pyexcel-cli Documentation

Release 0.0.2

Onni Software Ltd.

Jun 19, 2017

Contents

Author C.W.

Source code http://github.com/pyexcel/pyexcel-cli.git

Issues http://github.com/pyexcel/pyexcel-cli/issues

License New BSD License

Development 0.0.2

Released 0.0.1

Generated Jun 19, 2017

Introduction

pyexcel-cli brings pyexcel to make it easy to consume/produce information stored in excel files on command line interface. This library can turn the excel data into a list of lists, a list of records(dictionaries), dictionaries of lists. And vice versa. Hence it lets you focus on data in shell programming, instead of file formats.

Hightlighted features:

1. View data in the excel files without Microsoft Office or Open Office

2. Transcode data among supported excel file formats

3. Merge files in various excel file formats into one

4. Split a multi-sheet excel file into single sheet files

5. Find difference in data between two excel files

Usage

Here is an example usage:

```
$ pyexcel view https://github.com/pyexcel/pyexcel-cli/blob/master/tests/fixtures/
→multiple-sheets.xls
Sheet 1:
+---+---+---+
| 1 | 2 | 3 |
+---+---+---+
| 4 | 5 | 6 |
+---+---+---+
| 7 | 8 | 9 |
+---+---+---+
Sheet 2:
+---+---+---+
| X | Y | Z |
+---+---+---+
| 1 | 2 | 3 |
+---+---+---+
| 4 | 5 | 6 |
+---+---+---+
Sheet 3:
+---+---+---+
| O | P | Q |
+---+---+---+
| 3 | 2 | 1 |
+---+---+---+
| 4 | 3 | 2 |
+---+---+---+
```

Because pyexcel family is loosely coupled, especially for file format supports, you install the libraries that you need to. If you need to support xls format, you will need to install pyexcel-xls. For more information, please see the plugin section. Installation
==
You can install it via pip:

```
$ pip install pyexcel-cli
```

or clone it and install it:

```
$ git clone https://github.com/pyexcel/pyexcel-cli.git
$ cd pyexcel-cli
$ python setup.py install
```

List of plugins

Table 3.1: A list of file formats supported by external plugins

Package name	Supported file formats	Dependencies	Python versions
pyexcel-io	csv, csvz[2], tsv, tsvz[3]		2.6, 2.7, 3.3, 3.4, 3.5, 3.6 pypy
pyexcel-xls	xls, xlsx(read only), xlsm(read only)	xlrd, xlwt	same as above
pyexcel-xlsx	xlsx	openpyxl	same as above
pyexcel-xlsxw	xlsx(write only)	XlsxWriter	same as above
pyexcel-ods3	ods	ezodf, lxml	2.6, 2.7, 3.3, 3.4 3.5, 3.6
pyexcel-ods	ods	odfpy	same as above
pyexcel-odsr	ods(read only)	lxml	same as above
pyexcel-text	(write only)json, rst, mediawiki, html, latex, grid, pipe, orgtbl, plain simple	tabulate	2.6, 2.7, 3.3, 3.4 3.5, pypy, pypy3
pyexcel-handsontable	handsontable in html	handsontable	same as above[1]
pyexcel-pygal	svg chart	pygal	same as above[1]

In order to manage the list of plugins installed, you need to use pip to add or remove a plugin. When you use virtualenv, you can have different plugins per virtual environment. In the situation where you have multiple plugins that does the same thing in your environment, you need to tell pyexcel which plugin to use per function call. For example, pyexcel-ods and pyexcel-odsr, and you want to get_array to use pyexcel-odsr. You need to append get_array(..., library='pyexcel-odsr').

---

[2] zipped csv file
[3] zipped tsv file
[1] coming soon

Command line reference

Usage: pyexcel-script.py [OPTIONS] COMMAND [ARGS]...

Read and write data in different excel formats

file type in pyexcel refers to file formats, e.g. csv, xls

Options:

        --help   Show this message and exit.

Commands:

1. diff diff two excel files
2. merge Merge excel files into one
3. split Split a multi-sheet file into single ones
4. transcode Transcode an excel file
5. view View an excel file

Tutorial

How to see the difference between two excel files using pyexcel-cli

First of all, the difference in fonts, styles, formula and charts are not supported. It only finds the difference in data. In order to see all command line options, you could do:

```
$ pyexcel diff --help
```

In order to get you started, here is an example command line:

```
$ pyexcel diff filea.xls fileb.csv
```

The source file content could come from stdin, but the command line syntax changes a bit:

```
$ cat filea.xls | pyexcel diff --source-file-type xls - fileb.csv
```