hyperdiary Documentation

Release 0.6.1

luphord

Oct 21, 2020

Contents:

1 hyperdiary **1**
- 1.1 Installation . . . 1
- 1.2 Setup . . . 1
- 1.3 Usage . . . 2

2 Installation **3**
- 2.1 Stable release . . . 3
- 2.2 From sources . . . 3

3 hyperdiary **5**
- 3.1 hyperdiary package . . . 5

4 Contributing **13**
- 4.1 Types of Contributions . . . 13
- 4.2 Get Started! . . . 14
- 4.3 Pull Request Guidelines . . . 15
- 4.4 Tips . . . 15
- 4.5 Deploying . . . 15

5 Credits **17**
- 5.1 Development Lead . . . 17
- 5.2 Contributors . . . 17

6 History **19**
- 6.1 0.6.1 (2020-10-21) . . . 19
- 6.2 0.6.0 (2019-11-29) . . . 19
- 6.3 0.5.0 (2019-11-06) . . . 19
- 6.4 0.4.0 (2019-10-30) . . . 20
- 6.5 0.3.2 (2019-10-17) . . . 20
- 6.6 0.3.1 (2019-07-03) . . . 20
- 6.7 0.3.0 (2019-06-19) . . . 20
- 6.8 0.2.0 (2019-06-12) . . . 20
- 6.9 0.1.2 (2019-06-11) . . . 20
- 6.10 0.1.1 (2019-04-29) . . . 21
- 6.11 0.1.0 (2019-04-28) . . . 21

7 Indices and tables **23**

Python Module Index **25**

Index **27**

CHAPTER 1

hyperdiary

A command line tool for writing diaries with hyperlinks. Free software, licensed under MIT license.

1.1 Installation

hyperdiary requires Python version 3.5 or later. Once you have Python and pip installed on your machine (and available in your path) you can install hyperdiary by executing

```
pip install hyperdiary
```

1.2 Setup

A project file *hyperdiary.json* is required to setup your diary. It should be of this form:

```
{
        "sources": [
                "2019/05.yaml",
                "2019/06.yaml"
        ],
        "expected": [
                {"start": "2019-05-01", "end": "2019-05-01"},
                {"start": "2019-06-09", "end": "2019-06-10"}
        ],
        "localization": {
                "months": ["Jan", "Feb", "Mar", "Apr", "May", "Jun",
                           "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"],
                "date_fmt": "%-d.%-m.%Y"
        }
}
```

You diary content itself is entered in yaml files like this (*2019/06.yaml*):

```
2019-06-09:
  - Some entry goes here with a $special_identity|link
  - This entry is +surprise tagged +mytag +nexttag
  - This entry contains nothing new
2019-06-10:
  - Same $special_identity|link again and $New_Identity|NewIdentity
  - $test +mytag
```

Use *$target|linktext* to create a hyperlink to *target* showing text *linktext*. Use *+mytag* to add tag *mytag*.

1.3 Usage

```
usage: hyperdiary [-h]
          {check,stats,html,htmlfolder,hugo,tiddlers,tiddlywiki,view}
          ...

The hyperdiary main command line interface.

optional arguments:
        -h, --help            show this help message and exit

subcommands:
        {check,stats,html,htmlfolder,hugo,tiddlers,tiddlywiki,view}
                                Available subcommands
        check               Check entire diary for integrity up-to-dateness
        stats               Calculate impressive diary statistics
        html                Export diary to html
        htmlfolder          Export diary to html in folders
        hugo                Export diary to hugo static site format
        tiddlers            Export diary to tiddlywiki tiddlers format
        tiddlywiki          Export diary to tiddlywiki
        view                View entries on command line
```

CHAPTER 2

Installation

2.1 Stable release

To install hyperdiary, run this command in your terminal:

```
$ pip install hyperdiary
```

This is the preferred method to install hyperdiary, as it will always install the most recent stable release.

If you don't have pip installed, this Python installation guide can guide you through the process.

2.2 From sources

The sources for hyperdiary can be downloaded from the Github repo.

