



 Français

Home

Features

License

Download

Tutorials

Documentation

Forum

Wiki

Tasks / bugs

Github page

RSS 2.0

 Donate

WARNING : this is the documentation for an old version of SFML ; the documentation for the latest official release is available through the main menu

Tutorial - Getting started - SFML and Python

| [Index](#)

Introduction

The purpose of this tutorial is to explain you how to compile / install / use the Python binding of SFML (which is called **PySFML**).

Installing PySFML binaries

In order to use PySFML, you need to download and install the archive containing the development files for your OS (SFML-x.x-python-dev-xxx). It can be found on the [download page](#). This archive contains the binaries of PySFML which can be used directly. You won't need anything else to use PySFML.

Extract the archive, and copy the files to your installation of Python. If you don't know where it's located, by default it should be "C:\Python25" on Windows or "/usr/lib/python2.5" on Linux.

Once the files are properly installed, you're ready to use PySFML.

Installing PySFML SDK

Installing the PySFML SDK is optional, all you need to run PySFML are the development files. However, installing the full SDK is recommended especially if you're doing your first steps with PySFML.

First you must download the PySFML SDK archive (SFML-x.x-python-sdk), which can be found on the [download page](#). This archive contains the source code, documentation, samples and a setup file to build and install PySFML.

You can then extract the SDK and enjoy the documentation and samples directly.

If you want to build PySFML from the source code, you must have the SFML C++ headers and libraries in the SFML-x.x directory (as well as the "python" folder), and type the following command :

```
python setup.py build
```

On Windows, this command will use Visual C++ 2003 by default. But you can compile with MinGW if you want, by adding the "-cmingw32" option :

```
python setup.py build -cmingw32
```

Then you can install it with the following command:

```
python setup.py install
```

On Linux you'll need the root access to install the files.

```
sudo python setup.py install
```

Writing your first PySFML program

Unlike the C++ libraries, PySFML gathers all the SFML classes into a single module ("sf"). As Python supports native threading, and has its own networking classes, only some classes of the System, Window, Graphics and Audio modules are exposed in PySFML.

Here is a very simple piece of SFML code that opens a window and displays some text :

```
# Include the PySFML extension
from PySFML import sf

# Create the main window
window = sf.RenderWindow(sf.VideoMode(800, 600), "PySFML test")

# Create a graphical string to display
text = sf.String("Hello SFML")

# Start the game loop
running = True
while running:
    event = sf.Event()
    while window.GetEvent(event):
        if event.Type == sf.Event.Closed:
            running = False

    # Clear screen, draw the text, and update the window
    window.Clear()
    window.Draw(text)
    window.Display()
```

The PySFML documentation is included in the SDK archive. If you're using the SVN version, you can update the documentation by running the "gen_doc.py" script. You can also get the detailed description of a class by entering `help(sf.the_class)` in the interpreter. Moreover, all the classes are the same as the C++ version, so you can still use the full documentation and tutorials on the website.

The "samples" directory of the SDK contains a few demos using PySFML, you can try them to make sure everything is working after compiling and/or installing.

:: Copyright © 2007-2011 [Laurent Gomila](#), all rights reserved :: Icons from the Vista-Inspirate free icons set by [Alexandre Moore](#) ::