python-colourlovers Documentation

Release 0.1.1

Sebastian Vetter

December 06, 2016

Contents

1 ColourLovers 3

2 Colour 5

3 Palette 7

4 Pattern 9

5 Lover 11

6 Stat 13

7 The RGB and HSV colour classes 15

8 Indices and tables 17

Python Module Index 19

This module provides access to the ColourLovers API, a web service that allows users to publish colour themes and rate them. The API of this webservice allows to search for users (Lovers) and their posted contents (Colours, Patterns, Palettes).

Accessing the API requires an instance of `ColourLovers` which provides the methods to access the different content types: *Colour*, *Palette*, *Pattern*, *Lover*. Sending a request to ColourLovers is as easy as calling the corresponding method such as `ColourLovers.palettes()` to search for palettes. Additionally, the following arguments can be specified `new`, `top`, `random` with additional parameters. Please refer to the ColourLovers API documentation to find out more about the specific parameters and their restrictions at http://www.colourlovers.com/api.

You find the ColourLovers website at: http://www.colourlovers.com

Usage example:

```
>>> from colourlovers import ColourLovers
>>> cl = ColourLovers()
>>> cl.color('#37cbff')
[<Colour id='4767129' title='i feel pretty' rgb=(55, 203, 255)>]
>>> cl.palettes('new', keywords='funky', numResults=3)
[<Palette id='1940972' title='"Funky President"'>,
 <Palette id='1936394' title='Barbie Doll Blonde'>,
 <Palette id='1936247' title='Lily's Rainbow'>]
```

Another example:

```
>>> cl.patterns('random')
[<Pattern id='391644' title='acanalado'>]
>>> cl.colors('top', numResults=3)
[<Colour id='14' title='Black' rgb=(0, 0, 0)>,
 <Colour id='16' title='white' rgb=(255, 255, 255)>,
 <Colour id='1086335' title='dutch teal' rgb=(22, 147, 165)>]
```

CHAPTER 1

ColourLovers

class `colourlovers.`**`ColourLovers`**

`color`(*argument*, ***kwargs*)

`colors`(*argument=None*, ***kwargs*)

`lover`	Restrict results to specific COLOURlover by providing their user name.
`hue_range`	Hue range of the colour: `12,68`. Values must be in range [0,359] and the left value must be less than the right value.
`bri_range`	Brightness range of the colours: `2,88`. Values must be in range [0, 99] and the left value must be less than the right value.
`keywords`	String of keywords to search for in patterns.
`keyword_exact`	Perform an exact search for the keywords passed if set to 1. Default: 0.
`order_col`	`dateCreated`, `score`, `name`, ``numVotes` or `numViews`
`sort_by`	Sorting in asceding (`ASC`) or descending (`DESC`) order. Defaults to (`ASC`)
`num_results`	Number of results to be returned. Maximum: 100. Default: 20.
`result_offset`	Result offset, for paging. Default: 0.

`palette`(*argument*, ***kwargs*)

`palettes`(*argument=None*, ***kwargs*)

`lover`	Restrict results to specific COLOURlover by providing their user name.
`hue_option`	`yellow`, `orange`, `red` or `green`, `violet` or `blue`
`hex`	Restrict results to patterns with a specific colour given as any valid 6-char hex value without preceeding `#`.
`keywords`	String of keywords to search for in patterns.
`keyword_exact`	Perform an exact search for the keywords passed if set to 1. Default: 0.
`order_col`	`dateCreated`, `score`, `name`, ``numVotes` or `numViews`
`sort_by`	Sorting in asceding (`ASC`) or descending (`DESC`) order. Defaults to (`ASC`)
`num_results`	Number of results to be returned. Maximum: 100. Default: 20.
`result_offset`	Result offset, for paging. Default: 0.
`show_palette_widths`	Shows palette's colour's widths if set to `1`. Default: `0`

`pattern`(*argument*, ***kwargs*)

`patterns`(*argument=None*, ***kwargs*)

