Chronology

Jan 16, 2020

Contents:

CHAPTER 1

chronology module

Tools which help convert structured schedule data into events.

1.1 chronology.events module

Abstractions for some common types of events that Chronology can handle

class chronology.events.**NaiveOccurence**(*start*, *end*)

Bases: object

An event occurrence which consists of a naive datetime for start and end.

The event can be said to be occuring between start and end. Before start, the event is going to happen. After end, the event has happened.

end = None

A naive datetime object representing the beginning of the event

start = None

A naive datetime object representing the beginning of the event

class chronology.events.**NaiveRecurringEvent**(*start_date: datetime.date, end_date: datetime.date, start_time: datetime.time, end_time: datetime.time, days: List[chronology.notation.ISODayOfWeek]*)

Bases: object

A simple event which recurs.

The event occurs on the ISODayOfWeek days listed in days between start_date and end_date at the time between start_time and end_time.

Parameters

- **start_date** – The first date that the event may possibly occur. The first occurrence of the event is only on this date if it is contained in days, otherwise the first occurrence occurs the first time that a day in days is reached after start_date

- **end_date** – The last date that the event may possibly occur.
- **start_time** – The time that each occurrence of the event begins.
- **end_time** – The time that each occurrence of the event ends.
- **days** – The days that the event occurs on.

occurrences
Every occurrence of this event

1.2 chronology.notation module

Abstractions that handle notation of different time values

class chronology.notation.**ISODayOfWeek**
Bases: enum.IntEnum

Represents days of weeks. The values of this enum can be used as an ISO dow value (1-7) if needed

friday = 5

monday = 1

saturday = 6

sunday = 7

thursday = 4

tuesday = 2

wednesday = 3

chronology.notation.**days_from_fields**(*monday*, *tuesday*, *wednesday*, *thursday*, *friday*, *saturday*, *sunday*) → List[chronology.notation.ISODayOfWeek]
Converts seven truthy or falsy fields into ISODayOfWeek values

Given a field for each of Monday through Sunday, each is checked for its truthiness or falsiness. Each truthy field corresponds to a returned ISODayOfWeek value in the list.

chronology.notation.**time_from_24h_notation**(*timespec: str*) → datetime.time
Gets a naive time value from a 24-hour noted time (like 1155 for 11:55)

This function only accepts four-character strings. Trying to pass "155" for 01:55 is not valid. "0155" is valid.

Parameters **timespec** – An integer value of a 24-hour time, such as 1155.

Raises

- **ValueError** – A four-character string was not provided.
- **ValueError** – The four-character string provided was not numeric.

CHAPTER 2

Indices and tables

- genindex
- modindex
- search

Python Module Index

C

Index