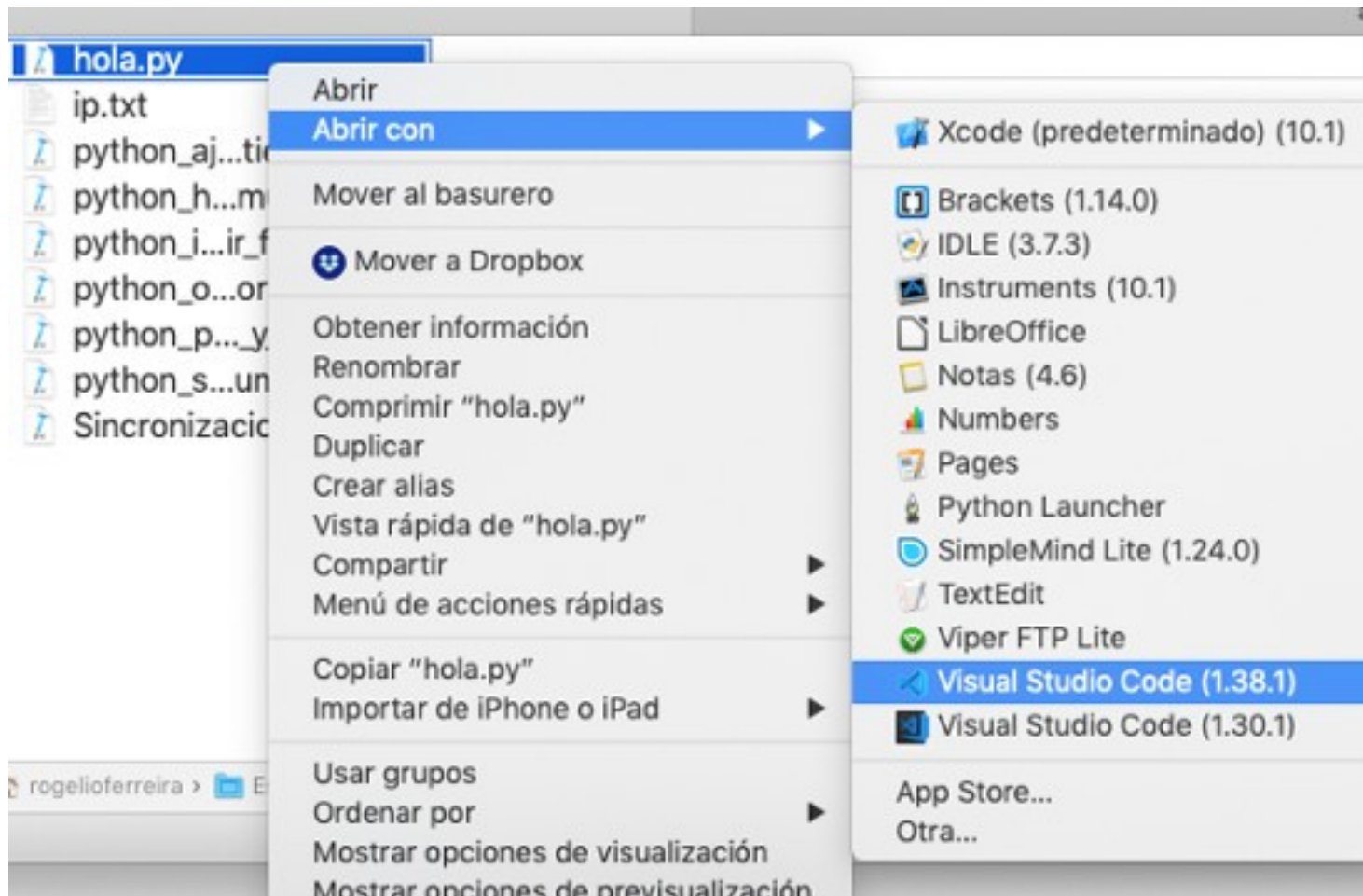
- > python hola.py

Hola Crayola!!!