`lover`	Restrict results to specific COLOURlover by providing their user name.
`hue_option`	`yellow`, `orange`, `red` or `green`, `violet` or `blue`
`hex`	Restrict results to patterns with a specific colour given as any valid 6-char hex value without preceeding `#`.
`keywords`	String of keywords to search for in patterns.
`keyword_exact`	Perform an exact search for the keywords passed if set to 1. Default: 0.
`order_col`	`dateCreated`, `score`, `name`, ``numVotes` or `numViews`
`sort_by`	Sorting in asceding (`ASC`) or descending (`DESC`) order. Defaults to (`ASC`)
`num_results`	Number of results to be returned. Maximum: 100. Default: 20.
`result_offset`	Result offset, for paging. Default: 0.

lover(*argument*, ***kwargs*)

`comments`	0 or 1: if 1, will show last 10 comments for given Lover

lovers(*argument=None*, ***kwargs*)

`order_col`	Order in which the colours will be shown. Possible values are: `dateCreated`, `score`, `name`, ``numVotes` or `numViews`.
`sort_by`	Sorting in asceding (`ASC`) or descending (`DESC`) order. Defaults to (`ASC`)
`num_results`	Number of results to be returned. Maximum: 100. Default: 20.
`result_offset`	Result offset, for paging. Default: 0.

stats(*stats_type*)

Return the stats for *stat_type*. *stat_type* refers to one of the content types on ColourLovers and can be `colors`, `lovers`, `patterns`, `palettes`. A `ColourLoversError` is raised when an invalid type is requested.

Args: *stat_type (str)*: content type to request stats for.

Returns: The requested stats as `Stat` object.

Request the statistical value (total number) for *stats_type*. *stats_type* is a string that can be either of the following: `colors`, `patterns`, `palettes`, `lovers`.

Args: stats_type (str): Content type to retrieve statistics for.

Returns: Total number of the requested content type on ColourLovers.com.

CHAPTER 2

Colour

class `colourlovers.`**`Colour`**`(`***kwargs*`)`

This class defines a ColourLovers colour in the RGB and HSV colour spaces. The colour values can be accessed through the `rgb` and `hsv` respectively and are of type *RGB* and *HSV*.

`id`
Unique id for this Color as `int`.

`title`
Title / Name of the Color.

`user_name`
Username of the Color's creator.

`num_views`
Number of views this Color has received as `int`.

`num_votes`
Number of votes this Color has received as `int`.

`num_comments`
Number of comments this Color has received as `int`.

`num_hearts`
Number of hearts this Color has in the range of [0,5] as `float`.

`rank`
This Color's rank on COLOURlovers.com as `int`.

`date_created`
Date this Color was created as a `datetime` object.

`description`
This Color's description.

`url`
This Color's COLOURlovers.com URL.

`image_url`
Link to a PNG version of this Color.

`badge_url`
Link to a COLOURlovers.com badge for this Color.

`api_url`
This Color's COLOURlovers.com API URL.

classmethod **`from_xml`**(*xml*)

Create a new colour instance from *xml*. *xml* is a DOM element with the root element name `color`.

Args: xml (Element): `color` DOM element.

Returns: New instance of `Colour`.

classmethod **`tag`**()

Returns the name of the XML tag for a colour object.

CHAPTER 3

Palette

class `colourlovers.`**`Palette`**`(`***kwargs*`)`

`id`
Unique id for this Palette as `int`.

`title`
Title / Name of the Palette.

`user_name`
Username of the Palette's creator.

`num_views`
Number of views this Palette has received as `int`.

`num_votes`
Number of votes this Palette has received as `int`.

`num_comments`
Number of comments this Palette has received as `int`.

`num_hearts`
Number of hearts this Palette has in the range [0,5] as `float`.

`rank`
This Palette's rank on COLOURlovers.com as `int`.

`date_created`
Date this Palette was created as `datetime` object.

`colors`
List of Colors within this Palette as hex values

`color_widths`
This Palette's Color's widths in the range [0.0, 1.0] as `float`.

Note: this attribute is optional and might not be present if not returned by the API response.

`description`
This Palette's description

`url`
This Palette's COLOURlovers.com URL.

`image_url`
Link to a png version of this Palette.

badge_url
Link to a COLOURlovers.com badge for this Palette.

api_url
This Palette's COLOURlovers.com API URL.

