

Tarea

1) Descargar Git, crear una cuenta en algunos de los repositorios remotos de Git: Bitbucket, GitHub

2) Descargar Python 3.

Recomendaciones:

- Si usa linux o mac, hacerlo desde el gestor de paquetes usando pip. (Esto es si usted tiene experiencia en el manejo de la terminal).
- Si no quiere complicarse una buena opción: Anaconda.

3) Crear un repositorio en Git

Repositorio del Curso

El contenido del curso se irá actualizando en el siguiente repositorio de GitHub

https://github.com/jsepulveda9714/curso_PythonBasico.git



Clone



HTTPS SSH GitHub CLI

`https://github.com/jsepulveda9714/curs`



Use Git or checkout with SVN using the web URL.

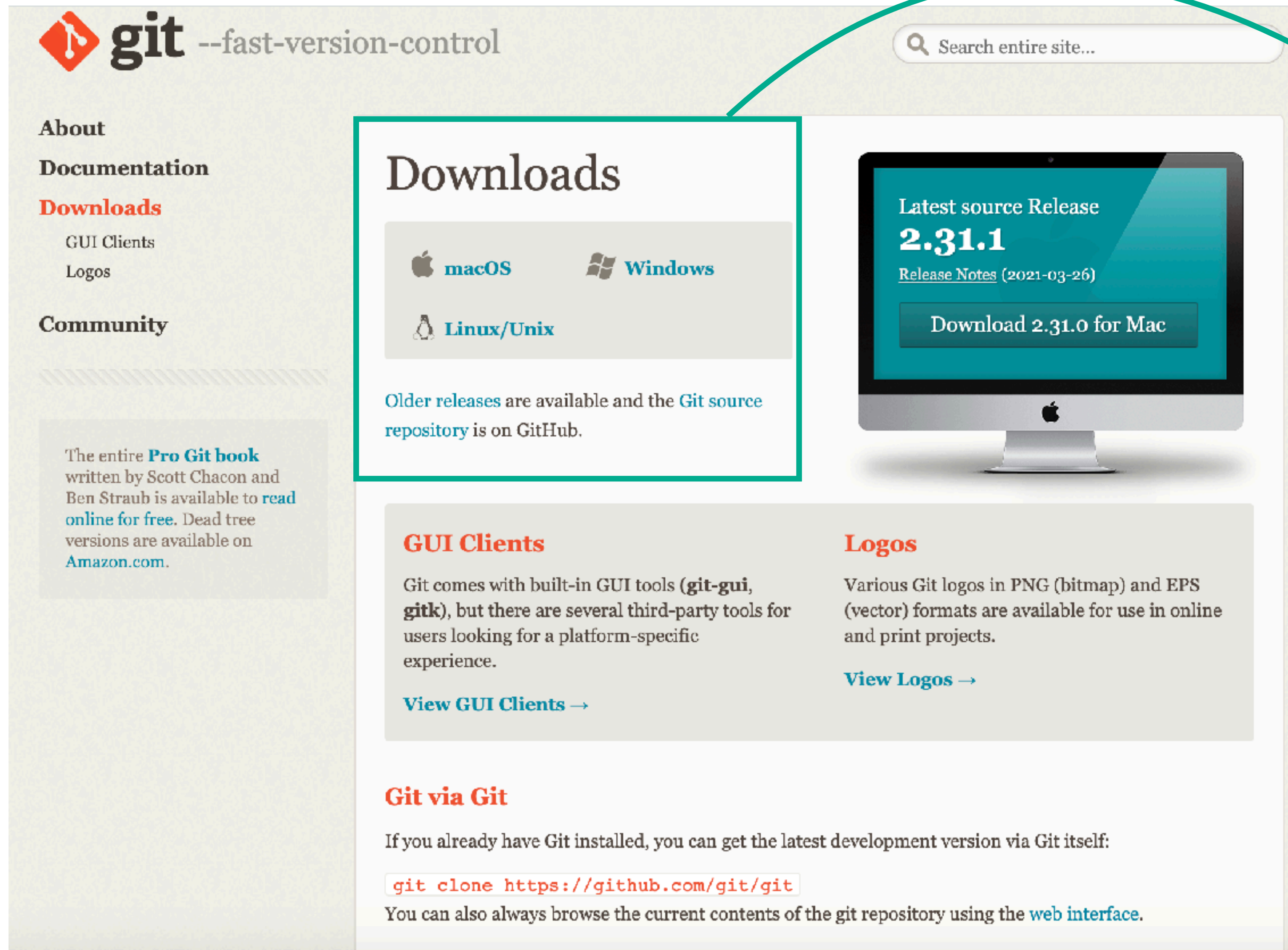


Open with GitHub Desktop



Download ZIP

Descargar Git para el OS de preferencia



git --fast-version-control

Search entire site...

About
Documentation
Downloads
GUI Clients
Logos
Community

The entire **Pro Git book** written by Scott Chacon and Ben Straub is available to [read online for free](#). Dead tree versions are available on [Amazon.com](#).

Downloads

macOS Windows Linux/Unix

Older releases are available and the [Git source repository](#) is on GitHub.