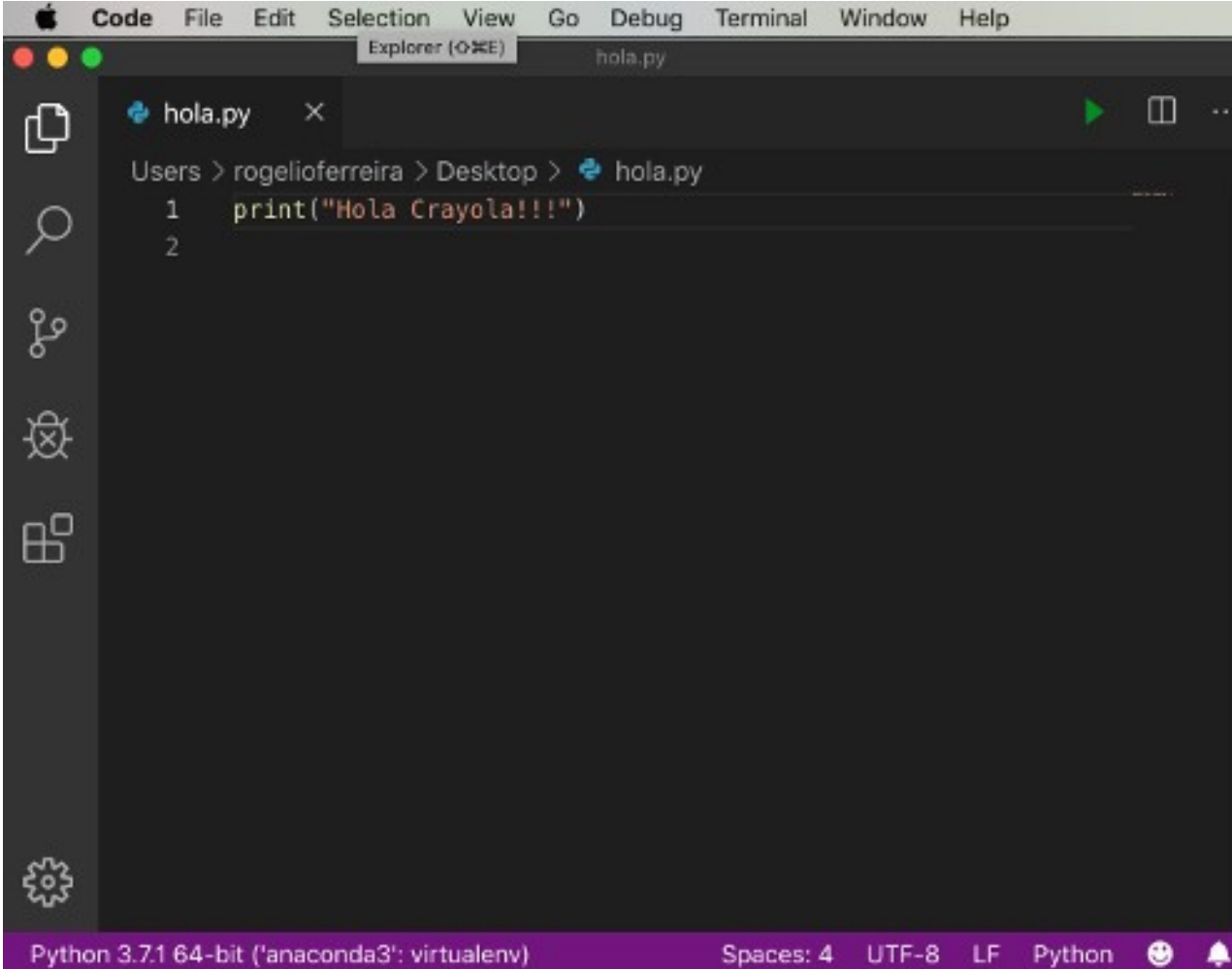
Python – Hola mundo (con VSCode)

- **Abrir el archivo con el Visual Studio Code (previamente instalado)**



Python – Hola mundo (con VSCode)

- Abrir el archivo con el Visual Studio Code (previamente instalado) y correrlo en la consola:



The screenshot shows the Visual Studio Code interface. The top menu bar includes 'Code', 'File', 'Edit', 'Selection', 'View', 'Go', 'Debug', 'Terminal', 'Window', and 'Help'. The Explorer sidebar on the left shows a file named 'hola.py'. The main editor area displays the following code:

