fptokens Documentation

Release 0.1.1

Florian Einfalt

Mar 06, 2017

Contents

Contents:

CHAPTER 1

Installation

To install `fptokens`, type:

```
$ pip install fptokens
```

CHAPTER 2

Getting Started

To get started with `fptokens`, type:

```
import fptokens as fpt
```

To create a file name, type:

```
filename = fpt.Filename(root='/Users/demo/Desktop',
                        folders=['assets', '$colors$'],
                        base=['asset', '$colors$', '1200px'])
```

This created a file name with default settings, _ as the separator, `jpg` as the extension and `$` as the escape character for the tokens.

To parse and convert the tokens of the file name to actual tokens, type:

```
filename.parse()
```

To get the results of the parsing, type:

```
print filename.tokens
[<Token: $color$>]
```

The list of tokens could now be used to create permutations of the tokenised file name for example for batch output of image assets.

To create a generator of all permutations, define a set of data for each of the tokens with the token name as the argument name:

```
for permutation in filename.resolve(colors=['white', 'black', 'red', 'blue']):
    # do something with the permutation
    print permutation.abspath
# prints
# /Users/demo/Desktop/assets/white/asset_white_1200px.jpg
# /Users/demo/Desktop/assets/black/asset_black_1200px.jpg
```

```
# /Users/demo/Desktop/assets/red/asset_red_1200px.jpg
# /Users/demo/Desktop/assets/blue/asset_blue_1200px.jpg
```

Once tokens have been replaced with real-world data to create permutations, the relevant folders can be created by typing:

```
permutation.make()
```

CHAPTER 3

API Documentation

class fptokens.**Token**(*name*, *escape='$'*)
Token object for use in *Filename*.

Parameters

- **name** (*str*) – Token name
- **escape** (*str*) – Escape character, default: $

name
Return the token name.

Returns Token name

Return type str

token
Return the full token including the escape characters either side.

Returns Token

Return type str

class fptokens.**Filename**(*root*, *folders=[]*, *base=[]*, *separator='_'*, *extension='jpg'*, *escape='$'*)
Filenames with support for tokens.

Parameters

- **root** (*str*) – Root location
- **folders** (*list of str*) – Folder names, attribute supports tokens
- **base** (*list of str*) – Basename of the file, attribute supports tokens
- **separator** (*str*) – Separator for basename elements
- **extension** (*str*) – Filename extension
- **escape** (*str*) – Escape character for tokens, default: $

abspath
Return the filename's full absolute path.

Returns Absolute path

Return type `Path`

basename
Return the filename's basename.

Returns Basename

Return type str

dirname
Return the filename's location.

Returns Dirname

Return type str

make()
Create the filename's location if it does not exist.

parse()
Parse the filename's `folders` and `base` attributes, detect components that match the token pattern and replace these with *`fptokens.Token`* objects.

resolve(***kwargs*)
Given a set of **kwargs, yield all possible permutations for the data set provided. Raise `TokenError` if *`Filename`* does not haven tokens or the data provided does not match the tokens.

Params **kwargs Permutation data

root
Return the filename's root location.

Returns Root location

Return type `Path`

tokens
Return a list of all tokens in `folders` and `base`.

Returns List of tokens

Return type list

CHAPTER 4

Indices and tables

- genindex
- modindex
- search

Index