Latest source Release
2.31.1
[Release Notes](#) (2021-03-26)
Download 2.31.0 for Mac

GUI Clients

Git comes with built-in GUI tools (**git-gui**, **gitk**), but there are several third-party tools for users looking for a platform-specific experience.
[View GUI Clients →](#)

Logos

Various Git logos in PNG (bitmap) and EPS (vector) formats are available for use in online and print projects.
[View Logos →](#)

Git via Git

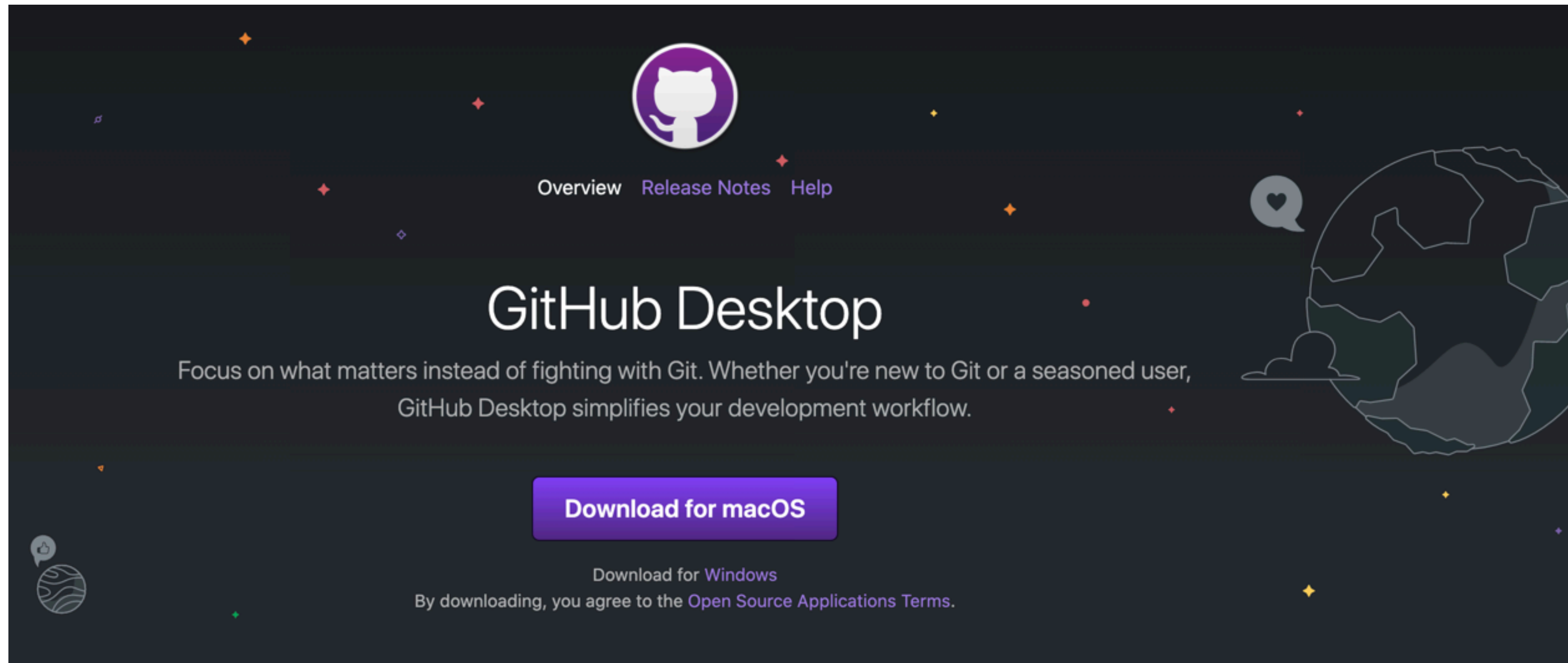
If you already have Git installed, you can get the latest development version via Git itself:

```
git clone https://github.com/git/git
```

You can also always browse the current contents of the git repository using the [web interface](#).

<https://git-scm.com/downloads>

Gestor de escritorio



Para evitar algunas complicaciones con el uso de comandos es recomendable una interfaz de escritorio.

<https://desktop.github.com/>

Inicio Rápido en Python

The screenshot displays the Google Colaboratory web interface. At the top, the browser address bar shows the URL `colab.research.google.com/notebooks/intro.ipynb?utm_source=scs-index`. The Colab logo and 'Welcome To Colaboratory' text are on the left, with a menu bar (File, Edit, View, Insert, Runtime, Tools, Help) below it. On the right, there are links for 'Share', 'Settings', and a 'Tiger' icon, along with an 'Update' button. A left sidebar contains a 'Table of contents' with links to 'Getting started', 'Data science', 'Machine learning', 'More Resources', 'Machine Learning Examples', and a 'Section' button. The main content area is titled 'What is Colaboratory?' and includes a list of features: 'Zero configuration required', 'Free access to GPUs', and 'Easy sharing'. It also contains a paragraph about Colab's benefits for students, data scientists, and AI researchers, with a link to 'Introduction to Colab'. A section titled 'Getting started' explains that the document is an interactive 'Colab notebook' and provides an example of a code cell with a Python script to calculate the number of seconds in a day, resulting in the output '86400'.

← → ↻ colab.research.google.com/notebooks/intro.ipynb?utm_source=scs-index 🔍 ☆ 🛑 🟢 🟡 ⚙️ 🦊 Update ⋮

co Welcome To Colaboratory
File Edit View Insert Runtime Tools Help

🔗 Share ⚙️ 🦊

☰ Table of contents ✕

- 🔍 Getting started
- 🔍 Data science
- 🔍 Machine learning
- 📁 More Resources
- 📁 Machine Learning Examples
- ➕ Section

+ Code + Text 📄 Copy to Drive

Connect ▾ | Editing | ^

co What is Colaboratory?

Colaboratory, or "Colab" for short, allows you to write and execute Python in your browser, with

- Zero configuration required
- Free access to GPUs
- Easy sharing

Whether you're a **student**, a **data scientist** or an **AI researcher**, Colab can make your work easier. Watch [Introduction to Colab](#) to learn more, or just get started below!

▼ Getting started

The document you are reading is not a static web page, but an interactive environment called a **Colab notebook** that lets you write and execute code.

For example, here is a **code cell** with a short Python script that computes a value, stores it in a variable, and prints the result:

```
[ ] seconds_in_a_day = 24 * 60 * 60
seconds_in_a_day
```

86400

Anaconda: Un pre-compilado muy completo

[Products ▾](#)[Pricing](#)[Solutions ▾](#)[Resources ▾](#)[Blog](#)[Company ▾](#)[Get Started](#)

Individual Edition

Your data science toolkit

With over 25 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries.

