PySnip Manual

Andrea Censi

February 19, 2021

1 Introduction

PySnip is inspired by `python.sty` developed by James Brotchie. The workflow is different and it allows heavier jobs which are processed offline.

2 Installation

```
python setup.py develop
```

3 Basic use

Use the \py command in your LaTeX file:

```
\py{print('Hello world' + '!'*5)}
```

Then compile as follows:

```
pdflatex file.tex
pysnip-make -c make
```

This is the result:

Hello world!!!!!

4 Advanced usage

You can pass a job name to \py:

```
\py[myjob]{print('Hello world' + '!'*5)}
```

Look in the `snippets/` directory. You will see the files:

`myjob.py` The source that you specified.

`myjob.tex` The resulting latex source.

`myjob.rc` The exit value for the script (0=success).

`myjob.pyo` The cached copy of the source.

5 Even more advanced usage

Re-run all the scripts:

```
pysnip-make -c remake
```

Re-run all the scripts starting with the string "`myjob`":

```
pysnip-make -c "remake myjob*""
```

Start a Compmake shell:

```
pysnip-make
```

6 Source code

```
    \py[myjob2]{
from pysnip import write_source_minted, MyClass;
write_source_minted(MyClass)
    }
```

This is the result:

```
class MyClass:
    def f(self):
        a = 2
        a = 2
```

This is the result:

```
class MyClass:
    def f(self):
        a = 2
        a = 2
```