CHAPTER 4

Pattern

class `colourlovers.`**`Pattern`**(***kwargs*)

`id`
Unique id for this Pattern as `int`.

`title`
Title / Name of the Pattern.

`user_name`
Username of the Pattern's creator.

`num_views`
Number of views this Pattern has received as `int`.

`num_votes`
Number of votes this Pattern has received as `int`.

`num_comments`
Number of comments this Pattern has received as `int`.

`num_hearts`
Number of Hearts this Pattern has in the range [0,5] as `float`.

`rank`
This Pattern's rank on COLOURlovers.com as `int`.

`date_created`
Date this Pattern was created as `datetime` object.

`colors`
List of colors within this Pattern as hex code.

`description`
This Pattern's description.

`url`
This Pattern's COLOURlovers.com URL

`image_url`
Link to a PNG version of this Pattern.

`badge_url`
Link to a COLOURlovers.com badge for this Pattern.

`api_url`
This Pattern's COLOURlovers.com API URL.

CHAPTER 5

Lover

class `colourlovers.`**`Lover`**`(`***kwargs*`)`

`id`
Unique id for this Lover as `int`.

`user_name`
This Lover's Username.

`date_registered`
Date this Lover registered with COLOURlovers.com as `datetime` object.

`date_last_active`
Date this Lover was last active on COLOURlovers.com as `datetime` object.

`rating`
This Lover's rating

`location`
This Lover's location.

`num_colors`
Number of Colors this Lover has made as `int`.

`num_palettes`
Number of Palettes this Lover has made as `int`.

`num_patterns`
Number of Patterns this Lover has made as `int`.

`num_comments_made`
Number of comments this Lover has made as `int`.

`num_lovers`
Number of Lovers [friends] this Lover has as `int`.

`num_comments_on_profile`
Number of comments this Lover has on their profile as `int`.

`comments`
Last 10 comments made on this Lover's profile as list of `Comment` instances.

Note: this attribute is optional. It is only returned when sending the parameter `comments=1` in the request.

`url`
This Lover's COLOURlovers.com URL.

`api_url`
This Lover's COLOURlovers.com API URL.

CHAPTER 6

Stat

class `colourlovers.`**`Stat`**(*total*, ***kwargs*)

`total`

Total number of colors, palettes, patterns or lovers in the COLOURlovers system.

CHAPTER 7

The RGB and HSV colour classes

class `colourlovers.`**`RGB`**(*red*, *green*, *blue*)

Define a RGB colour as a triple of integers in the range from 0 to 255. The colour channels are stored in attributes `red`, `green`, `blue`.

classmethod **`from_xml`**(*xml*)

Create an instance of `RGB` from *xml*.

Args: *xml (Element)*: `rgb` element from API response.

Returns: Instance of `RGB`.

`red`

Red colour channel in range [0, 255]

`green`

Green colour channel in range [0, 255]

`blue`

Blue colour channel in range [0, 255]

`hex`

Return hex colour code corresponding to the RGB value.

class `colourlovers.`**`HSV`**(*hue*, *saturation*, *value*)

Define a HSV colour instance from *hue*, *saturation* and *value*. The three values have to be integer values with *hue* in range [0, 360] and *saturation*, *value* in range [0, 255].

classmethod **`from_xml`**(*xml*)

Create an instance of `HSV` from *xml*.

Args: *xml (Element)*: `hsv` element from API response.

Returns: Instance of `HSV`.

`hue`

HSV hue channel in range [0, 360]

`saturation`

HSV saturation channel in range [0, 255]

`value`

HSV value channel in range [0, 255]

CHAPTER 8

Indices and tables

- genindex
- modindex
- search

Python Module Index

C

colourlovers, 3

A

api_url (colourlovers.Colour attribute), 5
api_url (colourlovers.Lover attribute), 11
api_url (colourlovers.Palette attribute), 8
api_url (colourlovers.Pattern attribute), 9

B

badge_url (colourlovers.Colour attribute), 5
badge_url (colourlovers.Palette attribute), 7
badge_url (colourlovers.Pattern attribute), 9
blue (colourlovers.RGB attribute), 15