[Download](#)

Windows

Python 3.8

64-Bit Graphical Installer (477 MB)

32-Bit Graphical Installer (409 MB)

MacOS

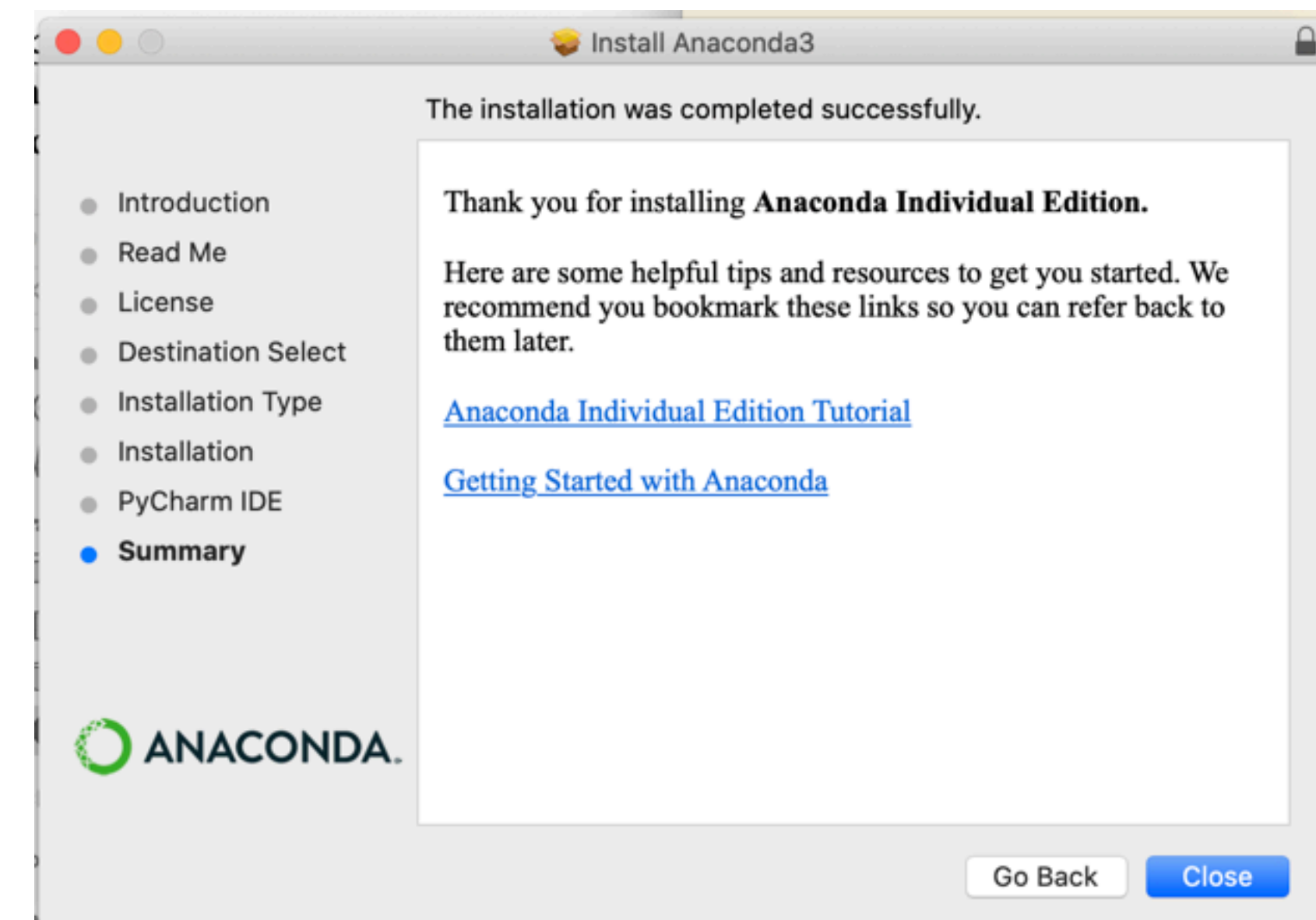
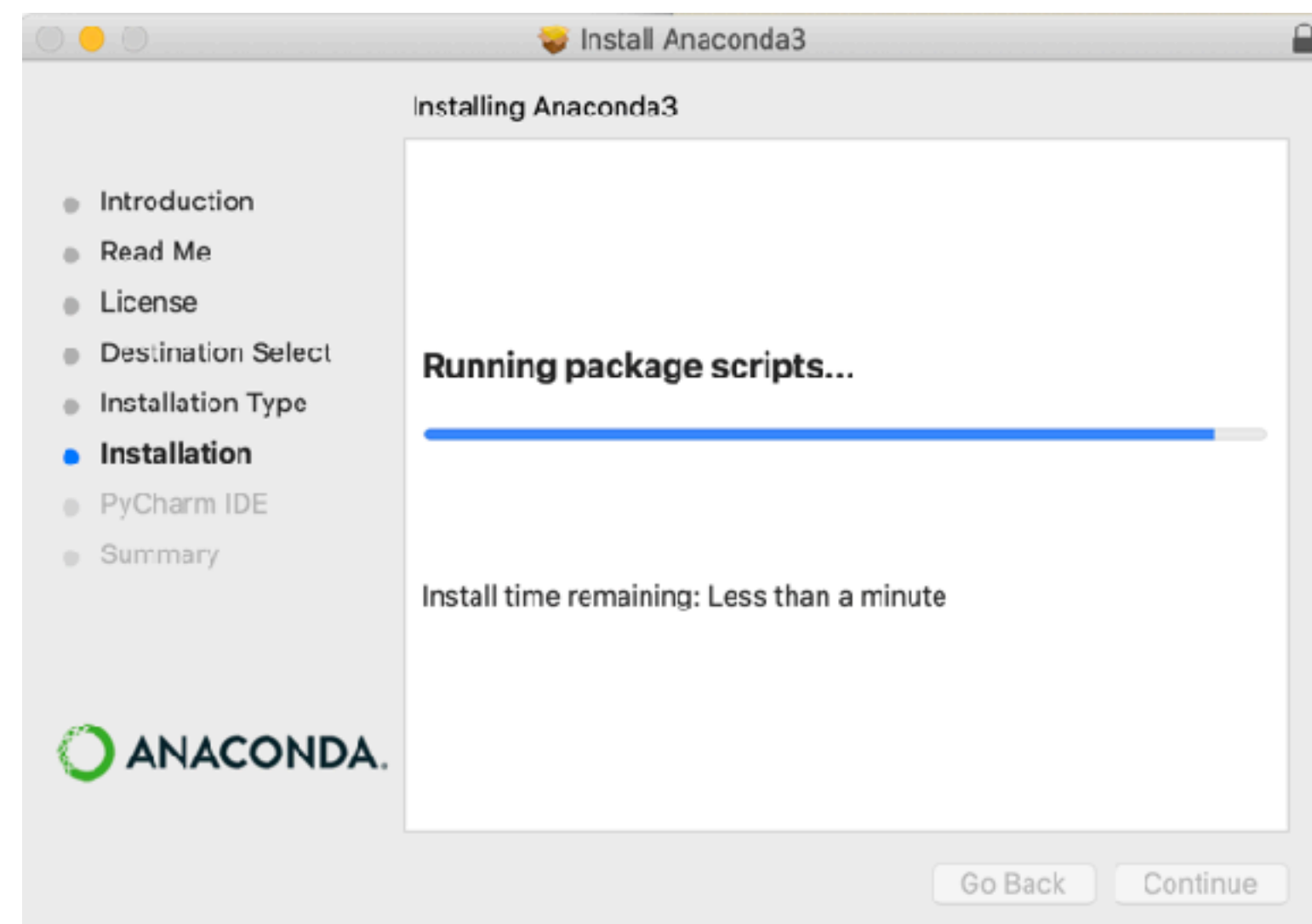
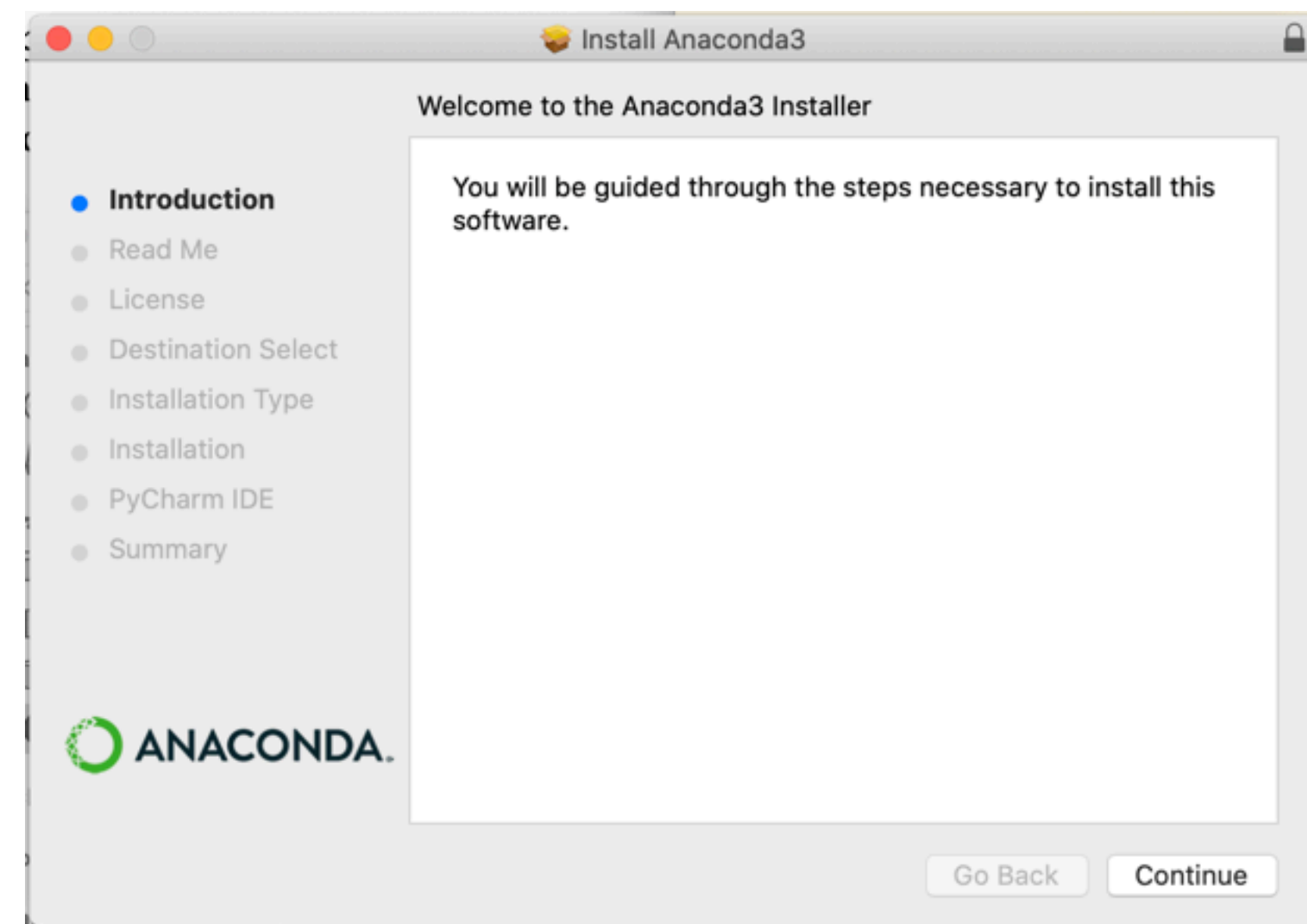
Python 3.8