You can either clone the public repository:

```
$ git clone git://github.com/luphord/hyperdiary
```

Or download the tarball:

```
$ curl  -OL https://github.com/luphord/hyperdiary/tarball/master
```

Once you have a copy of the source, you can install it with:

```
$ python setup.py install
```

CHAPTER 3

hyperdiary

3.1 hyperdiary package

3.1.1 Submodules

3.1.2 hyperdiary.check module

hyperdiary.check.**check**(*diary: hyperdiary.diary.Diary*) → None

3.1.3 hyperdiary.diary module

exception hyperdiary.diary.**BadEntryException**
Bases: Exception

class hyperdiary.diary.**DateRange**(*start: datetime.date*, *end: datetime.date*)
Bases: object

static from_json(*obj: Mapping[str, str]*) → hyperdiary.diary.DateRange

class hyperdiary.diary.**DayEntry**(*dt: datetime.date, lines: List[str]*)
Bases: object

Diary entry for a specific date, contains multiple lines of text.

iter_lines() → Iterable[Tuple[datetime.date, str]]

class hyperdiary.diary.**Diary**(*hyperdiary_json: Mapping[KT, VT_co]*)
Bases: object

static discover(*subpath: Union[pathlib.Path, str]*) → hyperdiary.diary.Diary

static discover_and_load(*path: Union[pathlib.Path, str] = '.'*) → hyperdiary.diary.Diary

iter_entries_by_year_and_month() → Iterable[Tuple[int, Iterable[Tuple[int, Iterable[hyperdiary.diary.DayEntry]]]]]

iter_lines() → Iterable[Tuple[datetime.date, str]]

load_entries() → None

nested_dicts_by_year_and_month() → Mapping[int, Mapping[int, Mapping[datetime.date, hyperdiary.diary.DayEntry]]]

class hyperdiary.diary.**Token**(*type: hyperdiary.diary.TokenType*, *text: str*, *ref: str = None*)
Bases: object

class hyperdiary.diary.**TokenType**
Bases: enum.Enum

An enumeration.

Id = 3

Tag = 2

Text = 1

hyperdiary.diary.**find_ids**(*line: str*) → Iterable[hyperdiary.diary.Token]

hyperdiary.diary.**find_tags**(*line: str*) → Iterable[hyperdiary.diary.Token]

```
>>> line = "+tag1 +tag2 some content goes here +tag3"
>>> res = [t.text for t in find_tags(line)]
>>> res == ["tag1", "tag2", "tag3"]
True
```

hyperdiary.diary.**find_token_of_type**(*line: str*, *token_type: hyperdiary.diary.TokenType*) → Iterable[hyperdiary.diary.Token]

hyperdiary.diary.**tokenize**(*line: str*) → Iterable[hyperdiary.diary.Token]

3.1.4 hyperdiary.html module

hyperdiary.html.**append_li_a**(*ul: hyperdiary.htmltags.HTMLElement*, *text: str*, *href: str*) → None

hyperdiary.html.**day_to_html**(*current: datetime.date, entry: Iterable[str], localization: hyperdiary.localization.Localization, link_to_id_fn: Callable[[Optional[str]], str] = None*) → hyperdiary.htmltags.HTMLElement

hyperdiary.html.**diary_to_html**(*diary_instance: hyperdiary.diary.Diary*, *fname: str*) → None

hyperdiary.html.**diary_to_html_folder**(*diary_instance: hyperdiary.diary.Diary*, *folder: str*) → None

hyperdiary.html.**path_for_date**(*dt: datetime.date*) → hyperdiary.simplepath.AbsolutePath

hyperdiary.html.**rel_path**(*spath: Union[datetime.date, str, int]*) → hyperdiary.simplepath.RelativePath

hyperdiary.html.**wrap_html_page**(*content: Union[HTMLElement, str], title: str = None, level: int = 0*) → hyperdiary.htmltags.html

hyperdiary.html.**wrap_page**(*body_content: Union[HTMLElement, str], page_title: str = None, encoding: str = 'utf-8', css_references: Iterable[str] = [], inline_css: str = None*) → hyperdiary.htmltags.html

