

# PyRuby

Today I saw a package called PyRuby in Github. The readme says:

PyRuby - Some Ruby for your Python!

PyRuby is a simple way to leverage the power of Ruby to make your Python code more readable and beautiful.

Usage

All you have to do is import the ruby module:

```
import ruby
```

From now on you should be able to write Ruby code within a regular Python module. An example:

```
1.upto(10) { |n| puts n }
```

Even PyPI has listed this as a package.

In the beginning I thought this was again a Ruby implementation by PyPy project. Or at least it use some magic trick to write ruby code directly in Python.

Then I browse into the source code of it.

It contains only one file: `ruby.py`

```

1  # -*- coding: utf-8 -*-
2
3  print("""
4
5      `.-:/+ossyhddmmmmmmNNNNNNNNmmmmmmddddd
dhhhyyyhhhyo:`
6      .:+sydNNNmmdhhyssso++/+++++++////:.....-` `` .....-
/oymms.
7      `:ohmdys+//:/:--:////:.-` `` .....` `` `` `.://:-`      `/d
Ns.
8      .+hNds:`.-:////:-----:////++///:-....-:////:-` .///.      `oMm/
9      /hNmO.` `` `` `` ....` `` `` `` `` `` `` `` ..-----:~:/+/-:/:~
/NMs
10     oMd/`      `:~:~.~:~//+`      //`      ` `` `` `` `~:~:~+/-` ~:~`      :NM
+
11     yN`      -+.`      `/`      o.      ` ``:~:~.``      :NN:
12     :Nm      -      ./      :      `.-://///:.-`      ` `` ``      :NN-
13     /NM/      .-:~:~.``      /      `:sdmdhyMMMMMMNNmy/`
      :mNo`
14     :hMd:      /dmddddNNmdy+-.      `smmy/-` `` `hMMMMMM
Mhydm/`-.` `` `.....:mMm+.
15     -hNd/-/o/-.:~.~ydmmmmmNNNNNNMMNh:/+-      dMN-`-+hmm
mmdhdddmmMN-`-/o:      .-:~:~:/oydms-
16     oNMo:+/::      ` `` .....-/ohNMNhs-      :hNmmdyo:.. `` `yo-` `` `.-
.-`-`-+shdddhst+` `` `./yms.
17     .MMo:/` o:~+sso+:~`      sM+      ./-`      /mNh+....-/ymNNd
o:~-/shd+`-` :mm:
18     /MM-o ./ohhsooohNmy::sh.      `yM/      `:oyyyyyyhys
+.:` hy      `/Nh` :~NN.

```

[illegible]

```
43         ``.....`
44
45     LOL U MAD?
46     """)
47
48     import sys
49     sys.exit(1)
```

Yes, indeed. The idea of using Ruby in Python is totally mad.