64-Bit Graphical Installer (440 MB)

64-Bit Command Line Installer (433 MB)

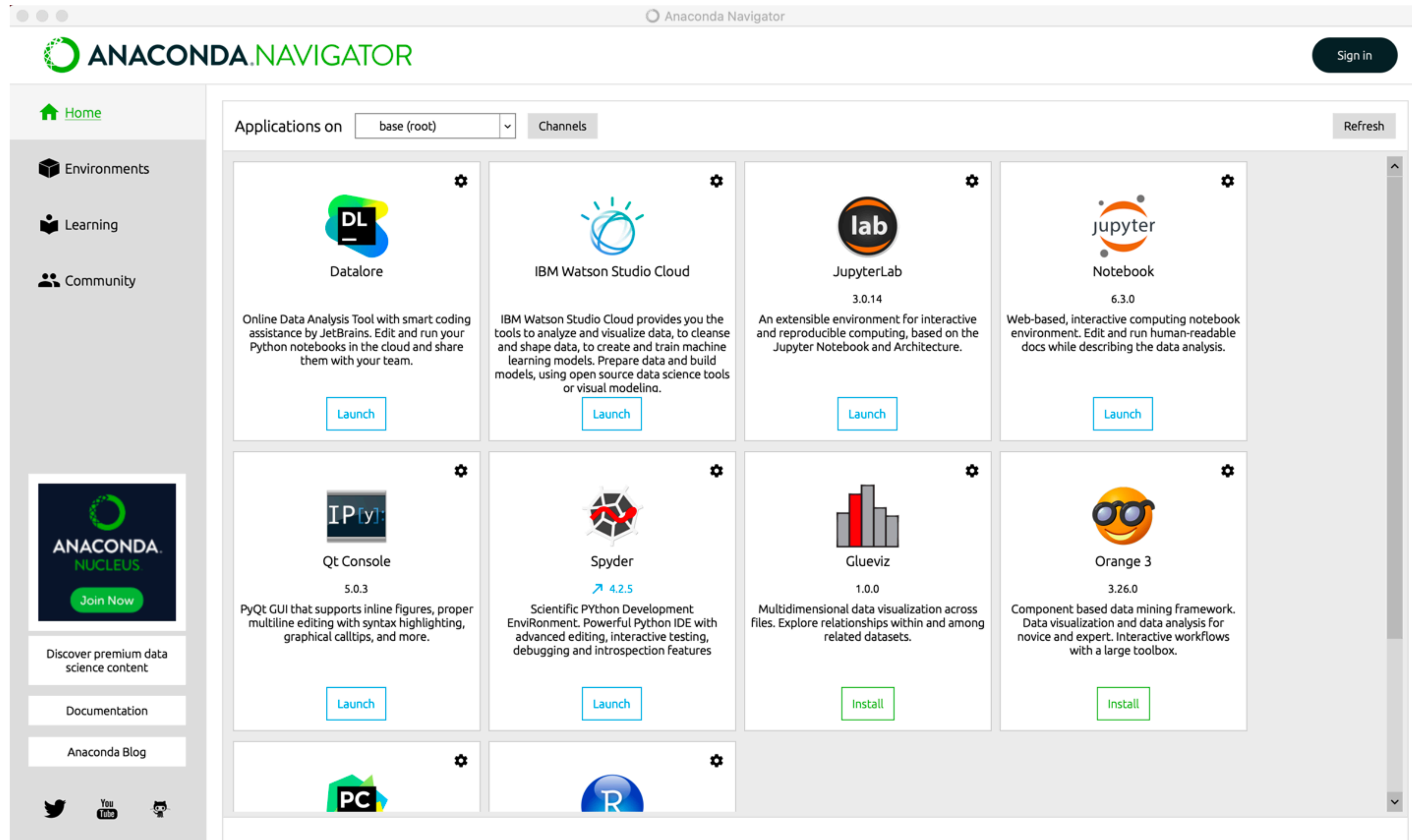
<https://www.anaconda.com/products/individual>

Proceso de instalación



En solamente es descargar la
versión continuar ... continuar si
a todo!

¿Que debería ver cuándo abra Anaconda?



El más usado en este curso: Jupyter

The image shows the Jupyter Notebook logo, which consists of an orange circle with three smaller grey circles around it, and the word "jupyter" in a dark grey sans-serif font. Below the logo, the word "Notebook" is written in a larger, bold, dark grey sans-serif font. Underneath that, the version number "6.3.0" is displayed in a smaller, dark grey sans-serif font. A paragraph of text follows: "Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis." At the bottom, there is a blue rectangular button with the word "Launch" in a white sans-serif font.

Jupyter Fourier_teorico Last Checkpoint: 08/15/2019 (unsaved changes)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

