em Documentation

Release 0.2.1

Igor Kalnitsky

January 25, 2014

em is a terminal tool that prints FILE(s), or standard input to standard output and highlights the expressions that are matched the PATTERN.

```
igor at hp in :~/devel/em on git:master workon em
$ tail setup.py | em 'license' yellow -i | em 'Linux' green
        'License :: OSI Approved :: BSD License',
        'Programming Language :: Python',
        'Programming Language :: Python :: 2',
        'Programming Language :: Python :: 2.7',
        'Programming Language :: Python :: 3',
        'Programming Language :: Python :: 3.2',
        'Programming Language :: Python :: 3.3',
    ],
    platforms=['Linux', 'MacOS', 'Unix']
)
```

The expression will be highlighted iff the terminal is ANSI-compatible. This criterion is met by most popular terminal emulators on Linux, Unix and MacOS.

CHAPTER 1

Installation

You will need Python 2.7, 3.2 or above to get started. If you already have it, feel free to install em throuh pip util:

```
$ (sudo) pip install em
```

CHAPTER 2

Quickstart

em is cool and easy to use. The most practical usecase for me:

```
$ tail -f /path/to/log | em "ERROR" red | em "iphone:" green
```

The example above highlights all occurences of «ERROR» in red and all occurences of «iphone:» in green when watching the log.

But you can use em in the following way:

```
$ em "ERROR|CRITICAL" red /path/to/log
```

which prints the log and highlights all occurences of «ERROR» or «CRITICAL» in red.

CHAPTER 3

Options

Here is a command line interface of em:

```
usage: em [-h] [-i] [-l] [-s] [--version] PATTERN FORMAT [FILE [FILE ...]]

em program is a terminal tool that prints FILE(s), or standard input to
standard output and highlights the expressions that are matched the PATTERN.

positional arguments:
  PATTERN            a pattern to highlight
  FORMAT             a color to highlight matched expressions
  FILE               search for pattern in these file(s)

optional arguments:
  -h, --help         show this help message and exit
  -i, --ignore-case  ignore case distinctions
  -l, --line-mode    highlight entire line
  -s, --safe-mode    highlight only when stdout refers to tty
  --version          show program's version number and exit

With no FILE, or when FILE is -, read standard input. The FORMAT option must
be one of: BOLD, UNDERLINE, GREY, RED, GREEN, YELLOW, BLUE, MAGENTA, CYAN or
WHITE.
```

The CLI is clear, so I don't need to comment. :)