

Python Standard Library & List of Important Libraries



Contents

What is the Python Libraries?	3
Python Standard Library	3
Important Python Libraries	4
a. Matplotlib.....	4
b. Pandas	4
c. Requests	5
d. NumPy	6
e. SQLAlchemy	6
f. BeautifulSoup.....	7
g. Pyglet	8
h. SciPy	9
i. Scrapy	9
j. PyGame	10
k. Python Twisted	11
l. Pillow.....	11
m. pywin32	12
n. wxPython	13
o. iPython.....	14
p. Nose	15
q. Flask.....	16
r. SymPy	16
s. Fabric	17
t. PyGTK.....	17
Conclusion	18



What is the Python Libraries?

We know that a module is a file with some Python code, and a package is a directory for sub packages and modules. But the line between a package and a **Python library** is quite blurred.

A Python library is a reusable chunk of code that you may want to include in your programs/ projects. Compared to languages like C++ or C, a Python libraries do not pertain to any specific context in Python. Here, a ‘library’ loosely describes a collection of core modules. Essentially, then, a library is a collection of modules. A package is a library that can be installed using a package manager like rubygems or npm.

Learn: [A Comprehensive Guide on Python Packages](#)

Python Standard Library

The Python Standard Library is a collection of exact syntax, token, and semantics of Python. It comes bundled with core Python distribution. We mentioned this when we began with an introduction.

It is written in C, and handles functionality like I/O and other core modules. All this functionality together makes Python the language it is. More than 200 core modules

sit at the heart of the standard library. This library ships with Python. But in addition to this library, you can also access a growing collection of several thousand components from the Python Package Index (PyPI). We mentioned it in the previous blog.

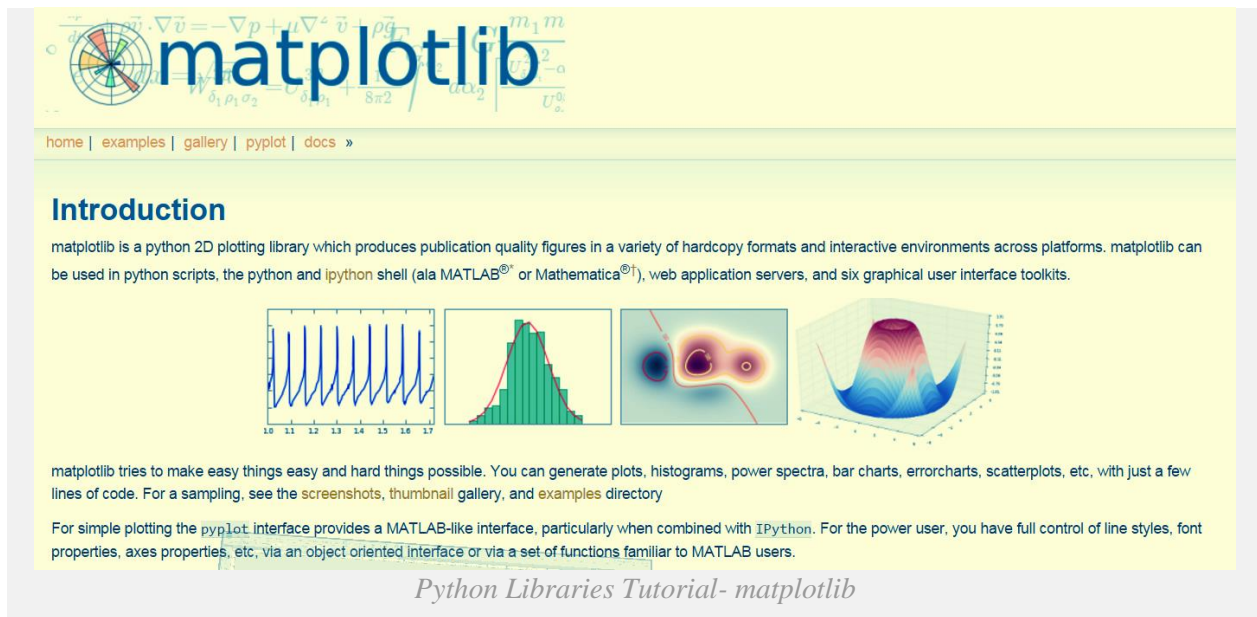
Learn: [Python Tuples vs Lists – Comparison between Lists and Tuples](#)

Important Python Libraries

Next, we will see twenty Python libraries list that will take you places in your journey with Python. These are also the Python libraries for Data Science.

a. Matplotlib

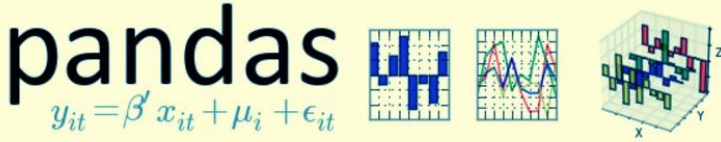
Matplotlib helps with data analyzing, and is a numerical plotting library. We talked about it in [Python for Data Science](#).



