# GeoExamples

Cython with PyCharm community

Apr 20, 2017

Some time ago, I explained how to use coverage and pylint with PyCharm community. Cython is also covered by PyCharm professional, but not the community edition, which makes working with cython a bit uncomfortable.
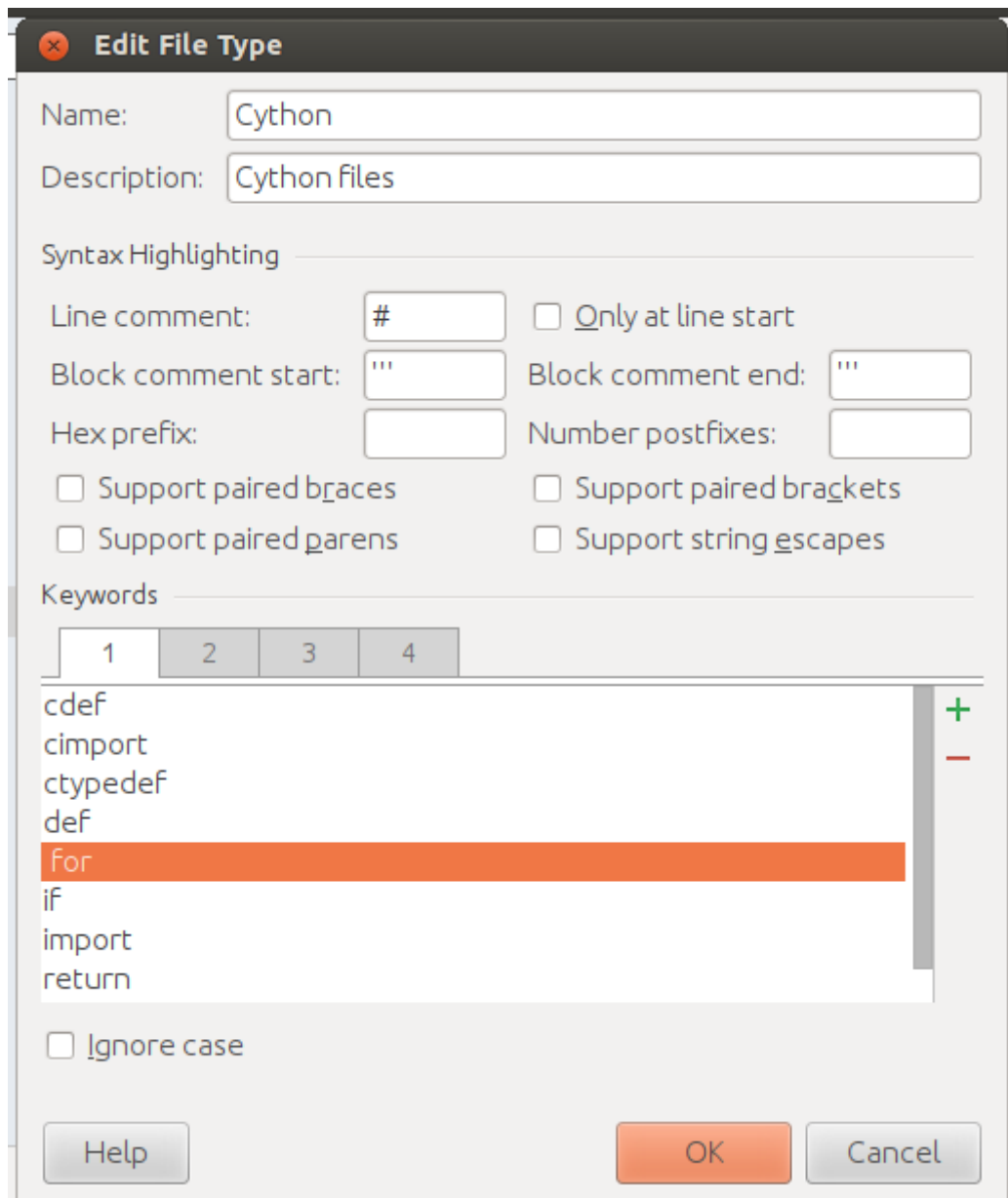
Here is how I managed to do it.

