
nemweb Documentation

Release 0.1a

Dylan McConnell

Jan 21, 2019

Contents:

| | | |
|----------|----------------------------|----------|
| 1 | nemweb | 3 |
| 1.1 | nemfile_reader | 3 |
| 1.2 | nemweb_current | 3 |
| 1.3 | nemweb_sqlite | 4 |
| 2 | Indices and tables | 7 |
| | Python Module Index | 9 |

This is a python package to directly download and process AEMO files from <http://www.nemweb.com.au/>. Main module within the package downloads the nemweb files and inserts the tables with into a local sqlite database.

The key modules are found below:

1.1 nemfile_reader

reading nemfiles and zipped nemfiles into pandas dataframes

class nemweb.nemfile_reader.ZipFileStreamer(*filename*)

Bases: zipfile.ZipFile

ZipFile subclass, with method to extract ZipFile as byte stream to memory

extract_stream(*member*)

Extract a member from the archive as a byte stream or string stream, using its full name. ‘member’ may be a filename or a ZipInfo object.

nemweb.nemfile_reader.nemfile_reader(*nemfile_object*)

Returns a dict containing a pandas dataframe each table in a nemfile. The fileobject needs to be unzipped csv (nemfile), and can be either a file or an in stream fileobject.

nemweb.nemfile_reader.nemzip_reader(*nemzip_object*)

Returns a dict containing a pandas dataframe each table in a zipped nemfile. The fileobject is needs to be a zipped csv (nemzip), and can be either a file or an in stream fileobject. Function checks there is only one file to unzip, unzips to a nemfile (csv) in memory, and passes nemfile_object to nemfile reader.

nemweb.nemfile_reader.zip_streams(*fileobject*)

Generator that yields each member of a zipfile as a BytesIO stream. Can take a filename or file-like object (BytesIO object) as an argument.

1.2 nemweb_current

Module for downloading data different ‘CURRENT’ nemweb dataset (selected data sets from files from <http://www.nemweb.com.au/Reports/CURRENT>)

Module includes one main superclass for handling generic nemweb current files. A series of namedtuples (stored in global constant DATASETS) contains the relevant data for specific datasets. Datasets included from 'CURRENT' index page:

- TradingIS_Reports
- DispatchIS_Reports
- Dispatch_SCADA
- Next_Day_Dispatch (DISPATCH_UNIT_SOLUTION)
- Next_Day_Actual_Gen (METER_DATA_GEN_DUID)
- ROOFTOP_PV/ACTUAL

`nemweb.nemweb_current.CurrentDataset`
alias of `nemweb.nemweb_current.NemwebCurrentFile`

class `nemweb.nemweb_current.CurrentFileHandler`
Bases: `object`

class for handling 'CURRENT' nemweb files from <http://www.nemweb.com.au> Requires a 'CurrentDataset' namedtuple with following fields:

- `nemweb_name`: the name of the dataset to be download (e.g. Dispatch_SCADA)
- `filename_pattern`: a regex expression to match and a determine datetime from file-name on nemweb. As example, for files in the Dispatch_SCADA dataset (e.g "PUBLIC_DISPATCHSCADA_201806201135_0000000296175732.zip") the regex `file_patten` is `PUBLIC_DISPATCHSCADA_([0-9]{12})_[0-9]{16}.zip`
- the format of the string to strip the datetime from. From the above example, the match returns '201806201135', so the string is `"%Y%m%d%H%M"`,
- the list of tables to insert from each dataset. This is derived from the 2nd and 3rd column in the nemweb dataset. For example, the 2nd column is in Dispatch_SCADA is "DISPATCH" and the 3rd is "SCADA_VALUE" and the name is "DISPATCH_UNIT_SCADA".

Several datasets contain multiple tables. Examples can be found in the DATASETS dict (`nemweb_reader.DATASETS`)

download (*link*)

Downloads nemweb zipfile from link into memory as a `byteIO` object. `nemfile` object is returned from the `byteIO` object

update_data (*dataset*, *print_progress=False*, *start_date=None*, *end_date='30001225'*,
db_name='nemweb_live.db')

Main method to process nemweb dataset - downloads the index page for the dataset - determines date to start downloading from - matches the start date against files in the index - inserts new files into database

`nemweb.nemweb_current.update_datasets` (*datasets*, *print_progress=False*)
function that updates a subset of datasets (as a list) contained in DATASETS

1.3 nemweb_sqlite

interfaces with `sqlite3` database

`nemweb.nemweb_sqlite.insert` (*dataframe*, *table_name*, *db_name='nemweb_live.db'*)

Inserts dataframe into a table (table name) in an `sqlite3` database (`db_name`). Database directory needs to be specified in `config.ini` file


```
nemweb.nemweb_sqlite.start_from(table_name, db_name='nemweb_live.db', times-  
                                tamp_col='SETTLEMENTDATE', start_date=None)
```

Returns a date to start downloading data from. Tries determining latest date from table in database. On fail prompts user to input date.

```
nemweb.nemweb_sqlite.table_latest_record(table_name, db_name='nemweb_live.db', times-  
                                           tamp_col='SETTLEMENTDATE')
```

Returns the latest timestamp from a table in an sqlite3 database as a datetime object.

Timestamp fields in nemweb files usually named "SETTLEMENTDATE". Sometimes INTERVAL_DATETIME is used.

CHAPTER 2

Indices and tables

- `genindex`
- `modindex`
- `search`

n

`nemweb.nemfile_reader`, 3
`nemweb.nemweb_current`, 3
`nemweb.nemweb_sqlite`, 4

C

CurrentDataset (in module nemweb.nemweb_current), 4
CurrentFileHandler (class in nemweb.nemweb_current),
4

D

download() (nemweb.nemweb_current.CurrentFileHandler
method), 4

E

extract_stream() (nemweb.nemfile_reader.ZipFileStreamer
method), 3

I

insert() (in module nemweb.nemweb_sqlite), 4

N

nemfile_reader() (in module nemweb.nemfile_reader), 3
nemweb.nemfile_reader (module), 3
nemweb.nemweb_current (module), 3
nemweb.nemweb_sqlite (module), 4
nemzip_reader() (in module nemweb.nemfile_reader), 3

S

start_from() (in module nemweb.nemweb_sqlite), 4

T

table_latest_record() (in module
nemweb.nemweb_sqlite), 5

U

update_data() (nemweb.nemweb_current.CurrentFileHandler
method), 4
update_datasets() (in module nemweb.nemweb_current),
4

Z

zip_streams() (in module nemweb.nemfile_reader), 3
ZipFileStreamer (class in nemweb.nemfile_reader), 3