The screenshot shows the Matplotlib website. At the top, there's a navigation bar with links: [home](#) | [examples](#) | [gallery](#) | [pyplot](#) | [docs](#) ». Below this is the 'Introduction' section. It states: 'matplotlib is a python 2D plotting library which produces publication quality figures in a variety of hardcopy formats and interactive environments across platforms. matplotlib can be used in python scripts, the python and [ipython](#) shell (ala MATLAB[®] or Mathematica[®]), web application servers, and six graphical user interface toolkits.' Below the text are four small plots: a line plot with multiple overlapping lines, a histogram with a red normal distribution curve, a contour plot with two peaks, and a 3D surface plot. Further down, it says: 'matplotlib tries to make easy things easy and hard things possible. You can generate plots, histograms, power spectra, bar charts, errorcharts, scatterplots, etc, with just a few lines of code. For a sampling, see the [screenshots](#), [thumbnail gallery](#), and [examples](#) directory'. It also mentions: 'For simple plotting the [pyplot](#) interface provides a MATLAB-like interface, particularly when combined with [IPython](#). For the power user, you have full control of line styles, font properties, axes properties, etc, via an object oriented interface or via a set of functions familiar to MATLAB users.' At the bottom, it says 'Python Libraries Tutorial- matplotlib'.

b. Pandas

Like we've said before, Pandas is a must for data-science. It provides fast, expressive, and flexible data structures to easily (and intuitively) work with structured (tabular, multidimensional, potentially heterogeneous) and time-series data.



[overview](#) // [get pandas](#) // [documentation](#) // [community](#) // [talks](#)

Python Data Analysis Library

pandas is an open source, BSD-licensed library providing high-performance, easy-to-use data structures and data analysis tools for the [Python](#) programming language.

VERSIONS


Release
0.17.0 - October 2015


[download](#) // [docs](#) // [pdf](#)

Python Libraries Tutorial – Pandas


c. Requests

Requests is a Python Library that lets you send HTTP/1.1 requests, add headers, form data, multipart files, and parameters with simple Python dictionaries. It also lets you access the response data in the same way.



 **15,782**

Requests is an elegant and simple HTTP library for Python, built for human beings.



Get Updates

Receive updates on new releases and upcoming projects.

Requests: HTTP for Humans

Release v2.8.1. ([Installation](#))

Requests is an [Apache2 Licensed](#) HTTP library, written in Python, for human beings.

Python's standard [urllib2](#) module provides most of the HTTP capabilities you need, but the API is thoroughly **broken**. It was built for a different time — and a different web. It requires an *enormous* amount of work (even method overrides) to perform the simplest of tasks.

Things shouldn't be this way. Not in Python.

```

>>> r = requests.get('https://api.github.com/user', auth=('user', 'pass'))
>>> r.status_code
200
>>> r.headers['content-type']
'application/json; charset=utf8'
>>> r.encoding
'utf-8'
>>> r.text
u'{"type": "User"...'}
>>> r.json()
{'u'private_gists': 419, u'total_private_repos': 77, ...}


```

Python Libraries Tutorial- Requests

Learn: [How to Install Python on Windows](#)

d. NumPy

It has advanced math functions and a rudimentary scientific computing package.



NumPy

Scipy.org

NumPy

NumPy is the fundamental package for scientific computing with Python. It contains among other things:

- a powerful N-dimensional array object
- sophisticated (broadcasting) functions
- tools for integrating C/C++ and Fortran code
- useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

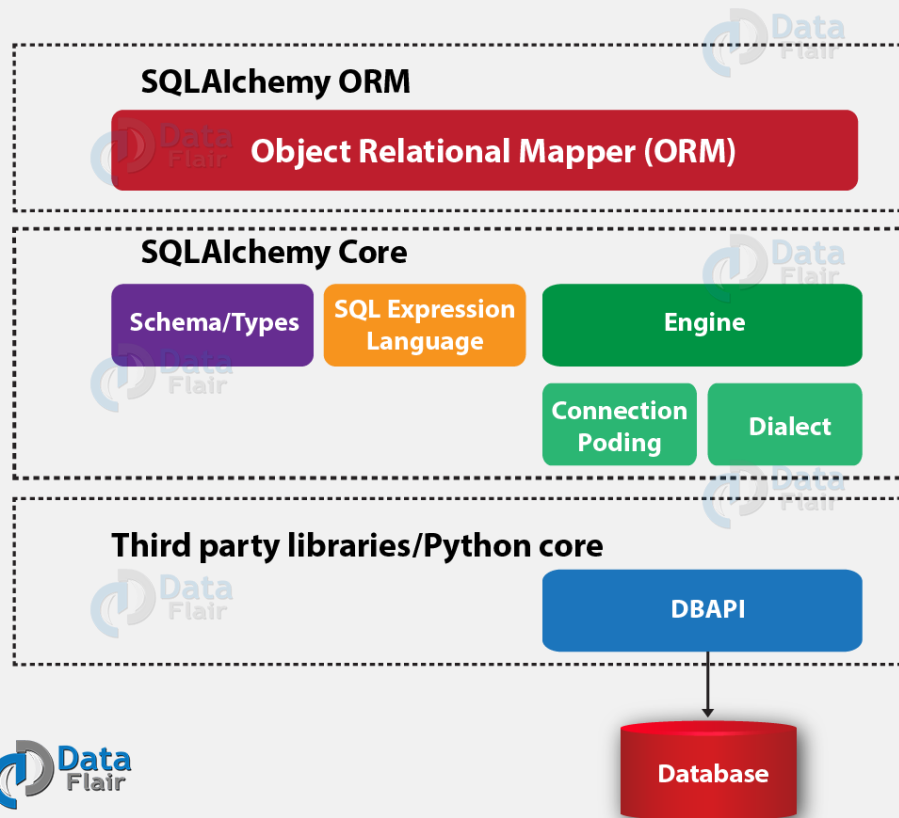
NumPy is licensed under the [BSD license](#), enabling reuse with few restrictions.

