

Ask Ubuntu is a question and answer site for Ubuntu users and developers. Join them; it only takes a minute:

Sign up

Here's how it works:

Anybody can ask a question

Anybody can answer

The best answers are voted up and rise to the top

Why does Python in Linux require the line `#!/usr/bin/python`?

Pretty simple question: In Linux, why does Python require the line


```
#!/usr/bin/python
```

at the start of a python file, since Windows doesn't?

What does it do? 'cause the description "Links to Python" is a bit vague...


python   scripts

edited Nov 11 '15 at 2:54

 **Seth** ♦

29.5k   23   106   155

asked Nov 4 '15 at 21:48

 **DevRobot**

6,456   13   51   101

- 27 The answers below are all correct, none of them address why Windows does not need this line. Windows depends upon the extension of the file (the part after the `.`) to determine what kind of file it is. Even Windows is moving away from this: examine the first few lines of the Microsoft Word file, and it will state that it is in fact, a Microsoft Word file. – Charles Green Nov 4 '15 at 22:16
- 9 The elephant in the room is that you should NEVER use `/usr/bin/python` unless you are compatible with python 2 and 3. Reason: arch symlinks it to python3, a move which is acknowledged by the PSF, who recommend against it themselves. – Yet Another User Nov 5 '15 at 3:50
- 5 It's implied, but not stated explicitly in the answers below that **it's not required**. It's required if you want to execute the script just from its own name. You can always run `python myscript.py` instead. – Chris H Nov 5 '15 at 10:32
- 3 @CharlesGreen we are not suppose to know why windows does not ;) That is up to SO. – Rinzwind Nov 5 '15 at 13:13
- 2 @YetAnotherUser It's been six years and eleven months since Python 3 was released, at this point I feel people would be better off defaulting to 3 and explicitly specifying the use of 2 when necessary. – JAB Nov 5 '15 at 15:56

7 Answers

Python does not have any such special requirement on Linux. It's the *program loader* on Unix/Linux that uses the "shebang" line, as it's called. This is actually a feature rather than a limitation, but we'll get to that in a moment. The Wiki page on "shebang" has more details, but I'll try to give an overview as well as a comparison to Windows here.

First, let's look at the situation on Windows:

- When you attempt to open or run a file, Windows first examines the *extension* of that file. This is the *last* part of the filename starting with `.`. In the case of Python files, this is typically `.py`.
- Windows looks up what action to take based on the file extension.
  - This information is recorded in the Windows registry; when Python is installed, it typically tells Windows that `.py` files should be opened using the newly-installed *application* Python (i.e. the Python interpreter).
  - Several file-types have built-in behaviors; for instance, executable files (such as the Python interpreter itself) must end in `.exe`, and `.bat` files are executed as Windows batch-scripts.