+

Run

Code

Transformada de Fourier en python

```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import math
import random

In [2]: def myfft(serie, time_step):
    size_f = len(serie)
    fourier = np.fft.fft(serie)
    power_spec = np.abs(fourier) ** 2.0
    amp = np.abs(fourier)
    freq = np.fft.fftfreq(size_f, time_step)
    period = 1./freq
    return fourier, power_spec, amp, freq, period

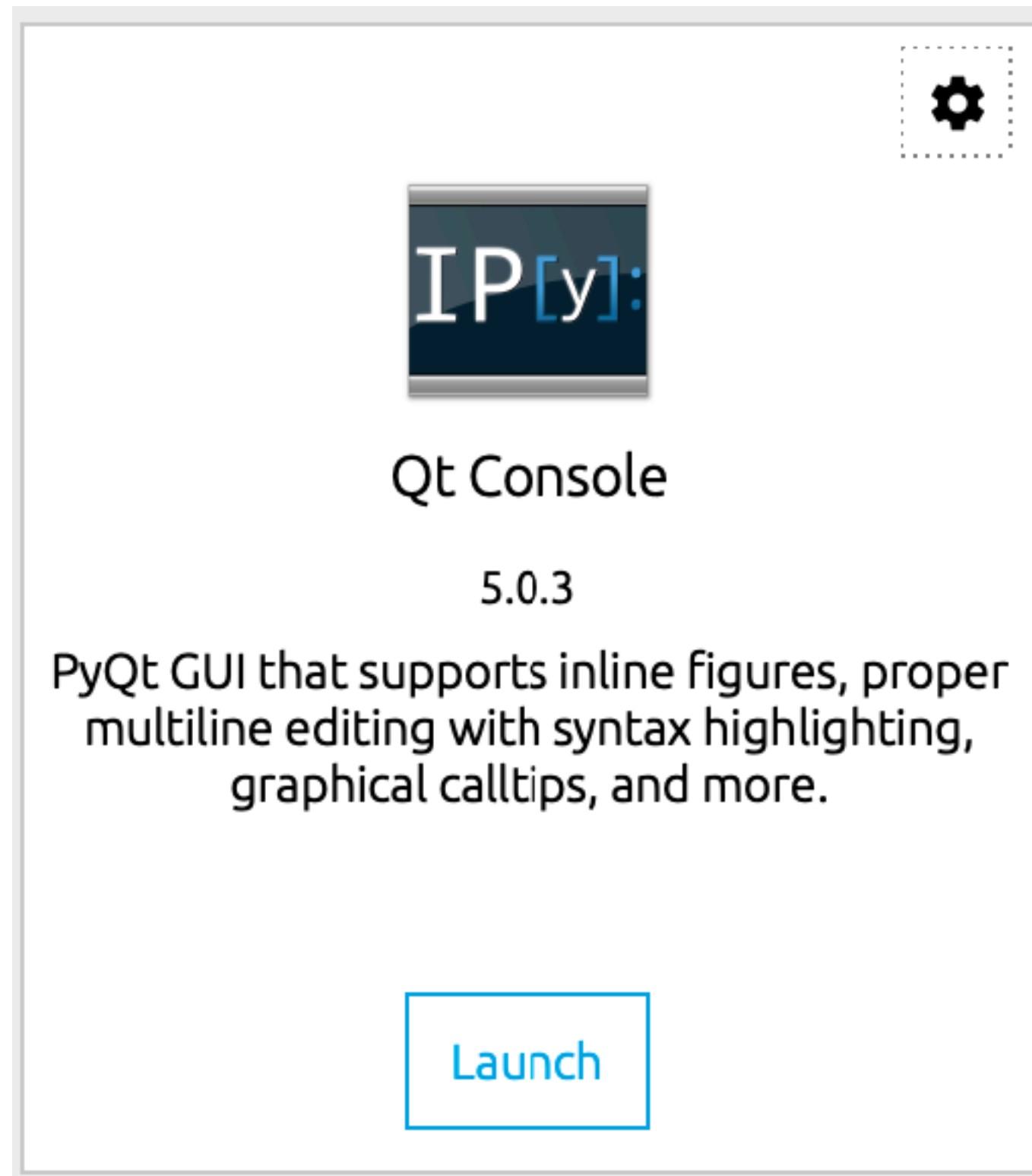
In [3]: time_step = 0.005
t = np.arange(0,10,time_step)
fun = np.sin(2*np.pi*t) + np.sin(2*2*np.pi*t)\
      + np.sin(4*2*np.pi*t)

fourier,power_spec,amp,freq,period = myfft(fun, time_step)

<ipython-input-2-8915982c9319>:7: RuntimeWarning: divide by zero encountered in true_divide
    period = 1./freq

In [4]: plt.close()
plt.cla()
plt.clf()
plt.xlim(-10.,10.)
plt.plot(freq, power_spec)
plt.show()
```