Python Libraries Tutorial – NumPy

e. SQLAlchemy

SQLAlchemy Overview

SQLAlchemy consists of the Core and the ORM



Python Libraries Tutorial – SQLAlchemy Overview

SQLAlchemy is a library with well-known enterprise-level patterns. It was designed for efficient and high-performing database-access.

f. BeautifulSoup

It may be a bit slow, BeautifulSoup has an excellent XML- and HTML- parsing library for beginners.

```
Terminal — bash — 80x24
Last login: Sun Oct 21 12:51:53 on ttys000
Gavin-Camerons-MacBook:~ gcameron$ python /users/gcameron/Desktop/map/colorize_svg.py
Traceback (most recent call last):
  File "/users/gcameron/Desktop/map/colorize_svg.py", line 4, in <module>
    from BeautifulSoup import BeautifulSoup
ImportError: No module named BeautifulSoup
Gavin-Camerons-MacBook:~ gcameron$
```

Python Libraries Tutorial – BeautifulSoup

g. Pyglet

Pyglet is an excellent choice for an object-oriented programming interface in developing games. In fact, it also finds use in developing other visually-rich applications for Mac OS X, Windows, and Linux. In the 90s, when people were bored, they resorted to playing Minecraft on their computers. Pyglet is the engine behind Minecraft.



pyglet: a cross-platform windowing and multimedia library for Python.

[home](#) | [download](#) | [documentation](#) | [contribute](#)

pyglet provides an object-oriented programming interface for developing games and other visually-rich applications for Windows, Mac OS X and Linux.

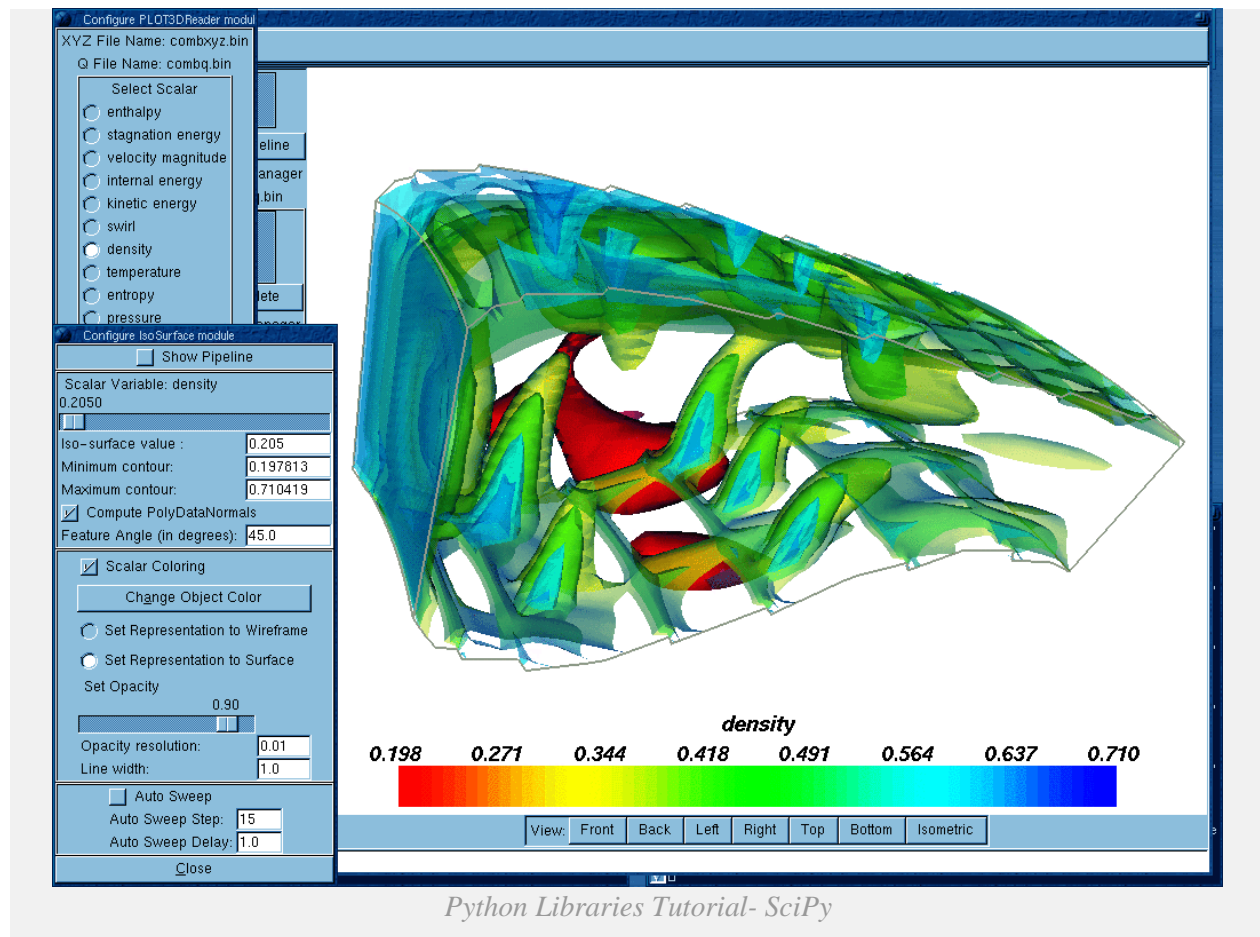
Some of the features of *pyglet* are:

- **No external dependencies or installation requirements.** For most application and game requirements, *pyglet* needs nothing else besides Python, simplifying distribution and installation.
- **Take advantage of multiple windows and multi-monitor desktops.** *pyglet* allows you to use as many windows as you need, and is fully aware of multi-monitor setups for use with fullscreen games.
- **Load images, sound, music and video in almost any format.** *pyglet* can optionally use AVbin to play back audio formats such as MP3, OGG/vorbis and WMA, and video formats such as DivX, MPEG-2, H.264, WMV and Xvid.
- **pyglet is provided under the BSD open-source license**, allowing you to use it for both commercial and other open-source projects with very little restriction.

Please join us on the mailing list, or download the SDK today!

h. SciPy

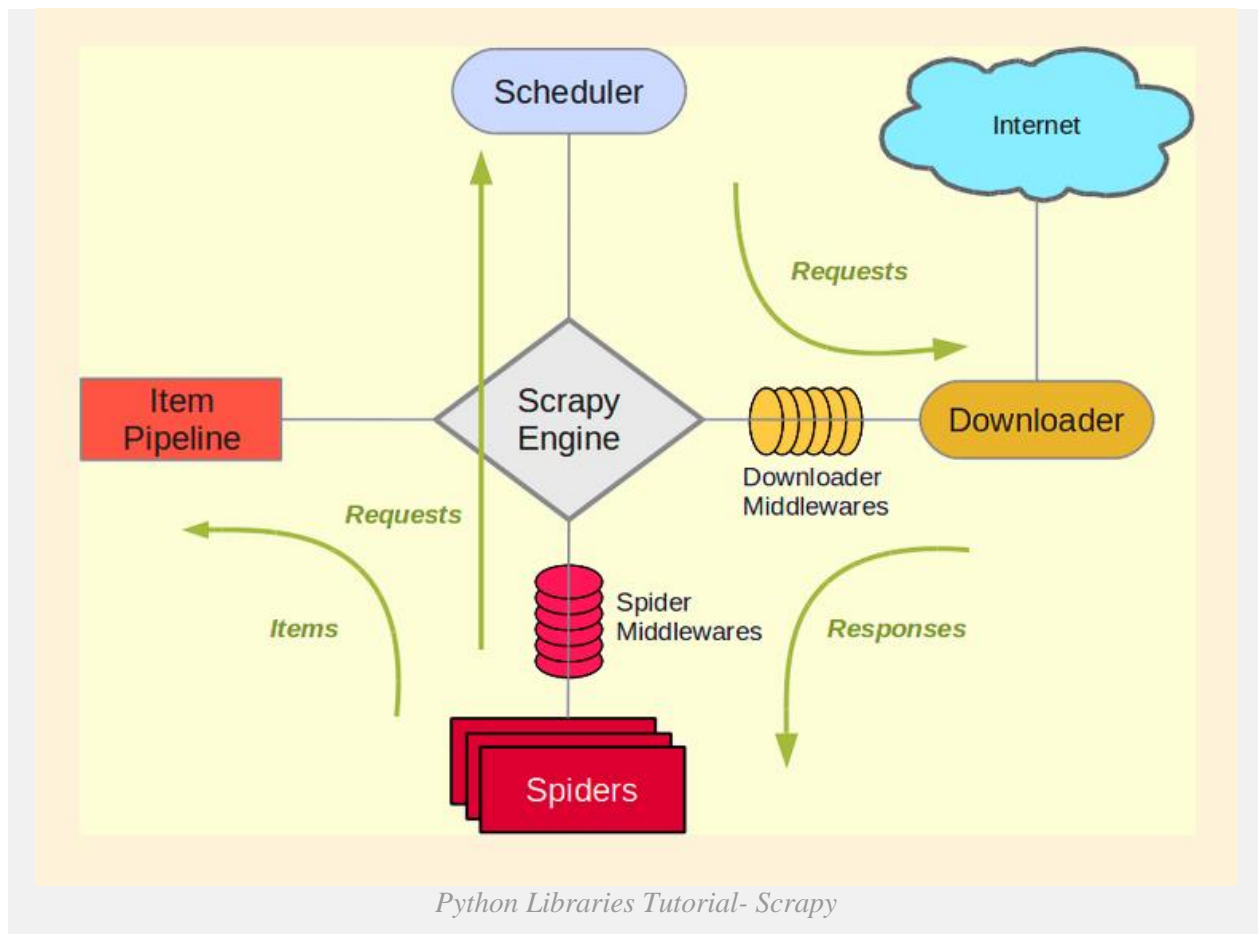
Next up is SciPy, one of the libraries we have been talking so much about. It has a number of user-friendly and efficient numerical routines. These include routines for optimization and numerical integration.



