



**pyiso8601**

<> Source

Commits

Branches

Pull requests

Pipelines

Deployments

Issues

Downloads

Michael Twomey

pyiso8601

Clone

Many file formats and standards use the ISO 8601 date format (e.g. 2007-01-14T20:34:22+00:00) to store dates in a neutral, unambiguous manner. This simple module parses the most common forms encountered and returns datetime objects.

default Filter files

/

Name	Size	Last commit	Message
docs			
iso8601			
.hgignore	97 B	2017-07-27	Add .python-version (from p
.hgtags	379 B	2017-07-27	Added tag 0.1.12 for change
LICENSE	1.04 KB	2015-11-03	Update README for release
MANIFEST.in	92 B	2013-10-15	Move description into READI
README.rst	7.02 KB	2017-07-27	Remove py32 from tests and
dev-requirements.txt	95 B	2017-07-27	Remove py32 from tests and
setup.py	674 B	2015-11-04	Updating for 0.1.12
tox.ini	144 B	2017-07-27	Remove py32 from tests and

README.rst

Simple module to parse ISO 8601 dates

This module parses the most common forms of ISO 8601 date strings (e.g. 2007-01-14T20:34:22+00:00) into datetime objects.

```
>>> import iso8601
>>> iso8601.parse_date("2007-01-25T12:00:00Z")
datetime.datetime(2007, 1, 25, 12, 0, tzinfo=<iso8601.Utc>)
>>>
```

See the LICENSE file for the license this package is released under.

If you want more full featured parsing look at:

- <http://labix.org/python-dateutil> - python-dateutil

Parsed Formats

You can parse full date + times, or just the date. In both cases a datetime instance is returned but with missing times defaulting to 0, and missing days / months defaulting to 1.

Dates

- YYYY-MM-DD
- YYYYMMDD



pyiso8601

- Source
- Commits
- Branches
- Pull requests
- Pipelines
- Deployments
- Issues
- Downloads

- YYYY-MM (defaults to 1 for the day)
- YYYY (defaults to 1 for month and day)

Times

- hh:mm:ss.nn
- hhmmss.nn
- hh:mm (defaults to 0 for seconds)
- hhmm (defaults to 0 for seconds)
- hh (defaults to 0 for minutes and seconds)

Time Zones

- Nothing, will use the default timezone given (which in turn defaults to UTC).
- Z (UTC)
- +/-hh:mm
- +/-hhmm
- +/-hh

Where it Differs From ISO 8601

Known differences from the ISO 8601 spec:

- You can use a " " (space) instead of T for separating date from time.
- Days and months without a leading 0 (2 vs 02) will be parsed.
- If time zone information is omitted the default time zone given is used (which in turn defaults to UTC). Use a default of None to yield naive datetime instances.

Homepage

- Documentation: <http://pyiso8601.readthedocs.org/>
- Source: <https://bitbucket.org/micktwomey/pyiso8601/>

This was originally hosted at <https://code.google.com/p/pyiso8601/>

References

- http://en.wikipedia.org/wiki/ISO_8601
- <http://www.cl.cam.ac.uk/~mgk25/iso-time.html> - simple overview
- <http://hydracen.com/dx/iso8601.htm> - more detailed enumeration of valid formats.

Testing

1. pip install -r dev-requirements.txt
2. tox

Note that you need all the pythons installed to perform a tox run (see below). pyenv helps hugely, use pyenv install for the versions you need then use 'pyenv local version ...' to link them in (the tox-pyenv plugin will pick them up).

Alternatively, to test only with your current python:

1. pip install -r dev-requirements.txt
2. py.test --verbose iso8601

Supported Python Versions

Tested against:

- Python 2.6
- Python 2.7
- Python 3.3
- Python 3.4
- Python 3.5
- Python 3.6
- PyPy
- PyPy 3



pyiso8601

- Source
- Commits
- Branches
- Pull requests
- Pipelines
- Deployments
- Issues
- Downloads

Python 3.0, 3.1 and 3.2 are untested but should work (tests didn't run under them when last tried).

Jython is untested but should work (tests failed to run).

Python 2.5 is not supported (too old for the tests for the most part). It could work with some small changes but I'm not supporting it.