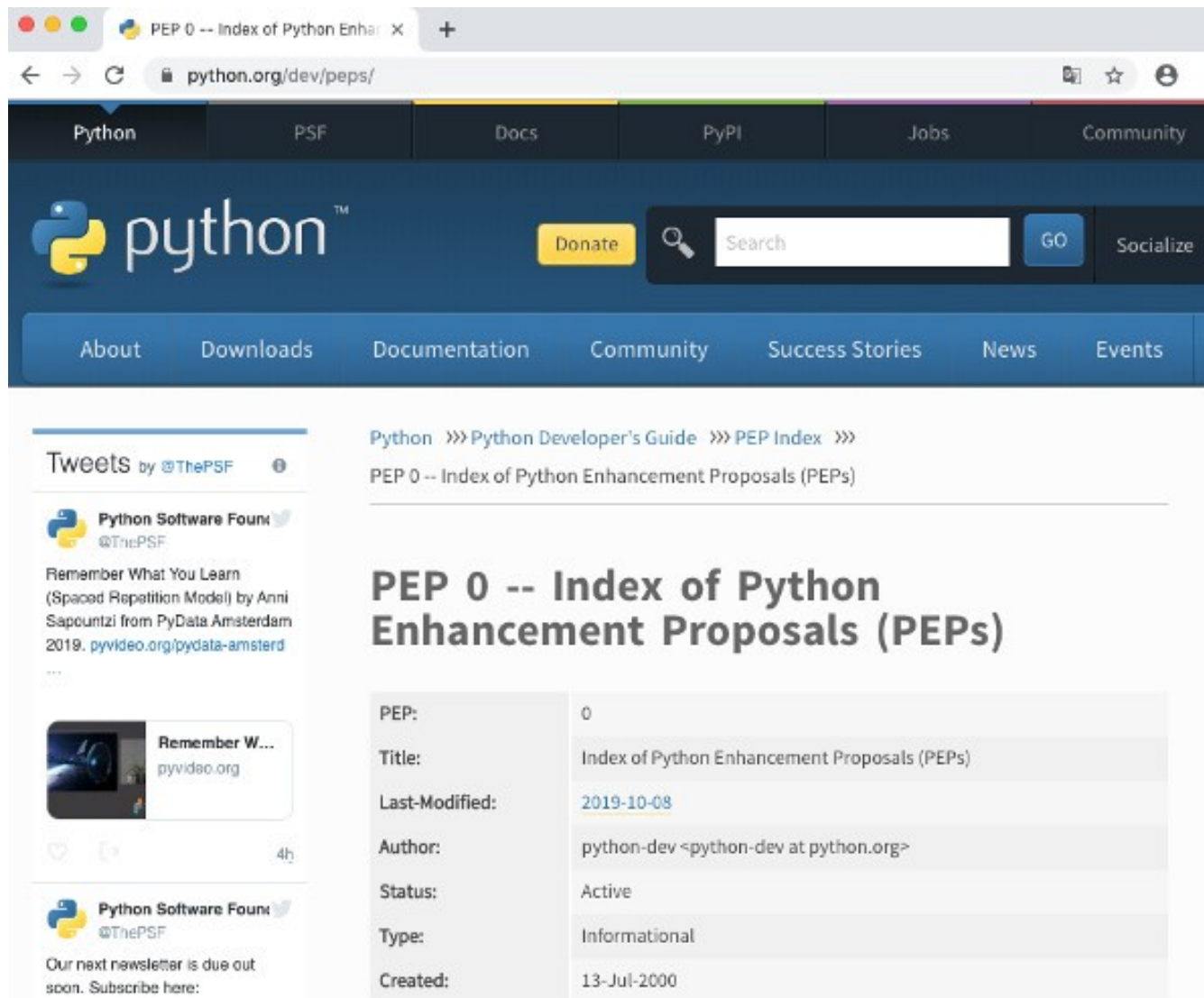
```
Users > rogelioferreira > Desktop > hola.py
1  print("Hola Crayola!!!")
2
```

The status bar at the bottom indicates 'Python 3.7.1 64-bit ('anaconda3': virtualenv)', 'Spaces: 4', 'UTF-8', 'LF', and 'Python'.



Zen of Python

- Es una lista de principios de diseño para el lenguaje Python:



The screenshot shows a web browser displaying the Python.org website. The page title is "PEP 0 -- Index of Python Enhancement Proposals (PEPs)". The page content includes a navigation menu with links for "About", "Downloads", "Documentation", "Community", "Success Stories", "News", and "Events". A search bar is visible with the text "Search" and a "GO" button. The main content area displays the title "PEP 0 -- Index of Python Enhancement Proposals (PEPs)" and a table with the following information:

PEP:	0
Title:	Index of Python Enhancement Proposals (PEPs)
Last-Modified:	2019-10-08
Author:	python-dev <python-dev at python.org>
Status:	Active
Type:	Informational
Created:	13-Jul-2000

On the left side of the page, there is a "Tweets by @ThePSF" section. The first tweet is from Python Software Foundation (@ThePSF) and reads: "Remember What You Learn (Spaced Repetition Model) by Anni Sapountzi from PyData Amsterdam 2019. pyvideo.org/pydata-amsterd...". Below the tweet is a video thumbnail with the text "Remember W... pyvideo.org". The second tweet is from Python Software Foundation (@ThePSF) and reads: "Our next newsletter is due out soon. Subscribe here:".