Learn: [7 Reasons Why Should I Learn Python in 2018](#)

i. Scrapy

If your motive is fast, high-level screen scraping and web crawling, go for Scrapy. You can use it for purposes from data mining to monitoring and automated testing.



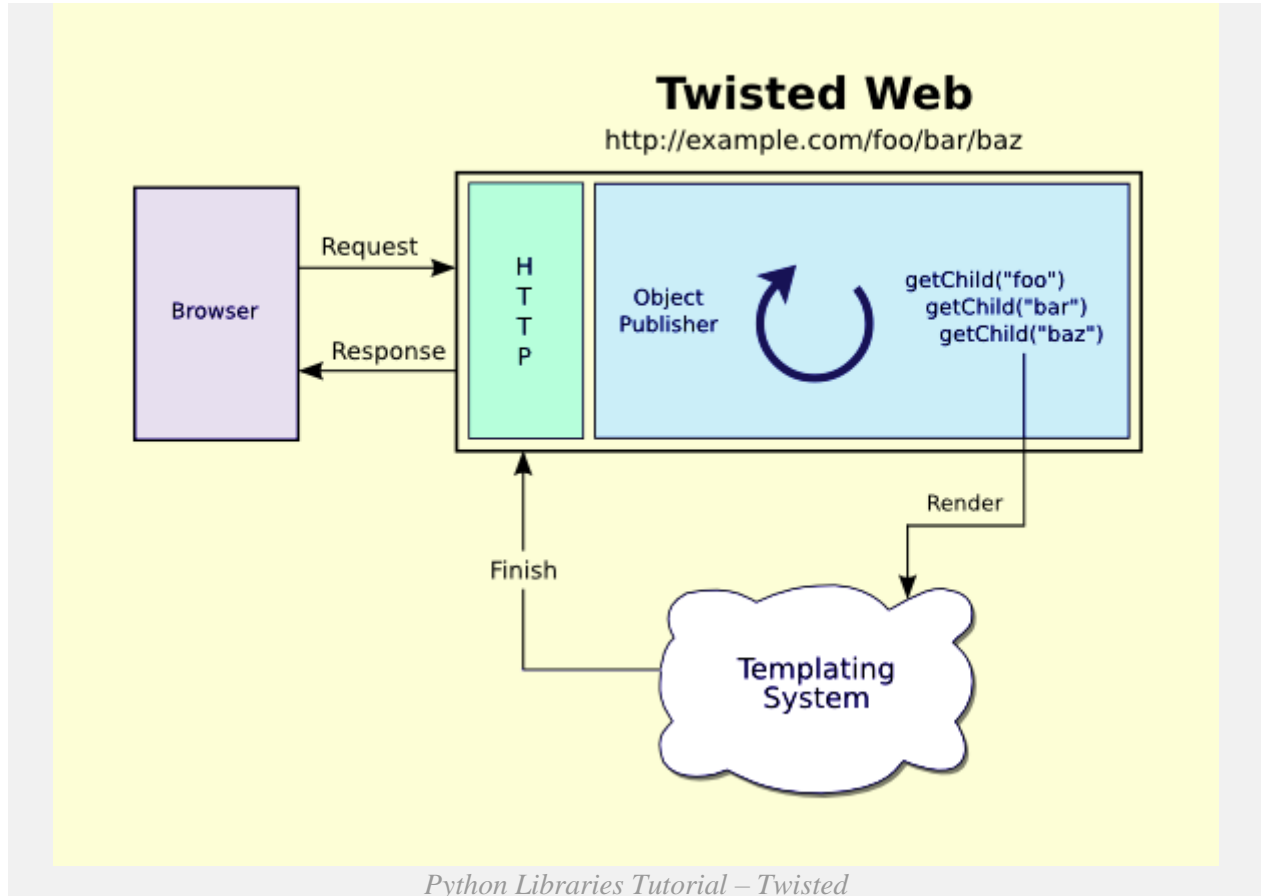
j. PyGame

PyGame provides an extremely easy interface to the Simple Directmedia Library (SDL) platform-independent graphic, audio, and input libraries.



k. Python Twisted

An event-driven networking engine, Twisted is written in Python, and licensed under the open-source MIT license.



l. Pillow

Pillow is a friendly fork of PIL (Python Imaging Library), but is more user-friendly. If you work with images, Pillow is your best friend.