C

color() (colourlovers.ColourLovers method), 3
color_widths (colourlovers.Palette attribute), 7
colors (colourlovers.Palette attribute), 7
colors (colourlovers.Pattern attribute), 9
colors() (colourlovers.ColourLovers method), 3
Colour (class in colourlovers), 5
ColourLovers (class in colourlovers), 3
colourlovers (module), 1
comments (colourlovers.Lover attribute), 11

D

date_created (colourlovers.Colour attribute), 5
date_created (colourlovers.Palette attribute), 7
date_created (colourlovers.Pattern attribute), 9
date_last_active (colourlovers.Lover attribute), 11
date_registered (colourlovers.Lover attribute), 11
description (colourlovers.Colour attribute), 5
description (colourlovers.Palette attribute), 7
description (colourlovers.Pattern attribute), 9

F

from_xml() (colourlovers.Colour class method), 5
from_xml() (colourlovers.HSV class method), 15
from_xml() (colourlovers.RGB class method), 15

G

green (colourlovers.RGB attribute), 15

H

hex (colourlovers.RGB attribute), 15
HSV (class in colourlovers), 15
hue (colourlovers.HSV attribute), 15

I

id (colourlovers.Colour attribute), 5
id (colourlovers.Lover attribute), 11
id (colourlovers.Palette attribute), 7
id (colourlovers.Pattern attribute), 9
image_url (colourlovers.Colour attribute), 5
image_url (colourlovers.Palette attribute), 7
image_url (colourlovers.Pattern attribute), 9

L

location (colourlovers.Lover attribute), 11
Lover (class in colourlovers), 11
lover() (colourlovers.ColourLovers method), 4
lovers() (colourlovers.ColourLovers method), 4

N

num_colors (colourlovers.Lover attribute), 11
num_comments (colourlovers.Colour attribute), 5
num_comments (colourlovers.Palette attribute), 7
num_comments (colourlovers.Pattern attribute), 9
num_comments_made (colourlovers.Lover attribute), 11
num_comments_on_profile (colourlovers.Lover attribute), 11
num_hearts (colourlovers.Colour attribute), 5
num_hearts (colourlovers.Palette attribute), 7
num_hearts (colourlovers.Pattern attribute), 9
num_lovers (colourlovers.Lover attribute), 11
num_palettes (colourlovers.Lover attribute), 11
num_patterns (colourlovers.Lover attribute), 11
num_views (colourlovers.Colour attribute), 5
num_views (colourlovers.Palette attribute), 7
num_views (colourlovers.Pattern attribute), 9
num_votes (colourlovers.Colour attribute), 5
num_votes (colourlovers.Palette attribute), 7
num_votes (colourlovers.Pattern attribute), 9

P

Palette (class in colourlovers), 7
palette() (colourlovers.ColourLovers method), 3
palettes() (colourlovers.ColourLovers method), 3
Pattern (class in colourlovers), 9
pattern() (colourlovers.ColourLovers method), 3
patterns() (colourlovers.ColourLovers method), 3

R

rank (colourlovers.Colour attribute), 5
rank (colourlovers.Palette attribute), 7
rank (colourlovers.Pattern attribute), 9
rating (colourlovers.Lover attribute), 11
red (colourlovers.RGB attribute), 15
RGB (class in colourlovers), 15

S

saturation (colourlovers.HSV attribute), 15
Stat (class in colourlovers), 13
stats() (colourlovers.ColourLovers method), 4

T

tag() (colourlovers.Colour class method), 6
title (colourlovers.Colour attribute), 5
title (colourlovers.Palette attribute), 7
title (colourlovers.Pattern attribute), 9
total (colourlovers.Stat attribute), 13

U

url (colourlovers.Colour attribute), 5
url (colourlovers.Lover attribute), 11
url (colourlovers.Palette attribute), 7
url (colourlovers.Pattern attribute), 9
user_name (colourlovers.Colour attribute), 5
user_name (colourlovers.Lover attribute), 11
user_name (colourlovers.Palette attribute), 7
user_name (colourlovers.Pattern attribute), 9

V

value (colourlovers.HSV attribute), 15