

Re: polling for output from a subprocess module

Source: <http://coding.derkeiler.com/Archive/Python/comp.lang.python/2008-02/msg00708.html>

- *From:* Thomas Bellman <bellman@xxxxxxxxxxxxxxxx>
 - *Date:* Wed, 6 Feb 2008 07:18:35 +0000 (UTC)
-

Ivo <noreply@xxxxxxxx> wrote:

Thomas Bellman wrote:

However, the `os.read()` function will only read what is currently available. Note, though, that `os.read()` does not do line-based I/O, so depending on the timing you can get incomplete lines, or multiple lines in one read.

be carefull that you specify how much you want to read at a time, otherwise it cat be that you keep on reading.

Specify `read(1024)` or `somesuch`.

Well, of course you need to specify how much you want to read. Otherwise `os.read()` throws an exception:

```
>>> import sys, os
>>> os.read(sys.stdin.fileno())
Traceback (most recent call last):
File "<stdin>", line 1, in <module>
TypeError: read() takes exactly 2 arguments (1 given)
```

In case of my `PPCEncoder` I recompiled the `mencoder` subprocess to deliver me lines that end with `\n`.

If anyone can tell me how to read a continues stream than I am really interested.

Re: polling for output from a subprocess module

I have never had any problem when using the `os.read()` function, as long as I understand the effects of output buffering in the subprocess. The `file.read()` method is a quite different animal.

(And then there's the problem of getting `mplayer/mencoder` to output any *useful* information, but that is out of the scope of this newsgroup. :-)

—

Thomas Bellman, Lysator Computer Club, Linköping University, Sweden
"God is real, but Jesus is an integer." ! bellman @ lysator.liu.se
! Make Love -- Nicht Wahr!

.