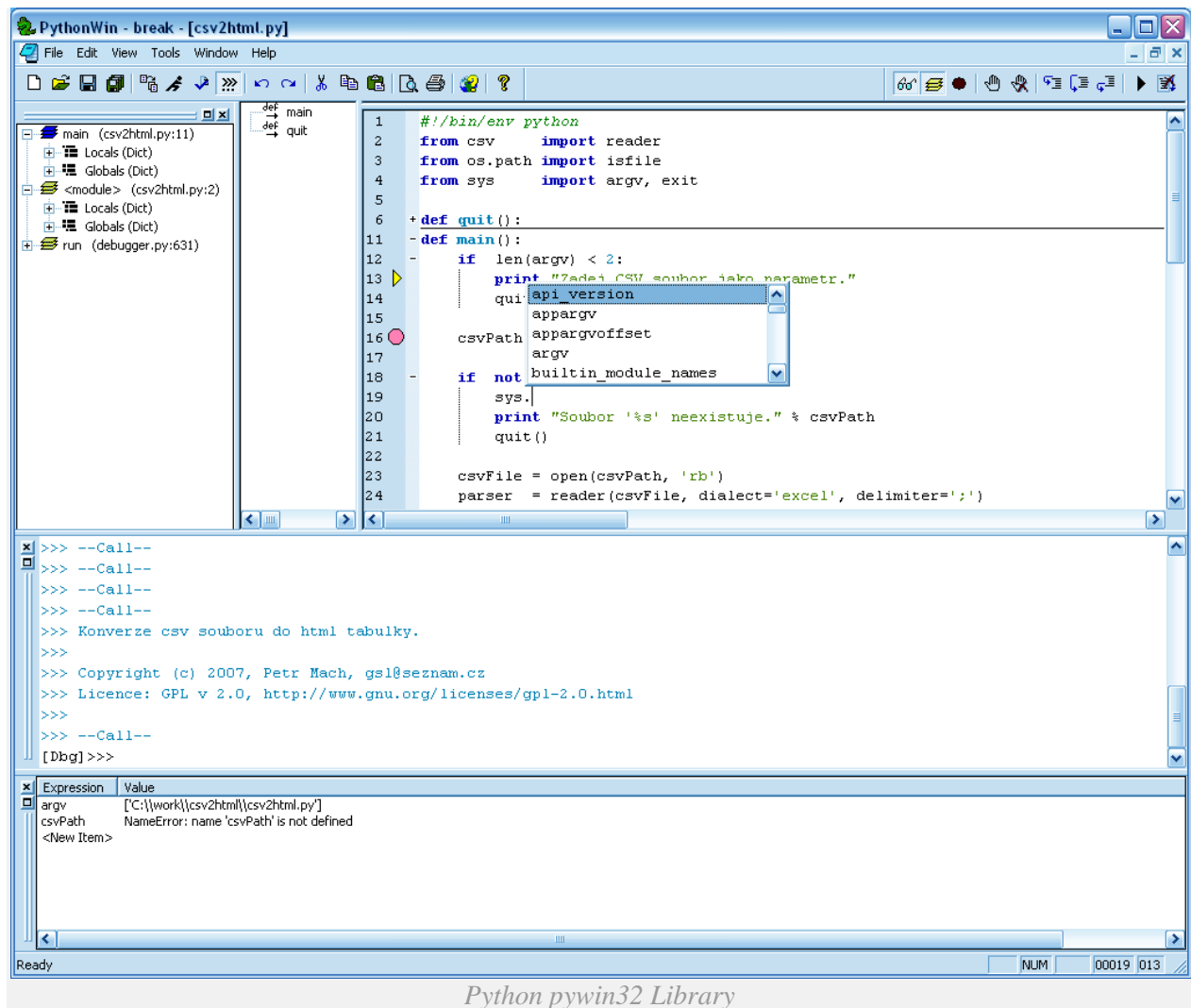


The friendly PIL fork

Python Libraries Tutorial- Pillow

m. pywin32

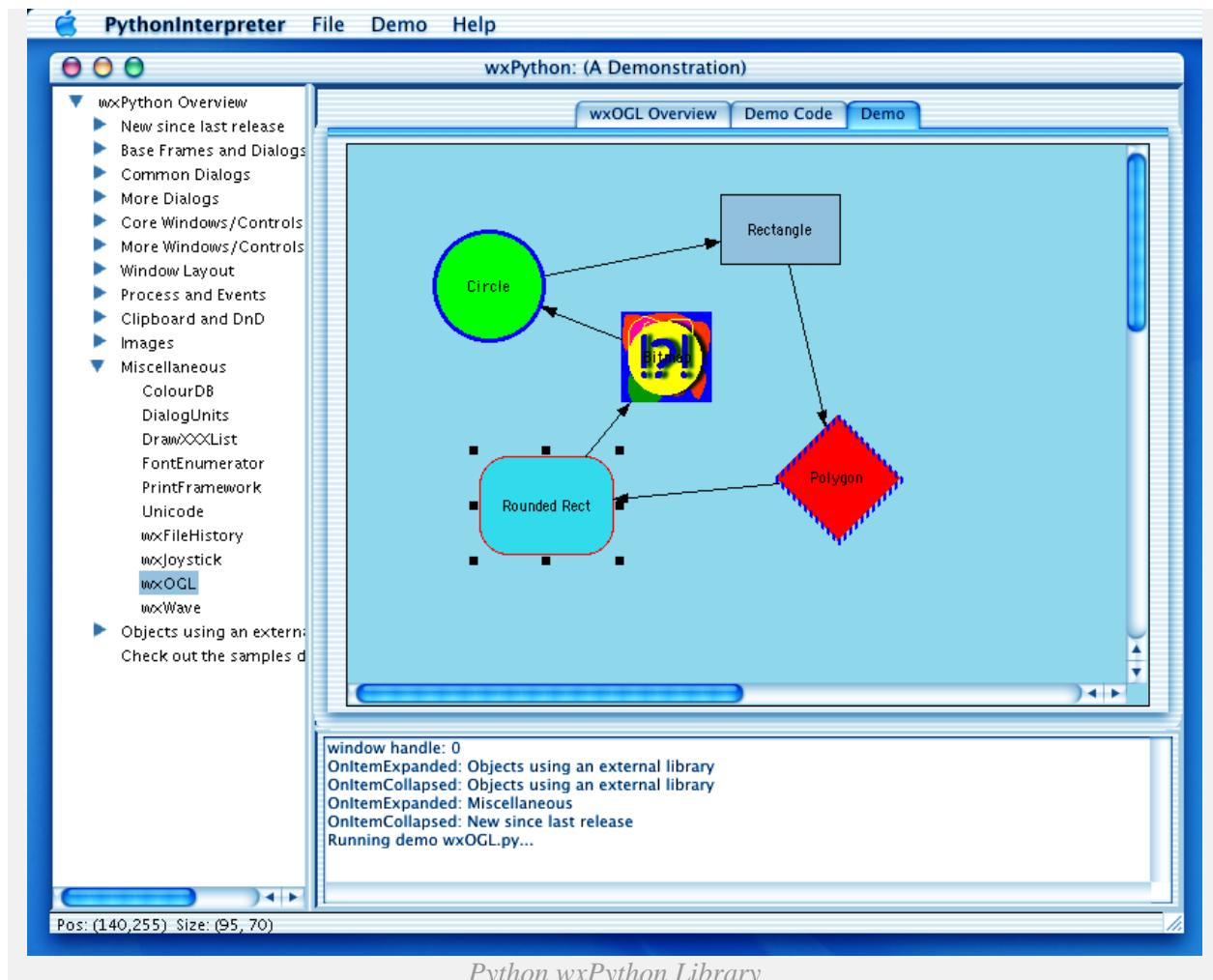
This provides useful methods and