El interprete: Ipython



```
Jupyter QtConsole
Jupyter QtConsole 5.0.3
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.22.0 -- An enhanced Interactive Python. Type '?' for help.

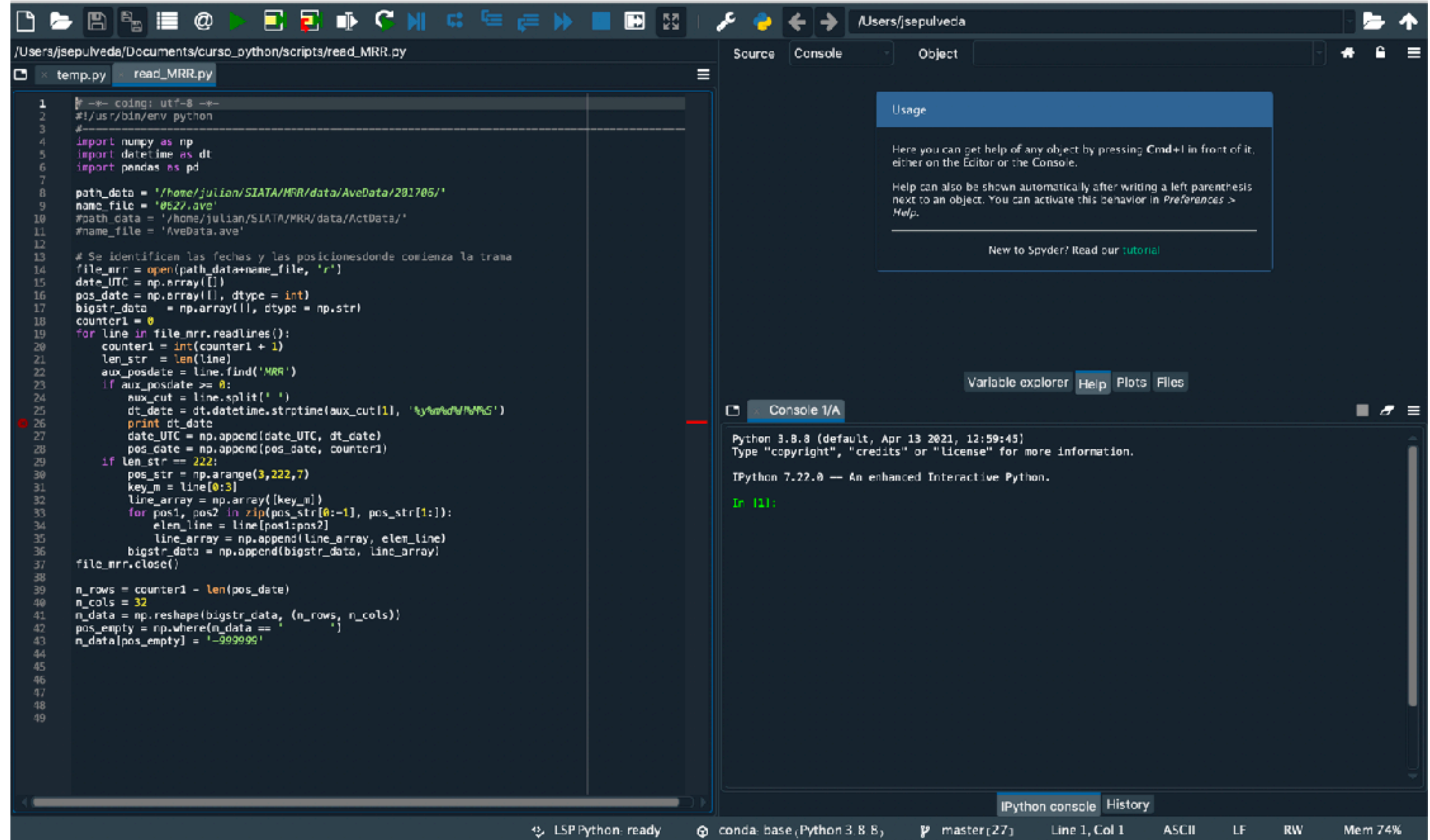
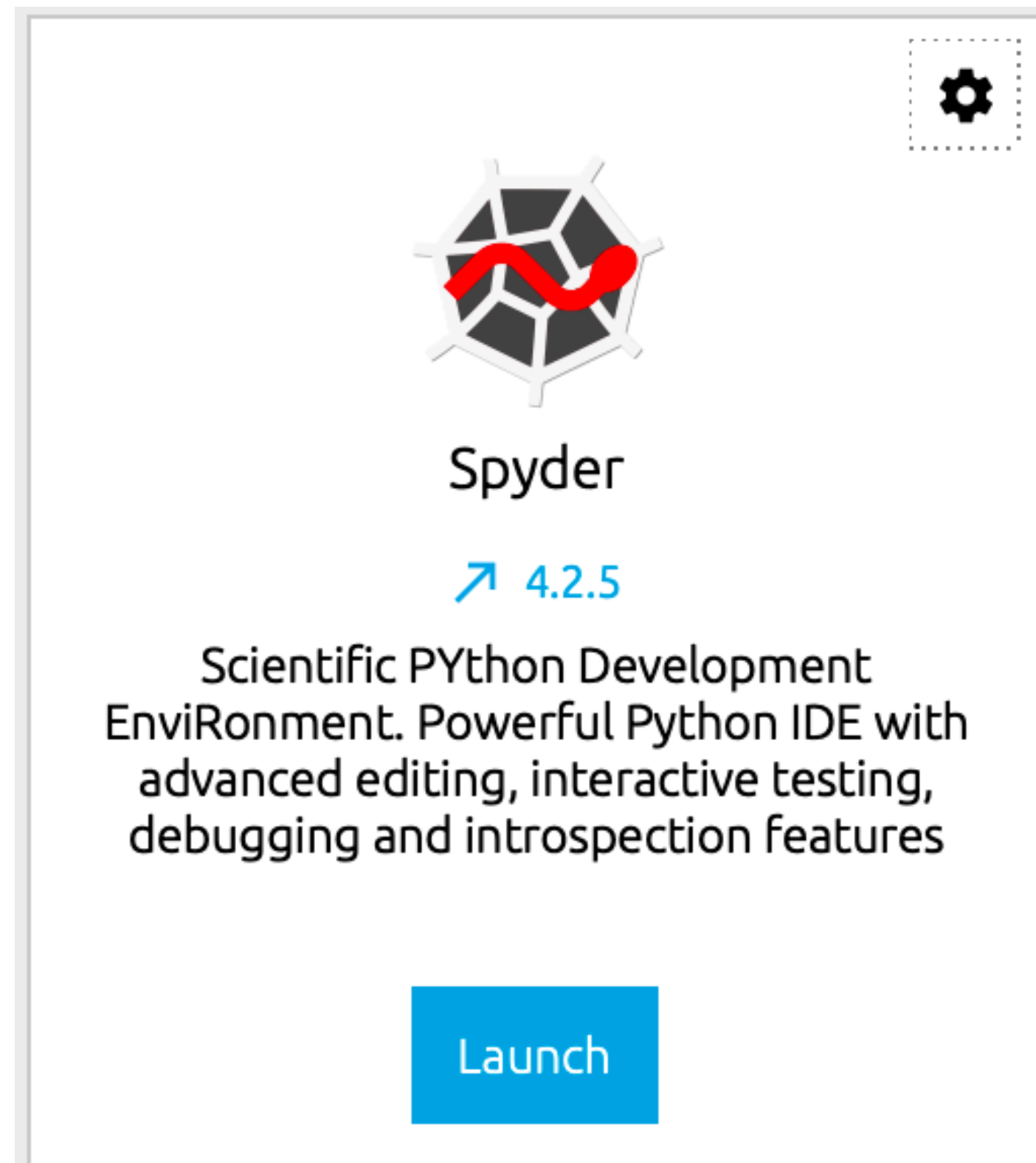
In [1]: pwd
Out[1]: '/Users/jsepulveda'

In [2]: a = 16.


In [3]: print (a)
16.0

In [4]: |
```


Agradable y completo: Spyder



Tutoriales al interior de Anaconda

 **ANACONDA**.NAVIGATOR


Sign in


Home


Environments


Learning


Community

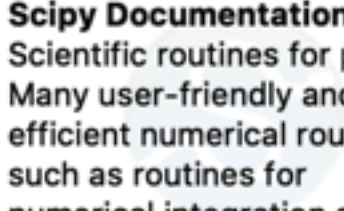

Python Tutorial
Read



Python Reference
Read



Anaconda Package List
Read



Pandas Documentation
Read



Numpy Documentation
Read



Scipy Documentation
Scientific routines for python. Many user-friendly and efficient numerical routines such as routines for numerical integration and optimization.
Scipy Documentation
Read