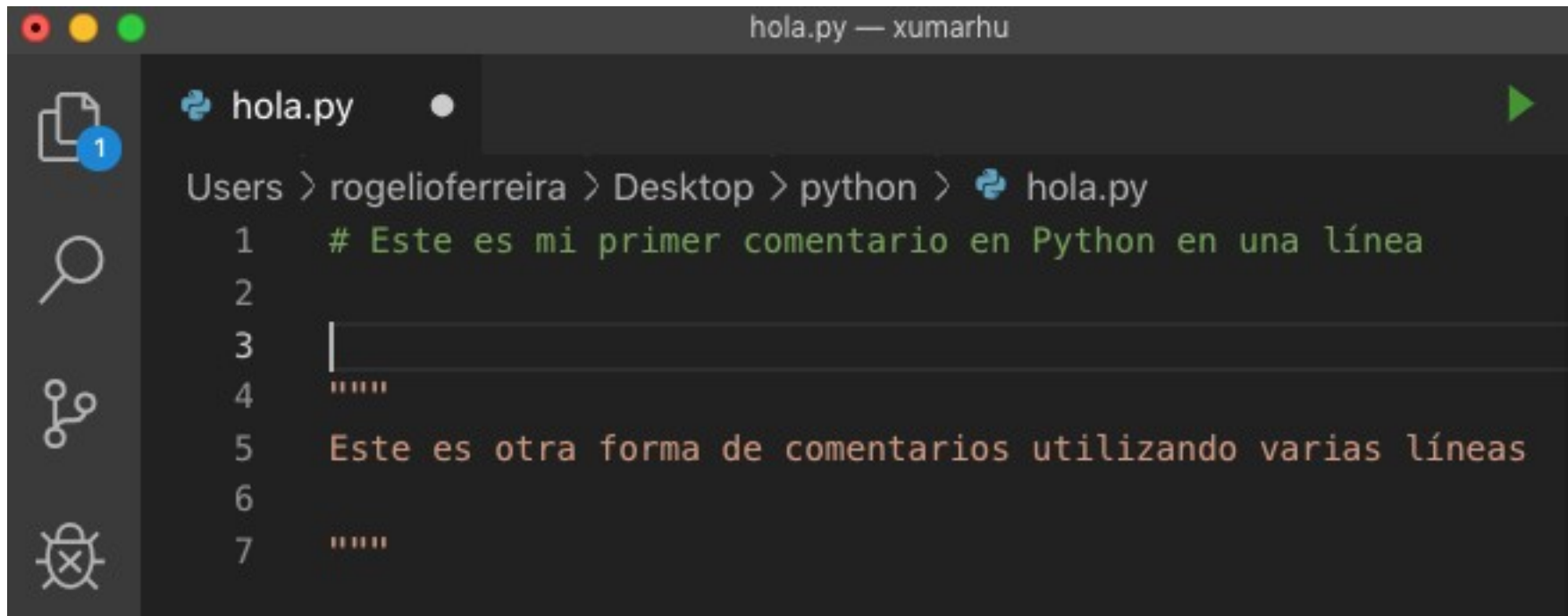
The Zen of Python

```
Beautiful is better than ugly.  
Explicit is better than implicit.  
Simple is better than complex.  
Complex is better than complicated.  
Flat is better than nested.  
Sparse is better than dense.  
Readability counts.  
Special cases aren't special enough to break the rules.  
Although practicality beats purity.  
Errors should never pass silently.  
Unless explicitly silenced.  
In the face of ambiguity, refuse the temptation to guess.  
There should be one-- and preferably only one --obvious way to do it.  
Although that way may not be obvious at first unless you're Dutch.  
Now is better than never.  
Although never is often better than *right* now.  
If the implementation is hard to explain, it's a bad idea.  
If the implementation is easy to explain, it may be a good idea.  
Namespaces are one honking great idea -- let's do more of those!
```



Comentarios

- Para poner comentarios en nuestro código:



