

Is it possible to decompile a compiled .pyc file into a .py file?

Is it possible to get some information out of the .pyc file that is generated from a .py file?

10 ANSWERS

`Uncompile2` worked well for me with Python 2.7 to decompile the .pyc bytecode into .py, whereas `unpyc` crashed with an exception.

For Python 3 and Python 2.7, you can try the more recent [Uncompyle6](#).

See [this answer on uncomplye2](#) for some other comments.

Yes, you can get it with `unpyclib` that can be found on *pypi*.

```
$ pip install unpyclib
```

Then you can decompile your .pyc file

You may try [Easy Python Decompiler](#). It's based on Decompyle++ and Uncompyle2. It's supports decompiling python versions 1.0-3.3

Note: I am the author of the above tool.

Decompile++ (pycdc) was the only one that worked for me:
<https://github.com/zrax/pycdc>

was suggested in [Decompile Python 2.7 .pyc](#)

Yes, it is possible.

There is a perfect open-source Python (.PYC) decompiler, called Decompyle++
<https://github.com/zrax/pycdc/>

Decompile++ aims to translate compiled Python byte-code back into valid and human-readable Python source code. While other projects have achieved this with varied success, Decompile++ is unique in that it seeks to support byte-code from any version of Python.

Online version is available here: <http://www.javadecompilers.com/pvc>

I have no idea if it's any good, but a quick google search turned up [decompile](#).