class for interaction with Windows, as the name suggests.



Python pywin32 Library

n. wxPython

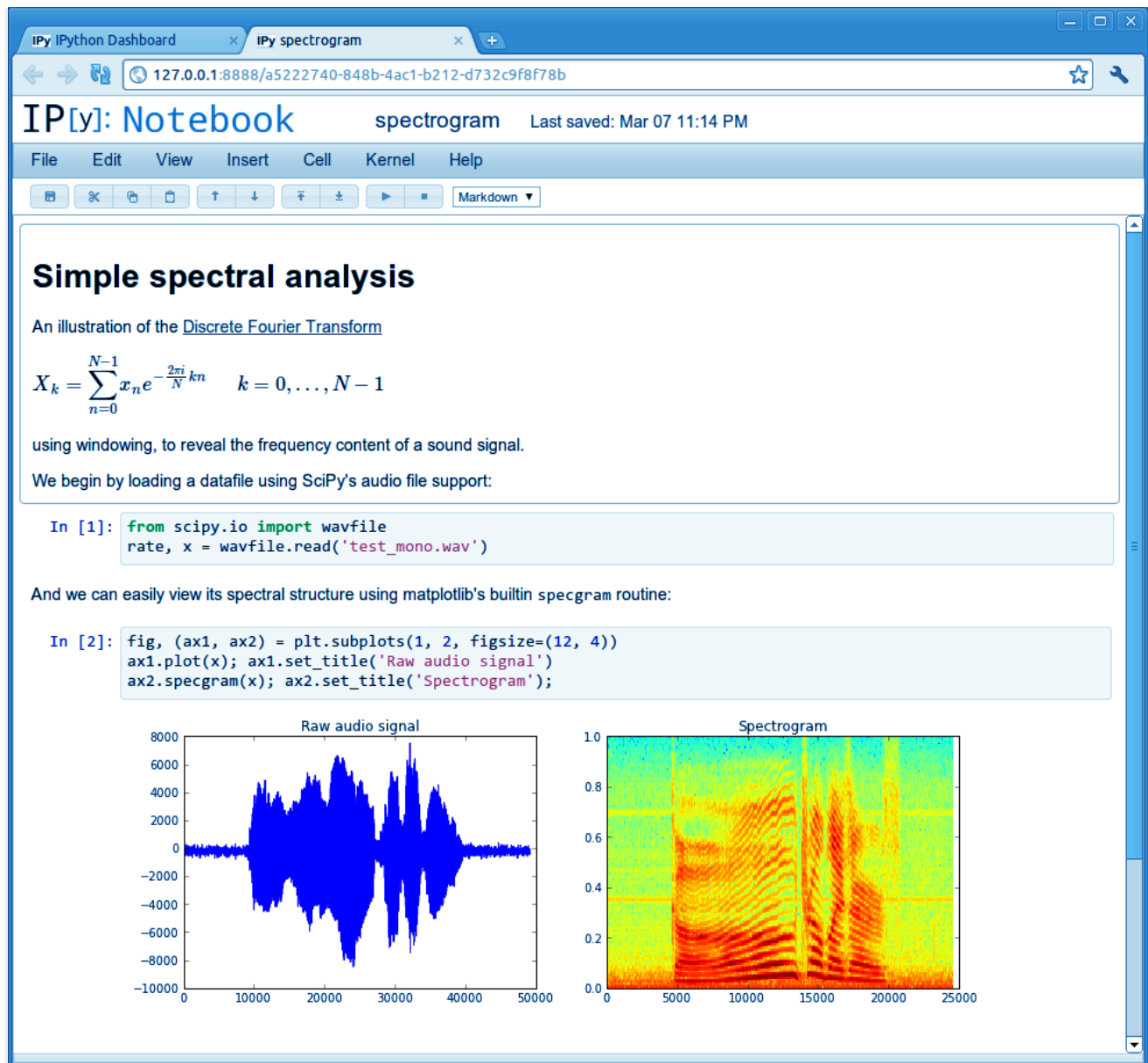
It is a wrapper around wxWidgets for Python.