3.1.5 hyperdiary.htmltags module

class hyperdiary.htmltags.**HTMLElement**(**content*, ***attributes*)
Bases: object

append(**items*) → hyperdiary.htmltags.HTMLElement

lazy_render(*indent: str = ''*, *add_indent: str = ''*) → Iterable[str]

lazy_render_attributes() → Iterable[str]

render_compact = False

subelement(*item: hyperdiary.htmltags.HTMLElement*) → hyperdiary.htmltags.HTMLElement

tag = 'div'

write(*fname: str*) → None

class hyperdiary.htmltags.**a**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'a'

class hyperdiary.htmltags.**article**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'article'

class hyperdiary.htmltags.**body**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'body'

class hyperdiary.htmltags.**button**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'button'

class hyperdiary.htmltags.**div**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'div'

class hyperdiary.htmltags.**footer**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'footer'

class hyperdiary.htmltags.**form**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'form'

class hyperdiary.htmltags.**h1**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'h1'

class hyperdiary.htmltags.**h2**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'h2'

class hyperdiary.htmltags.**h3**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'h3'

class hyperdiary.htmltags.**h4**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'h4'

class hyperdiary.htmltags.**head**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'head'

class hyperdiary.htmltags.**header**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'header'

class hyperdiary.htmltags.**hr**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'hr'

class hyperdiary.htmltags.**html**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'html'

class hyperdiary.htmltags.**img**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'img'

class hyperdiary.htmltags.**li**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'li'

class hyperdiary.htmltags.**link**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'link'

class hyperdiary.htmltags.**meta**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'meta'

class hyperdiary.htmltags.**nav**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'nav'

class hyperdiary.htmltags.**ol**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'ol'

class hyperdiary.htmltags.**p**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'p'

class hyperdiary.htmltags.**small**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'small'

class hyperdiary.htmltags.**span**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'span'

class hyperdiary.htmltags.**style**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'style'

class hyperdiary.htmltags.**table**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'table'

class hyperdiary.htmltags.**tbody**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'tbody'

class hyperdiary.htmltags.**td**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'td'

class hyperdiary.htmltags.**th**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'th'

class hyperdiary.htmltags.**thead**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'thead'

class hyperdiary.htmltags.**title**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = True

tag = 'title'

class hyperdiary.htmltags.**tr**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'tr'

class hyperdiary.htmltags.**ul**(**content*, ***attributes*)
Bases: *hyperdiary.htmltags.HTMLElement*

render_compact = False

tag = 'ul'

3.1.6 hyperdiary.hugo module

hyperdiary.hugo.**diary_to_hugo**(*diary_instance: hyperdiary.diary.Diary, folder: Union[pathlib.Path, str]*) → None

hyperdiary.hugo.**nice_date**(*dt: datetime.date*) → str

3.1.7 hyperdiary.localization module

class hyperdiary.localization.**Localization**(*months: List[str] = ['January', 'February', 'March', 'April', 'May', 'June', 'July', 'August', 'September', 'October', 'November', 'December'], days_short: List[str] = ['Mon', 'Tue', 'Wed', 'Thu', 'Fri', 'Sat', 'Sun'], date_fmt: str = '%Y-%m-%d'*)
Bases: object

format_date(*dt: datetime.date*) → str
Format to localized string. >>> l = Localization() >>> l.format_date(date(2019, 11, 3)) '2019-11-03' >>> l = Localization(date_fmt='%d.%m.%Y') >>> l.format_date(date(2019, 11, 3)) '03.11.2019'

get_day_short(*i: int*) → str
Get the localized name of day i (zero-based index starting Monday). >>> l = Localization() >>> l.get_day_short(0) 'Mon' >>> l.get_day_short(6) 'Sun' >>> l.get_day_short(7) Traceback (most recent call last): ... AssertionError >>> Localization(days_short=['Just Mon']) Traceback (most recent call last): ... AssertionError >>> l2 = Localization(days_short=list('1234567')) >>> l2.get_day_short(3) '4' >>> l2.get_day_short(6) '7'

get_month(*i: int*) → str
Get the localized name of month i (zero-based index). >>> l = Localization() >>> l.get_month(0) 'January' >>> l.get_month(11) 'December' >>> l.get_month(12) Traceback (most recent call last): ... AssertionError >>> Localization(months=['Just January']) Traceback (most recent call last): ... AssertionError >>> l2 = Localization(months=list('123456789ond')) >>> l2.get_month(3) '4' >>> l2.get_month(10) 'n'