The screenshot shows a code editor window titled "hola.py — xumarhu". The editor displays the following Python code:

```
Users > rogelioferreira > Desktop > python > hola.py
1  # Este es mi primer comentario en Python en una línea
2
3  |
4  """"
5  Este es otra forma de comentarios utilizando varias líneas
6
7  """"
```

Tipos de Datos

Enteros (int)

Number	Okay?	Reason
1	Good	A whole number (integer)
1.1	Good	A number with a decimal point
1234567.89	Good	A large number with a decimal point and no commas
-2	Good	A negative number, as indicated by the starting hyphen
.99	Good	A number that starts with a decimal point because it's less than one.
\$1.99	Bad	Contains a \$
12,345.67	Bad	Contains a comma
1101 3232	Bad	Contains a space
91740-3384	Bad	Contains a hyphen
123-45-6789	Bad	Contains two hyphens
123 Oak Tree Lane	Bad	Contains spaces and words
(267)555-1234	Bad	Contain parentheses and hyphens
127.0.0.1	Bad	Only one decimal point is allowed

Cadenas (strings)

```
"Hi there, I am a string"  
'Hello world'  
"123 Oak Tree Lane"  
"(267)555-1234"  
"18901-3384"
```

Boleanas (boolean)

```
x = True
```

```
X = False
```

Operadores

Operadores Aritméticos

Operator	Description	Example
+	Addition	$1 + 1 = 2$
-	Subtraction	$10 - 1 = 9$
*	Multiplication	$3 * 5 = 15$
/	Division	$10 / 5 = 2$
%	Modulus (remainder after division)	$11 \% 5 = 1$
**	Exponent	$3 ** 2 = 9$
//	Floor division	$11 // 5 = 2$

Operadores de comparación

Operator	Meaning
<	Less than
<=	Less than or equal to
>	Greater than
>=	Greater than or equal to
==	Equal to
!=	Not equal to
is	Object identity
is not	Negated object identity

Operadores Booleanos

Operator	Code Example	What It Determines
or	<code>x or y</code>	Either x or y is true
and	<code>x and y</code>	Both x and y are true
not	<code>not x</code>	x is not full

Variables

Asignación de valores a variables

```
variablename = value
```

```
x = 10
```

```
user_name = "Alan"
```

Manejo de variables

```
hello.py  x
1  # This is a Python comment in my first Python app.
2  # This variable contains an integer
3  quantity = 10
4  # This variable contains a float
5  unit_price = 1.99
6  # This variable contains the result of multiplying quantity times unit price
7  extended_price = quantity * unit_price
8  # Show the results
9  print(extended_price)
10 |
```

Funciones integradas

Funciones integradas

- Python cuenta con algunas funciones ya integradas al lenguaje:

Built-In Function	Purpose
<code>abs(x)</code>	Returns the absolute value of number <i>x</i> (converts negative numbers to positive)
<code>bin(x)</code>	Returns a string representing the value of <i>x</i> converted to binary.
<code>float(x)</code>	Converts a string or number <i>x</i> to a the float data type
<code>format(x, y)</code>	Returns <i>x</i> formatted as directed by format string <i>y</i> . In modern Python you're more likely to use f-strings, as described later in this chapter
<code>hex(x)</code>	Returns a string containing <i>x</i> converted to hexadecimal, prefixed with 0x.
<code>int(x)</code>	Converts <i>x</i> to the integer data type by truncating (not rounding) the decimal point and any digits after it.
<code>max(x, y, z ...)</code>	Takes any number of numeric arguments and returns whichever one is the largest.
<code>min(x, y, z ...)</code>	Takes any number of numeric arguments and returns whichever one is the smallest.
<code>oct(x)</code>	Converts <i>x</i> to an octal number, prefixed with 0o to indicate octal.
<code>round(x, y)</code>	Rounds the number <i>x</i> to <i>y</i> number of decimal places.
<code>str(x)</code>	Converts number <i>x</i> to the string data type.
<code>type(x)</code>	Returns a string indicating the data type of <i>x</i> .

Toma de Decisiones

Toma de decisiones (if)

```
num = 10
if num > 0:
    print("Positive number")
else:
    print("Negative number")
```



Rogelio Ferreira Escutia

***Instituto Tecnológico de Morelia
Departamento de Sistemas y Computación***

***Correo: rogelio@itmorelia.edu.mx
 rogeplus@gmail.com***

***Página Web: http://sagitario.itmorelia.edu.mx/~rogelio/
 http://www.xumarhu.net/***

Twitter: http://twitter.com/rogeplus

Facebook: http://www.facebook.com/groups/xumarhu.net/