Introducing Maya: Datetimes for Humans™

2016

2 min read • 401 words

Datetimes are a headache to deal with in Python, especially when dealing with timezones, especially when dealing with different machines with different locales.

Maya exists to do all the hard work for you, so you can focus on what you're trying to do — import or export simple datetime data in known human and machine-readable formats. This continues the "for Humans" design philosophy of making complex technical tasks accessible through intuitive APIs.

Example Usage of Maya (v0.1.0)

```
>>> now = maya.now()
<MayaDT epoch=1481850660.9>

>>> tomorrow = maya.when('tomorrow')
<MayaDT epoch=1481919067.23>

>>> tomorrow.slang_date()
'tomorrow'

>>> tomorrow.slang_time()
'23 hours from now'

>>> tomorrow.iso8601()
'2016-12-16T15:11:30.263350Z'

>>> tomorrow.rfc2822()
'Fri, 16 Dec 2016 20:11:30 -0000'

>>> tomorrow.datetime()
datetime.datetime(2016, 12, 16, 15, 11, 30, 263350, tzinfo=<UTC>)

# Automatically parse datetime strings and generate naive datetimes.
>>> scraped = '2016-12-16 18:23:45.423992+00:00'
>>> maya.parse(scraped).datetime(to_timezone='US/Eastern', naive=True)
datetime.datetime(2016, 12, 16, 13, 23, 45, 423992)

>>> rand_day = maya.when('2011-02-07', timezone='US/Eastern')
<MayaDT epoch=1297036800.0>

# Note how this is the 6th, not the 7th.
>>> rand_day.day
6

# Always.
>>> rand_day.timezone
'UTC'
```

Why is this useful?

- All timezone algebra will behave identically on all machines, regardless of system locale.
- Complete symmetric import and export of both **ISO 8601** and **RFC 2822** datetime stamps.
- Fantastic parsing of both dates written for/by humans and machines (**maya.when()** vs. **maya.parse()**).
- Support for human slang, both import and export (e.g. 'an hour ago').
- Datetimes can very easily be generated, with or without timezone information attached (naive).
- This library is based around epoch time, but dates before Jan 1 1970 are indeed supported, via negative integers.
- Maya never panics, and always carries a towel.

What about Delorean, Arrow, & Pendulum?

Arrow, for example, is a fantastic library, but isn't what I wanted in a datetime library. In many ways, it's better than Maya for certain things. In some ways, in my opinion, it's not.

I simply desire a sane API for datetimes that made sense to me for all the things I'd ever want to do—especially when dealing with timezone algebra. Arrow doesn't do all of the things I need (but it does a lot more!). Maya does do exactly what I need.

I think these projects complement each-other, personally. Maya is great for parsing websites, for example. Arrow supports floors and ceilings and spans of dates, which Maya does not at all.

Installing Maya

```
$ pip install maya
```

External Links

- Maya on GitHub
- Maya in The Cheeseshop
- Say Thanks™

Generated from kennethreitz.org • 2025