3.1.8 hyperdiary.simplepath module

class hyperdiary.simplepath.**AbsolutePath**(*path: str*)
Bases: object

exception hyperdiary.simplepath.**InvalidPathError**
Bases: Exception

exception hyperdiary.simplepath.**NotARelativePathError**
Bases: *hyperdiary.simplepath.InvalidPathError*

exception hyperdiary.simplepath.**NotAnAbsolutePathError**
Bases: *hyperdiary.simplepath.InvalidPathError*

class hyperdiary.simplepath.**RelativePath**(*path: str*)
Bases: object

3.1.9 hyperdiary.stats module

hyperdiary.stats.**stats**(*diary: hyperdiary.diary.Diary*) → Dict[str, int]

3.1.10 hyperdiary.tiddlywiki module

class hyperdiary.tiddlywiki.**Tiddler**(*fname: Union[pathlib.Path, str], **fields*)
Bases: object

static from_entry(*dt: datetime.date, entry: Iterable[str], localization: hyperdiary.localization.Localization*) → hyperdiary.tiddlywiki.Tiddler

text

title

to_div() → str

to_tid() → str

hyperdiary.tiddlywiki.**diary_to_tiddlers**(*diary_instance: hyperdiary.diary.Diary*) → Iterator[hyperdiary.tiddlywiki.Tiddler]

hyperdiary.tiddlywiki.**diary_to_tiddlers_export**(*diary_instance: hyperdiary.diary.Diary, tiddler_dir: Union[pathlib.Path, str]*) → None

hyperdiary.tiddlywiki.**diary_to_tiddlywiki_export**(*diary_instance: hyperdiary.diary.Diary, file: Union[pathlib.Path, str], tiddlywiki_base_file: Union[pathlib.Path, str]*) → None

hyperdiary.tiddlywiki.**make_tiddler_filename**(*o: Union[str, datetime.date]*) → str

```
>>> make_tiddler_filename(" Frühstück Ähre Grüße Föhn")
'fruehstueck-aehre-gruesse-foehn.tid'
>>> make_tiddler_filename("a.v-_'üöß' tiddler")
'av-_ueoess-tiddler.tid'
>>> make_tiddler_filename(date(2019, 2, 3))
'2019-02-03.tid'
```

3.1.11 hyperdiary.view module

hyperdiary.view.**view**(*diary: hyperdiary.diary.Diary*, *dt: datetime.date*) → None

3.1.12 Module contents

hyperdiary.**main**() → None

hyperdiary.**parse_date**(*sdate: str*) → datetime.date

CHAPTER 4

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/luphord/hyperdiary/issues.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

4.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with "bug" and "help wanted" is open to whoever wants to implement it.

4.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with "enhancement" and "help wanted" is open to whoever wants to implement it.

4.1.4 Write Documentation

hyperdiary could always use more documentation, whether as part of the official hyperdiary docs, in docstrings, or even on the web in blog posts, articles, and such.

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/luphord/hyperdiary/issues.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

4.2 Get Started!

Ready to contribute? Here's how to set up *hyperdiary* for local development.

1. Fork the *hyperdiary* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/hyperdiary.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv hyperdiary
$ cd hyperdiary/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

   Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 hyperdiary tests
$ python setup.py test or py.test
$ tox
```

   To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

4.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.
3. The pull request should work for Python 3.5, 3.6 and 3.7. Check https://travis-ci.com/github/luphord/hyperdiary/pull_requests and make sure that the tests pass for all supported Python versions.

4.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_hyperdiary
```

4.5 Deploying

A reminder for the maintainers on how to deploy. Make sure all your changes are committed (including an entry in HISTORY.rst). Then run:

```
$ bumpversion patch # possible: major / minor / patch
$ git push
$ git push --tags
```

Travis will then deploy to PyPI if tests pass.

CHAPTER 5

Credits

5.1 Development Lead

- luphord <luphord@protonmail.com>

5.2 Contributors

None yet. Why not be the first?