Syntax hightlighting

cython files end with *.pyx*. The syntaxis is similar to python, but the program doesn't recognize the file. Also, it has some special words that would give error in python, such as *cdef*. So a new file type must be created. Go to *File->Settings->Editor->File Types* and add a file type:
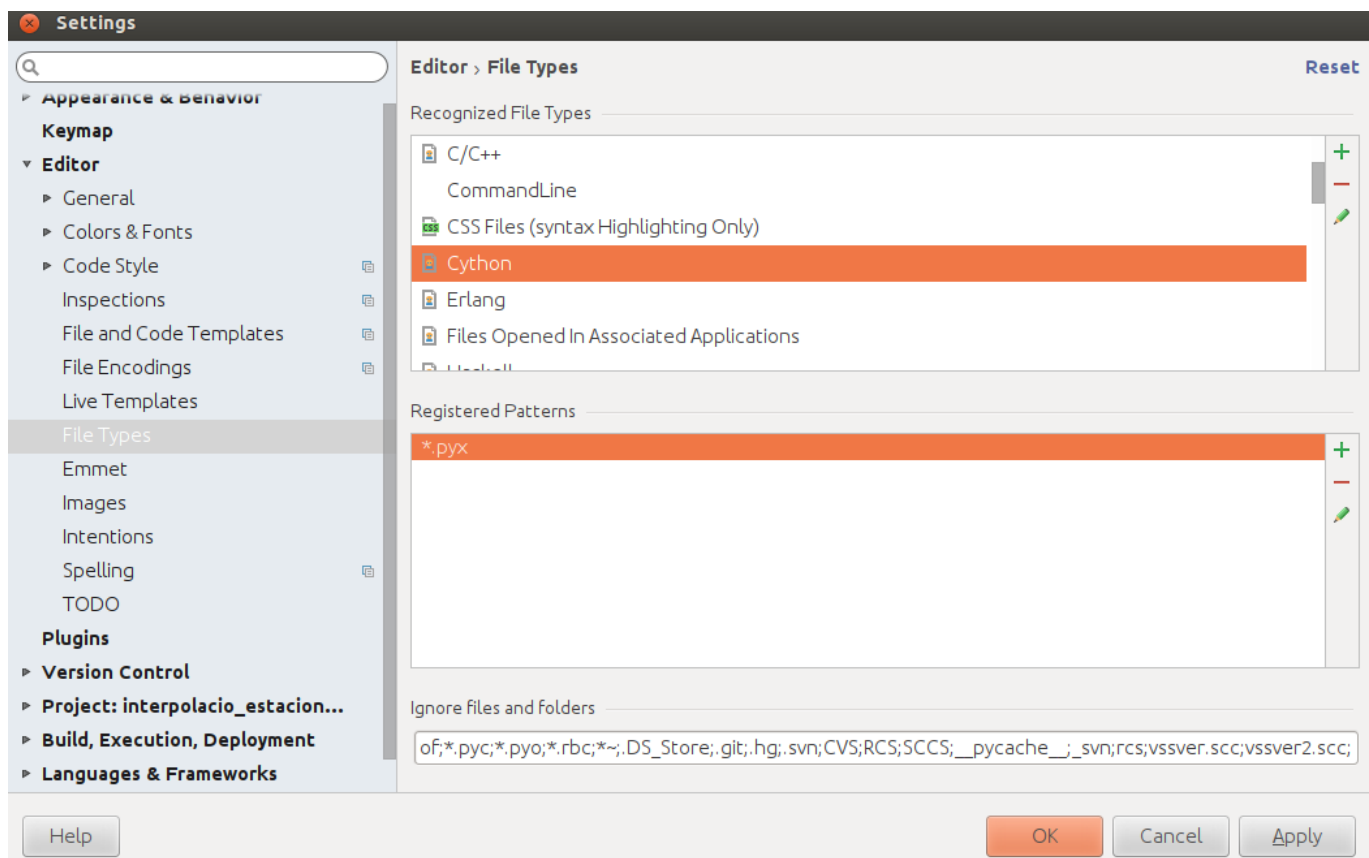
Then, edit it to make it good for the cython syntax:

Since there are many words to add, and no way to do it fast, I have exported my settings in this file, which you can import from *File->Import Settings*. I'll try to keep it updated by adding more keywords.

