

Get print quotes before printing for real with pykotme

Table of Contents

Chapter 1. Get print quotes before printing for real with pykotme.....1
 Examples.....1
 Reference.....2
 Notes.....2

Chapter 1. Get print quotes before printing for real with pykotme

Table of Contents

[Examples](#)

[Reference](#)

Last modified on \$Date: 2006-06-01 15:56:00 +0200 (jue 01 de jun de 2006) \$

PyKota features a print quote generator, named **pykotme**. This command line tool can be used to learn in advance how much a print job will cost to you if you really send it to a printer. You can then decide if printing your document is a good idea or not, and do it knowingly.

To get a print quote, you have to launch **pykotme** from the command line, passing your print job's content in any format recognized by PyKota [1], either in the form of one or more file names on the command line, or on its standard input (i.e. in a shell pipe or redirection).

Without any command line argument, **pykotme** automatically reads your job's data from its standard input.

By default, **pykotme** will display a quote for your job on all printers. You can restrict this by using the `-p` | `--printer` command line option followed with either a printer's name or a wildcard which will be used to search all printers which name matches. You can pass more than one printer name wildcard, separating them with commas.

pykotme's functionality is also available through the use of the **pykotme.cgi** CGI script. However without authentication **pykotme.cgi** will only return the print job's size and not its cost, because the cost depends on the value of the user's overcharging factor and the printer being used. That's why if you want to obtain the same functionality with the CGI script than with the command line tool, you must configure your web server to force users to authenticate whenever they access to **pykotme.cgi**. Then they will be able to choose one or more printers (all by default), and also to enter their username, and the exact cost of the print job will be computed.

Examples

Here are some examples of **pykotme** usage.

The simplest form :

```
port50-2:~$ pykotme pykota.ps
Your account balance : 16.25
Job size : 22 pages
Cost on printer apple : 1.35
port50-2:~$
```

More complex, with printer name wildcard and within a pipe :

```
port50-2:~$ a2ps -o - pykota/README | pykotme --printer "a*"
[pykota/README (plain): 6 pages on 3 sheets]
[Total: 6 pages on 3 sheets] sent to the standard output
Your account balance : 16.25
Job size : 3 pages
```

```
Cost on printer apple : 0.40  
port50-2:~$
```

Reference

pykotme [-v | --version] [-h | --help] [-P *printername* | --printer *printername*] [*filename1* | *filename2* | ... | *filenameN*]

Notes

- [1] PyKota now uses pkpgcounter which recognizes PostScript (both binary and DSC compliant), PCLXL (aka PCL6), PDF, PCL3/4/5, ESC/P2, TIFF, DVI, OpenOffice Writer, and OpenOffice Impress.