CHAPTER 6

History

6.1 0.6.1 (2020-10-21)

- add support for Python 3.8
- upgrade to flake8 3.8.4 and fix linter issues
- replace deprecated test command
- move to travis-ci.com

6.2 0.6.0 (2019-11-29)

- promote status to alpha
- introduced localization using hyperdiary.json; date format and month names so far
- added tiddlers for month (calendar) and year overview with hyperlinks
- Diary supports iteration by year/month
- make docs work

6.3 0.5.0 (2019-11-06)

- removed EntryType, refactored iteration, primarily passing Diary instances around
- fix bug when an ID is followed by a bracket)
- only line entries are allowed, no more dicts

6.4 0.4.0 (2019-10-30)

- full type annotations added for package
- linting and type checking in CI
- EntryTypes Dict and DictLine are deprecated and will be removed in 0.5 (check issues a warning)

6.5 0.3.2 (2019-10-17)

- fix linter configuration (and activate it)
- make all hyperdiary source files PEP8 conform
- set up bumpversion config

6.6 0.3.1 (2019-07-03)

- open date ranges iterate until yesterday, not today
- fix loading hyperdiary.json without expected field

6.7 0.3.0 (2019-06-19)

- Expected date ranges for check subcommand can be specified
- subcommand for export to tiddlywiki
- Tests for various output formats

6.8 0.2.0 (2019-06-12)

- stats subcommand can be invoked for all entries
- Rename *tiddlywiki* subcommand to *tiddlers* for different future usage
- Integrate Travis CI for automated testing
- Add basic tests
- Refactor diary/entry passing to subcommands

6.9 0.1.2 (2019-06-11)

- Fix htmlfolder subcommand by packaging picnic.min.css
- Fix view subcommand by using correct relative import
- Avoid deprecation warning in stats subcommand by using yaml.SafeLoader

6.10 0.1.1 (2019-04-29)

- Fix setup

6.11 0.1.0 (2019-04-28)

- First release on PyPI
- Import from fossil

CHAPTER 7

Indices and tables

- genindex
- modindex
- search

Python Module Index

h

hyperdiary, 12
hyperdiary.check, 5
hyperdiary.diary, 5
hyperdiary.html, 6
hyperdiary.htmltags, 7
hyperdiary.hugo, 10
hyperdiary.localization, 10
hyperdiary.simplepath, 11
hyperdiary.stats, 11
hyperdiary.tiddlywiki, 11
hyperdiary.view, 12

Index

A

a (class in hyperdiary.htmltags), 7
AbsolutePath (class in hyperdiary.simplepath), 11
append() (hyperdiary.htmltags.HTMLElement method), 7
append_li_a() (in module hyperdiary.html), 6
article (class in hyperdiary.htmltags), 7

B

BadEntryException, 5
body (class in hyperdiary.htmltags), 7
button (class in hyperdiary.htmltags), 7

C

check() (in module hyperdiary.check), 5

D

DateRange (class in hyperdiary.diary), 5
day_to_html() (in module hyperdiary.html), 6
DayEntry (class in hyperdiary.diary), 5
Diary (class in hyperdiary.diary), 5
diary_to_html() (in module hyperdiary.html), 6
diary_to_html_folder() (in module hyperdiary.html), 6
diary_to_hugo() (in module hyperdiary.hugo), 10
diary_to_tiddlers() (in module hyperdiary.tiddlywiki), 11
diary_to_tiddlers_export() (in module hyperdiary.tiddlywiki), 11
diary_to_tiddlywiki_export() (in module hyperdiary.tiddlywiki), 12
discover() (hyperdiary.diary.Diary static method), 5
discover_and_load() (hyperdiary.diary.Diary static method), 5
div (class in hyperdiary.htmltags), 7

F

find_ids() (in module hyperdiary.diary), 6
find_tags() (in module hyperdiary.diary), 6
find_token_of_type() (in module hyperdiary.diary), 6
footer (class in hyperdiary.htmltags), 7
form (class in hyperdiary.htmltags), 7
format_date() (hyperdiary.localization.Localization method), 10
from_entry() (hyperdiary.tiddlywiki.Tiddler static method), 11
from_json() (hyperdiary.diary.DateRange static method), 5