Matplotlib Documentation
Read



Bokeh User Guide
Read



Anaconda Cloud Documentation
Read



Anaconda Documentation
Read


Anaconda Navigator Documentation
Read



The Comprehensive R Archive Network (CRAN)
Read


The Python Package Index (PyPI)
Read


Dask documentation
Read


Conda & Conda-Build
Read

Instalación de paquetes



Sign in

Home

Environments

Learning

Community

Search Environments

base (root)

All

Channels

Update index...

netcdf

Name	T	Description	Version
<input type="checkbox"/> cftime		Time-handling functionality from netcdf4-python	1.4.1
<input type="checkbox"/> libnetcdf		Libraries and data formats that support array-oriented scientific data.	4.7.3
<input type="checkbox"/> netcdf4		Provides an object-oriented python interface to the netcdf version 4 library.	1.5.6
<input type="checkbox"/> pynio		Pynio is a multi-format data i/o package with a netcdf-style interface.	1.5.0
<input type="checkbox"/> r-rnetcdf			1.9_1

Old School

```
scripts — vim ozono.py — 103x56
1 # -*- coding: utf-8 -*-
2
3 import pandas as pd
4 import numpy as np
5 import matplotlib.pyplot as plt
6 from pandas import datetime
7 from matplotlib import ticker
8 import datetime as dt
9
10 #Funcion para leer excel de variables meteorologicas
11 def creardf_m(excelfilename,sheetname):
12     df = pd.read_excel(path,sheetname = sheetname, usecols=[0,24,26,30,32,34], skipfooter = 0,
13         index_col=0, parse_dates=[0])
14     df.index.name = 'Fecha'
15     return df
16
17 #Leer excel meteorologico
18 path = "3_2017v1.xlsx"
19 sheetname = "Consolidado"
20 df = creardf_m(path, sheetname = sheetname)
21
22 #Funcion para leer excel ozono
23 def creardf_c(excelfilename,sheetname):
24     df_c = pd.read_excel(path,sheetname = sheetname, usecols=[0,14], skipfooter = 0, index_col=0,
25         parse_dates=[0])
26     df_c.index.name = 'Fecha'
27     return df_c
28
29 #Leer el ozono
30 df_c = creardf_c(path, sheetname = sheetname)
31
32 #pasar a microgramo/metro cubico
33 df_c['ozono'] = df_c['ozono'].multiply(47.997/24.45)
34
35 #crear series horarias y mensuales
36 O3_3diario = df_c.resample("d").mean()
37 O3_3mes = df_c.resample("M").mean()
38
39 #Quita los valores nan de la serie, no se puede trabajar con nan para estandarizar una serie
40 df_csnan=df_c.dropna(0)
41
42 #grafico sencillo
43 ax = O3_3mes['ozono'].plot(legend=True, figsize=(9,5))
44 ax.set_ylabel(u'Concentración de Ozono [μg/m³]',fontsize=12)
45 ax.set_xlabel('Meses',fontsize=12)
46 ax.grid()
47
48 #Estandarización de la serie diaria
49 def hourstand(serie): #ingresa dataframe
50     serie['stand'] = np.ones(len(serie)) * -999.
51     mean_array = np.empty((24))
52     std_arr = np.empty((24))
53     for i in np.arange(0,24):
54         dumm = serie.loc[serie.index.hour == i]
55         mean_h = np.nanmean(dumm.ozono.values)
56         std_h = np.nanstd(dumm.ozono.values)
57         mean_array[i] = mean_h
58         std_arr[i] = std_h
59
60 ozono.py
61 "ozono.py" [dos] 971 31900
```

```
scripts — ipython — 100x54
~/Documents/curso_python/scripts — ipython
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.22.0 -- An enhanced Interactive Python. Type '?' for help.

[In [1]: exit
[(base) jsepulveda@julians-MacBook-Pro-2 ~ % cd /Users/jsepulveda/Documents/curso_python/scripts
[(base) jsepulveda@julians-MacBook-Pro-2 scripts % ls
ozono.py          read_MRR_improve.py  read_ascii.py
read_MRR.py       read_MRR_improveV2.py read_nc_wrf.py
[(base) jsepulveda@julians-MacBook-Pro-2 scripts % ipython
Python 3.8.8 (default, Apr 13 2021, 12:59:45)
Type 'copyright', 'credits' or 'license' for more information
IPython 7.22.0 -- An enhanced Interactive Python. Type '?' for help.

[In [1]: run ozono.py
/Users/jsepulveda/Documents/curso_python/scripts/ozono.py:6: FutureWarning: The pandas.datetime class
is deprecated and will be removed from pandas in a future version. Import from datetime module instead.
  from pandas import datetime

-----
TypeError                                Traceback (most recent call last)
~/Documents/curso_python/scripts/ozono.py in <module>
    16 path = "3_2017v1.xlsx"
    17 sheetname = "Consolidado"
--> 18 df = creardf_m(path, sheetname = sheetname)
    19
    20 #Funcion para leer excel ozono

~/Documents/curso_python/scripts/ozono.py in creardf_m(excelfilename, sheetname)
    10 #Funcion para leer excel de variables meteorologicas
    11 def creardf_m(excelfilename,sheetname):
--> 12     df = pd.read_excel(path,sheetname = sheetname, usecols=[0,24,26,30,32,34], skipfooter =
0, index_col=0, parse_dates=[0])
    13     df.index.name = 'Fecha'
    14     return df

~/opt/anaconda3/lib/python3.8/site-packages/pandas/util/_decorators.py in wrapper(*args, **kwargs)
    297     )
    298     warnings.warn(msg, FutureWarning, stacklevel=stacklevel)
--> 299     return func(*args, **kwargs)
    300
    301     return wrapper

TypeError: read_excel() got an unexpected keyword argument 'sheetname'

In [2]:
```


La verdadera esencia de la programación