Python wxPython Library

o. iPython

iPython Python Library has an architecture that facilitates parallel and distributed computing. With it, you can develop, execute, debug, and monitor parallel applications.



Python Library – iPython

Learn: [Python Regular Expressions](#)

p. Nose

Nose delivers an alternate test discovery and running process for unittest. This intends to mimic py.test's behavior as much as it can.

nose

is nicer testing for python

nose extends unittest to make testing easier.

Python Nose Library

q. Flask

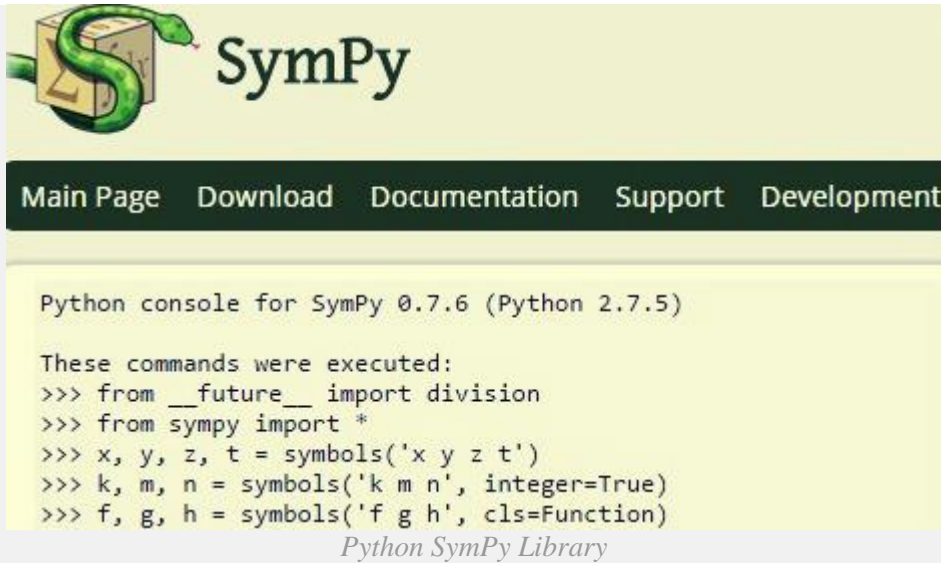
A web framework, Flask is built with a small core and many extensions.



Python Flask Library

r. SymPy

It is an open-source library for symbolic math. With very simple and comprehensible code that is easily extensible, SymPy is a full-fledged Computer Algebra System (CAS). It is written in Python, and hence does not need external libraries.



s. Fabric

Along with being a library, Fabric is a command-line tool for streamlining the use of SSH for application deployment or systems administration tasks. With it, you can execute local or remote shell commands, upload/download files, and even prompt running user for input, or abort execution.



t. PyGTK

PyGTK lets you easily create programs with a GUI (Graphical User Interface) with Python.

 [Users](#) · [Developers](#) · [Browse Sources](#) · [FTP](#) · [Bugzilla](#) · [FAQ](#) · [Contact](#)

PyGTK: GTK+ for Python

What is PyGTK?



PyGTK lets you to **easily create programs with a graphical user interface** using the Python programming language. The underlying GTK+ library provides all kind of visual elements and utilities for it and, if needed, you can develop full featured applications for the GNOME Desktop.

PyGTK applications are truly **multiplatform** and they're able to run, unmodified, on Linux, Windows, MacOS X and other platforms.

Other distinctive features of PyGTK are, besides its **ease of use** and rapid prototyping, its first class accessibility support or the capability to deal with complex multilingual or bidirectional text for fully localized applications.

PyGTK is **free software**, so you can use, modify, distribute and study it with very few restrictions (LGPL license).

***Note:** New users are encouraged to use GTK+3 through the [PyGObject bindings](#) instead of using PyGTK with GTK+2. Windows users may still want to keep using PyGTK until more convenient installers are published.*

Python PyGTK Library

Learn: [The Tremendous Python Career Opportunities in 2018](#)

So, this was all about Python Libraries Tutorial. Hope you like our explanation,

Conclusion

Now you know which libraries to go for if you choose to extend a career in Python. Many of these help us with data-science as well. Or if you wish to go out of your way, create your own library, and get it published with the PyPI; help the community grow. Furthermore, if you have any query, please share with us!