G

get_day_short() (hyperdiary.localization.Localization method), 11
get_month() (hyperdiary.localization.Localization method), 11

H

h1 (class in hyperdiary.htmltags), 7
h2 (class in hyperdiary.htmltags), 8
h3 (class in hyperdiary.htmltags), 8
h4 (class in hyperdiary.htmltags), 8
head (class in hyperdiary.htmltags), 8
header (class in hyperdiary.htmltags), 8
hr (class in hyperdiary.htmltags), 8
html (class in hyperdiary.htmltags), 8
HTMLElement (class in hyperdiary.htmltags), 7
hyperdiary (module), 12
hyperdiary.check (module), 5
hyperdiary.diary (module), 5
hyperdiary.html (module), 6
hyperdiary.htmltags (module), 7
hyperdiary.hugo (module), 10
hyperdiary.localization (module), 10
hyperdiary.simplepath (module), 11
hyperdiary.stats (module), 11
hyperdiary.tiddlywiki (module), 11
hyperdiary.view (module), 12

I

Id (hyperdiary.diary.TokenType attribute), 6
img (class in hyperdiary.htmltags), 8
InvalidPathError, 11

iter_entries_by_year_and_month() (hyperdiary.diary.Diary method), 5
iter_lines() (hyperdiary.diary.DayEntry method), 5
iter_lines() (hyperdiary.diary.Diary method), 5

L

lazy_render() (hyperdiary.htmltags.HTMLElement method), 7
lazy_render_attributes() (hyperdiary.htmltags.HTMLElement method), 7
li (class in hyperdiary.htmltags), 8
link (class in hyperdiary.htmltags), 9
load_entries() (hyperdiary.diary.Diary method), 6
Localization (class in hyperdiary.localization), 10

M

main() (in module hyperdiary), 12
make_tiddler_filename() (in module hyperdiary.tiddlywiki), 12
meta (class in hyperdiary.htmltags), 9

N

nav (class in hyperdiary.htmltags), 9
nested_dicts_by_year_and_month() (hyperdiary.diary.Diary method), 6
nice_date() (in module hyperdiary.hugo), 10
NotAnAbsolutePathError, 11
NotARelativePathError, 11

O

ol (class in hyperdiary.htmltags), 9

P

p (class in hyperdiary.htmltags), 9
parse_date() (in module hyperdiary), 12
path_for_date() (in module hyperdiary.html), 6

R

rel_path() (in module hyperdiary.html), 6
RelativePath (class in hyperdiary.simplepath), 11
render_compact (hyperdiary.htmltags.a attribute), 7
render_compact (hyperdiary.htmltags.article attribute), 7
render_compact (hyperdiary.htmltags.body attribute), 7
render_compact (hyperdiary.htmltags.button attribute), 7
render_compact (hyperdiary.htmltags.div attribute), 7
render_compact (hyperdiary.htmltags.footer attribute), 7
render_compact (hyperdiary.htmltags.form attribute), 7
render_compact (hyperdiary.htmltags.h1 attribute), 8
render_compact (hyperdiary.htmltags.h2 attribute), 8
render_compact (hyperdiary.htmltags.h3 attribute), 8
render_compact (hyperdiary.htmltags.h4 attribute), 8
render_compact (hyperdiary.htmltags.head attribute), 8
render_compact (hyperdiary.htmltags.header attribute), 8
render_compact (hyperdiary.htmltags.hr attribute), 8
render_compact (hyperdiary.htmltags.html attribute), 8
render_compact (hyperdiary.htmltags.HTMLElement attribute), 7
render_compact (hyperdiary.htmltags.img attribute), 8
render_compact (hyperdiary.htmltags.li attribute), 8
render_compact (hyperdiary.htmltags.link attribute), 9
render_compact (hyperdiary.htmltags.meta attribute), 9
render_compact (hyperdiary.htmltags.nav attribute), 9
render_compact (hyperdiary.htmltags.ol attribute), 9
render_compact (hyperdiary.htmltags.p attribute), 9
render_compact (hyperdiary.htmltags.small attribute), 9
render_compact (hyperdiary.htmltags.span attribute), 9
render_compact (hyperdiary.htmltags.style attribute), 9
render_compact (hyperdiary.htmltags.table attribute), 9
render_compact (hyperdiary.htmltags.tbody attribute), 9
render_compact (hyperdiary.htmltags.td attribute), 10
render_compact (hyperdiary.htmltags.th attribute), 10
render_compact (hyperdiary.htmltags.thead attribute), 10
render_compact (hyperdiary.htmltags.title attribute), 10
render_compact (hyperdiary.htmltags.tr attribute), 10
render_compact (hyperdiary.htmltags.ul attribute), 10