Changes

Unreleased

- Remove python 3.2 from tests

0.1.12

- Fix class reference for iso8601.Utc in module docstring (thanks to felixschwarz in <https://bitbucket.org/micktwomey/pyiso8601/pull-requests/7/fix-class-reference-for-iso8601utc-in/diff>)

0.1.11

- Remove logging (thanks to Quentin Pradet in <https://bitbucket.org/micktwomey/pyiso8601/pull-requests/6/remove-debug-logging/diff>)
- Add support for , as separator for fractional part (thanks to ecksun in <https://bitbucket.org/micktwomey/pyiso8601/pull-requests/5/add-support-for-as-separator-for/diff>)
- Add Python 3.4 and 3.5 to tox test config.
- Add PyPy 3 to tox test config.
- Link to documentation at <http://pyiso8601.readthedocs.org/>

0.1.10

- Fixes <https://bitbucket.org/micktwomey/pyiso8601/issue/14/regression-yyyy-mm-no-longer-parses> (thanks to Kevin Gill for reporting)
- Adds YYYY as a valid date (uses 1 for both month and day)
- Woo, semantic versioning, .10 at last.

0.1.9

- Lots of fixes tightening up parsing from jdanjou. In particular more invalid cases are treated as errors. Also includes fixes for tests (which is how these invalid cases got in in the first place).
- Release addresses <https://bitbucket.org/micktwomey/pyiso8601/issue/13/new-release-based-on-critical-bug-fix>

0.1.8

- Remove +/- chars from README.rst and ensure tox tests run using LC_ALL=C. The setup.py egg_info command was failing in python 3.* on some setups (basically any where the system encoding wasn't UTF-8). (<https://bitbucket.org/micktwomey/pyiso8601/issue/10/setuppy-broken-for-python-33>) (thanks to klmitch)

0.1.7

- Fix parsing of microseconds (<https://bitbucket.org/micktwomey/pyiso8601/issue/9/regression-parsing-microseconds>) (Thanks to dims and bnemec)

0.1.6

- Correct negative timezone offsets (<https://bitbucket.org/micktwomey/pyiso8601/issue/8/015-parses-negative-timezones-incorrectly>) (thanks to Jonathan Lange)

0.1.5

- Wow, it's alive! First update since 2007
- Moved over to <https://bitbucket.org/micktwomey/pyiso8601>
- Add support for python 3. <https://code.google.com/p/pyiso8601/issues/detail?id=23> (thanks to zefciu)
- Switched to py.test and tox for testing



The sidebar shows the repository name 'pyiso8601' and a list of navigation options: Source (selected), Commits, Branches, Pull requests, Pipelines, Deployments, Issues, and Downloads. At the bottom, there are icons for help (?) and user profile.

- Make seconds optional in date format ("1997-07-16T19:20+01:00" now valid). <https://bitbucket.org/micktwomey/pyiso8601/pull-request/1/make-the-inclusion-of-seconds-optional-in/diff> (thanks to Chris Down)
- Correctly raise `ParseError` for more invalid inputs (<https://bitbucket.org/micktwomey/pyiso8601/issue/1/raise-parseerror-for-invalid-input>) (thanks to manish.tomar)
- Support more variations of ISO 8601 dates, times and time zone specs.
- Fix microsecond rounding issues (<https://bitbucket.org/micktwomey/pyiso8601/issue/2/roundoff-issues-when-parsing-decimal>) (thanks to nielsenb@jetfuse.net)
- Fix pickling and deepcopy of returned datetime objects (https://bitbucket.org/micktwomey/pyiso8601/issue/3/dates-returned-by-parse_date-do-not) (thanks to fogathmann and john@openlearning.com)
- Fix timezone offsets without a separator (<https://bitbucket.org/micktwomey/pyiso8601/issue/4/support-offsets-without-a-separator>) (thanks to joe.walton.gglcd)
- "Z" produces default timezone if one is specified (<https://bitbucket.org/micktwomey/pyiso8601/issue/5/z-produces-default-timezone-if-one-is>) (thanks to vfaronov). This one may cause problems if you've been relying on `default_timezone` to use that timezone instead of UTC. Strictly speaking that was wrong but this is potentially backwards incompatible.
- Handle compact date format (<https://bitbucket.org/micktwomey/pyiso8601/issue/6/handle-compact-date-format>) (thanks to rvandolson@esri.com)

0.1.4

- The `default_timezone` argument wasn't being passed through correctly, UTC was being used in every case. Fixes issue 10.

0.1.3

- Fixed the microsecond handling, the generated microsecond values were way too small. Fixes issue 9.

0.1.2

- Adding `ParseError` to `__all__` in `iso8601` module, allows people to import it. Addresses issue 7.
- Be a little more flexible when dealing with dates without leading zeroes. This violates the spec a little, but handles more dates as seen in the field. Addresses issue 6.
- Allow date/time separators other than T.

0.1.1

- When parsing dates without a timezone the specified default is used. If no default is specified then UTC is used. Addresses issue 4.