when

Release 0.11.1

unknown

Feb 20, 2022

CONTENTS

1 Installation 3

2 Supports 5

3 Docs & Source 7

4 Methods 9

5 Indices and tables 11

Python Module Index 13

Index 15

A wrapper library for date/time conversion. Takes many of the standard ISO date formats as strings and creates an internally consistent datetime object.

CHAPTER

ONE

INSTALLATION

```
$ pip install pywhen
```

CHAPTER

TWO

SUPPORTS

pywhen has been tested with Python 3.6, 3.7, 3.8, 3.9, and 3.10

Previously it was tested with Python 2.7, not much has changed since then and it should still work, but it isn't part of the automatic tests anymore.

CHAPTER

THREE

DOCS & SOURCE

Docs: http://pywhen.readthedocs.io/en/latest/

Source: https://github.com/cltrudeau/pywhen

Version: 0.11.1

CHAPTER

FOUR

METHODS

exception when.**TimeOnlyError**

Exception indicating that a date operation was attempted on a *When* object that only wraps a python time instance.

class when.**When**(***kwargs*)

Date/time conversion utility. A When object wraps either a python datetime or time instance, providing a common mechanism for building and converting them into different formats.

If When is wrapping a time object then some methods will raise a *TimeOnlyError*.

Supported formats for parsing and display:

```
'date': '%Y-%m-%d'
'time': '%H:%M'
'time_sec': '%H:%M:%S'
'datetime': '%Y-%m-%d %H:%M'
'datetime_sec': '%Y-%m-%d %H:%M:%S'
'datetime_utc': '%Y-%m-%dT%H:%MZ'
'datetime_sec_utc': '%Y-%m-%dT%H:%M:%SZ'
'iso_micro': '%Y-%m-%dT%H:%M:%S.%fZ'
```

Warning: All python datetime objects in this class are naive and have no timezone information associated with them, this includes various formats labelled "utc".

__init__(***kwargs*)

Create a When object. The constructor accepts a variety of keywords depending on what type of date or time information you wish to convert. Most keywords result in a python datetime object being wrapped, but certain cases use a time object. If the When is only a time wrapper then some of the methods will not be available.

Parameters

- **datetime** – Create When using a python datetime object
- **date** – Create When using a python date object, can be used in conjunction with the time keyword. If used without the time keyword the time portion will be set to midnight.
- **time** – Create When using a python time object. If used on its own the date based methods of When will not be allowed. This keyword can be used in conjunction with the date keyword to create a fully qualified When.

- **time_string** – Create `When` using a string containing time information. Handles either 'hour:minute' or 'hour:minute:second'. Like the `time` keyword, can be used in conjunction with `date`.
- **epoch** – Create `When` using an integer epoch value
- **milli_epoch** – Create `When` using an integer that is 1000 * an epoch value with the last thousands being milli-epoch.
- **detect** – Create `When` by parsing a string which is compared against the list of available string parsers.
- **parse_*** – Create `When` by parsing a string using the specific *Supported formats* given. For example, `parse_iso_micro` expects the ISO 8601 format.

Raises

- **ValueError** – If a bad string is passed to `detect` or `parse_*` keywords
- **AttributeError** – If the constructor was called without sufficient arguments to result in a date or time being wrapped.

property date
Returns a python `date` object.

property datetime
Returns a python `datetime` object.

property epoch
Returns an integer version of epoch, i.e. the number of seconds since Jan 1, 1970.

property milli_epoch
Returns an int of the epoch * 1000 + milliseconds.

property string
Returns a placeholder object that has an attribute for each one of the *Supported formats*.

Example:

```
>>> When(datetime=d).string.iso_micro
1972-01-31T13:45:00.2
```

property time
Returns a python `time` object.

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

W

when, 9

INDEX

Symbols

__init__() (*when.When method*), 9

D

date() (*when.When property*), 10
datetime() (*when.When property*), 10

E

epoch() (*when.When property*), 10

M

milli_epoch() (*when.When property*), 10
module
    when, 9

S

string() (*when.When property*), 10

T

time() (*when.When property*), 10
TimeOnlyError, 9

W

when
    module, 9
When (*class in when*), 9