S

small (class in hyperdiary.htmltags), 9
span (class in hyperdiary.htmltags), 9
stats() (in module hyperdiary.stats), 11
style (class in hyperdiary.htmltags), 9
subelement() (hyperdiary.htmltags.HTMLElement method), 7

T

table (class in hyperdiary.htmltags), 9
Tag (hyperdiary.diary.TokenType attribute), 6
tag (hyperdiary.htmltags.a attribute), 7
tag (hyperdiary.htmltags.article attribute), 7
tag (hyperdiary.htmltags.body attribute), 7
tag (hyperdiary.htmltags.button attribute), 7
tag (hyperdiary.htmltags.div attribute), 7
tag (hyperdiary.htmltags.footer attribute), 7
tag (hyperdiary.htmltags.form attribute), 7
tag (hyperdiary.htmltags.h1 attribute), 8
tag (hyperdiary.htmltags.h2 attribute), 8
tag (hyperdiary.htmltags.h3 attribute), 8
tag (hyperdiary.htmltags.h4 attribute), 8
tag (hyperdiary.htmltags.head attribute), 8
tag (hyperdiary.htmltags.header attribute), 8
tag (hyperdiary.htmltags.hr attribute), 8
tag (hyperdiary.htmltags.html attribute), 8
tag (hyperdiary.htmltags.HTMLElement attribute), 7
tag (hyperdiary.htmltags.img attribute), 8
tag (hyperdiary.htmltags.li attribute), 8
tag (hyperdiary.htmltags.link attribute), 9

tag (hyperdiary.htmltags.meta attribute), 9
tag (hyperdiary.htmltags.nav attribute), 9
tag (hyperdiary.htmltags.ol attribute), 9
tag (hyperdiary.htmltags.p attribute), 9
tag (hyperdiary.htmltags.small attribute), 9
tag (hyperdiary.htmltags.span attribute), 9
tag (hyperdiary.htmltags.style attribute), 9
tag (hyperdiary.htmltags.table attribute), 9
tag (hyperdiary.htmltags.tbody attribute), 10
tag (hyperdiary.htmltags.td attribute), 10
tag (hyperdiary.htmltags.th attribute), 10
tag (hyperdiary.htmltags.thead attribute), 10
tag (hyperdiary.htmltags.title attribute), 10
tag (hyperdiary.htmltags.tr attribute), 10
tag (hyperdiary.htmltags.ul attribute), 10
tbody (class in hyperdiary.htmltags), 9
td (class in hyperdiary.htmltags), 10
Text (hyperdiary.diary.TokenType attribute), 6
text (hyperdiary.tiddlywiki.Tiddler attribute), 11
th (class in hyperdiary.htmltags), 10
thead (class in hyperdiary.htmltags), 10
Tiddler (class in hyperdiary.tiddlywiki), 11
title (class in hyperdiary.htmltags), 10
title (hyperdiary.tiddlywiki.Tiddler attribute), 11
to_div() (hyperdiary.tiddlywiki.Tiddler method), 11
to_tid() (hyperdiary.tiddlywiki.Tiddler method), 11
Token (class in hyperdiary.diary), 6
tokenize() (in module hyperdiary.diary), 6
TokenType (class in hyperdiary.diary), 6
tr (class in hyperdiary.htmltags), 10

U

ul (class in hyperdiary.htmltags), 10

V

view() (in module hyperdiary.view), 12

W

wrap_html_page() (in module hyperdiary.html), 6
wrap_page() (in module hyperdiary.html), 6
write() (hyperdiary.htmltags.HTMLElement method), 7