Compiling the cython file

As explained in the official cython basic tutorial, the best way to compile the files is creating a *setup.py* file with the compilation options:

```
from distutils.core import setup
from Cython.Build import cythonize


setup(
    ext_modules = cythonize("helloworld.pyx")
)
```
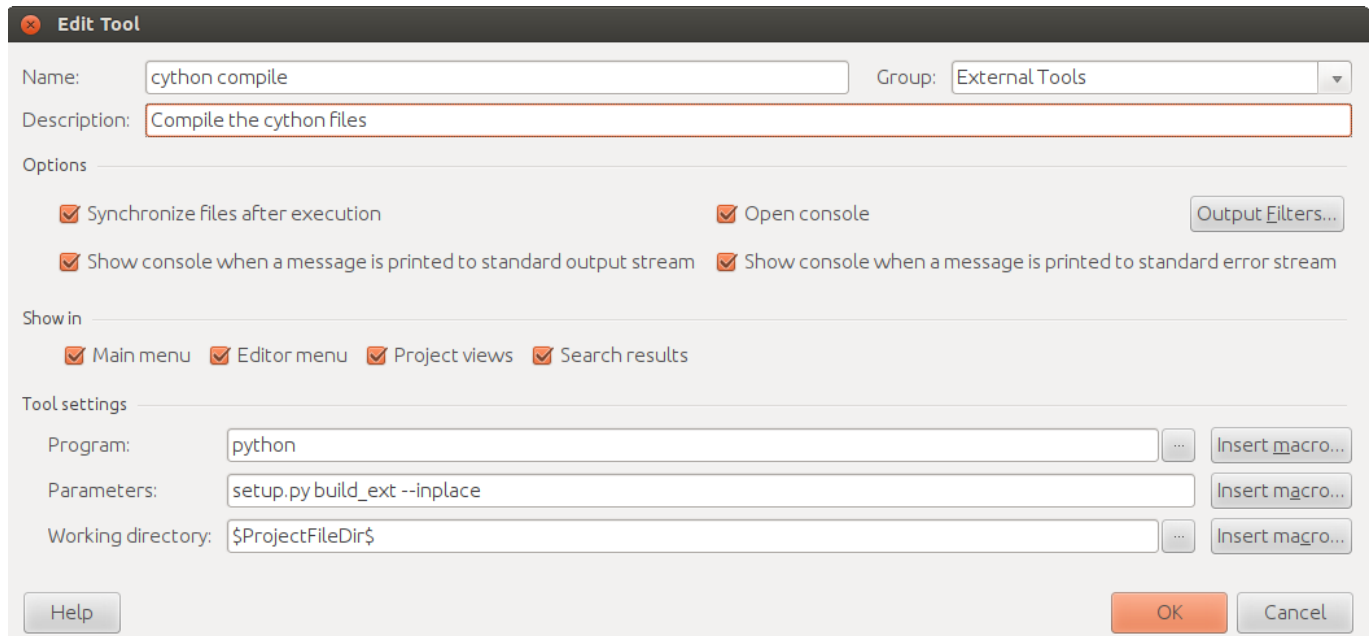
The compilation is then made by running:

```
python setup.py build_ext --inplace
```

Is possible to automatize this with an *external tool*:

Open *File->Settings->Tools->External tools* and click the + button. You will get a window that has to be filled like this:

You can run the compilation by going to *tools->External tools->cython compile* any moment or right clicking the file name at the project pane. Then, re-run it by clicking on the play button on the lower pane.

Usually, all the cython files will be compiled at once with the *setup.py*, so there is no need for creating special configurations depending on the file.

Links

- The settings file
- Coverage with Pycharm community
- PyCharm
- Cython
- Cython basic tutorial
- New file types in PyCharm