
PhiDB Documentation

Release 0.1.0

Andrew MacDonald

Aug 13, 2017

Contents

1	phildb package	3
1.1	Submodules	3
1.2	phildb.console module	3
1.3	phildb.constants module	3
1.4	phildb.create module	3
1.5	phildb.database module	3
1.6	phildb.exceptions module	6
1.7	phildb.log_handler module	7
1.8	phildb.reader module	7
1.9	phildb.writer module	7
1.10	Module contents	8
2	Indices and tables	9
	Python Module Index	11

Contents:

Submodules

phildb.console module

phildb.constants module

phildb.create module

`phildb.create.create` (*tsdb_path*)

`phildb.create.main` ()

phildb.database module

`class phildb.database.PhilDB` (*tsdb_path*)

Bases: object

`add_attribute` (*attribute_id*, *description*)

Define an attribute.

Parameters

- `attribute_id` (*string*) – Identifier of the attribute.
- `description` (*string*) – Description of the attribute.

`add_attribute_value` (*attribute_id*, *value*)

Store an attribute value.

Parameters

- **attribute_id** (*string*) – Identifier of the attribute.
- **value** (*string*) – The attribute value to store.

add_mesurand (*mesurand_short_id, mesurand_long_id, description*)

Create a mesurand entry.

Mesurand being a measurable timeseries type. e.g. Streamflow, Temperature, Rainfall, etc.

Parameters

- **mesurand_short_id** (*string*) – Short identifier of the mesurand.
- **mesurand_long_id** (*string*) – Long identifier of the mesurand.
- **description** (*string*) – Description of the mesurand.

add_source (*source, description*)

Define a source.

Source being the origin of the data. For example the source used in the examples/hrs example is BOM_HRS. Indicated the origin of the data was the Bureau of Meteorology Hydrologic Reference Stations project.

Parameters

- **source** (*string*) – Identifier of the source.
- **description** (*string*) – Description of the source.

add_timeseries (*identifier*)

Create a timeseries entry to be identified by the supplied ID.

Parameters **identifier** (*string*) – Identifier of the timeseries.

add_timeseries_instance (*identifier, freq, initial_metadata, **kwargs*)

Define an instance of a timeseries.

A timeseries instance is a combination of a timeseries, frequency and attributes.

Parameters

- **identifier** (*string*) – Identifier of the timeseries.
- **freq** (*string*) – Data frequency (e.g. 'D' for day, as supported by pandas.)
- **initial_metadata** (*string*) – Store some metadata about this series. Potentially freeform header from a source file about to be loaded.
- ****kwargs** (*kwargs*) – Any additional attributes to attach to the timeseries instance.

get_file_path (*identifier, freq, ftype='tsdb', **kwargs*)

Get a path to a file for a given timeseries instance.

Parameters

- **identifier** (*string*) – Identifier of the timeseries.
- **ftype** (*string*) – File extension to use (i.e. the type of file). (Default='tsdb')

Returns string – Path to file for a timeseries instance identified by the given arguments.

help ()

List methods of the PhilDB class with the first line of their docstring.

list_ids ()

Returns list of timeseries IDs for all timeseries records.

Returns list(string) – Sorted list of timeseries identifiers.

list_measurands ()

Returns list of measurand short IDs for all measurand records.

Returns list(string) – Sorted list of timeseries identifiers.

list_sources ()

Returns list of source IDs for all sources.

Returns list(string) – Sorted list of source identifiers.

list_timeseries_instances (**kwargs)

Returns list of timeseries instances for all instance records.

Can filter by using keyword arguments.

Returns list(string) – Sorted list of timeseries instances.

read (identifier, freq, **kwargs)

Read the entire timeseries record for the requested timeseries instance.

Parameters

- **identifier** (string) – Identifier of the timeseries.
- **freq** (string) – Timeseries data frequency.
- **kwargs** (kwargs) – Attributes to match against timeseries instances (e.g. source, measurand).

Returns pandas.DataFrame – Timeseries data.

read_all (freq, excludes=None, **kwargs)

Read the entire timeseries record for all matching timeseries instances. Optionally exclude timeseries from the final DataFrame by specifying IDs in the exclude argument.

Parameters

- **identifier** (string) – Identifier of the timeseries.
- **freq** (string) – Timeseries data frequency.
- **excludes** (array[string]) – IDs of timeseries to exclude from final DataFrame.
- **kwargs** (kwargs) – Attributes to match against timeseries instances (e.g. source, measurand).

Returns pandas.DataFrame – Timeseries data.

read_dataframe (identifiers, freq, **kwargs)

Read the entire timeseries record for the requested timeseries instances.

Parameters

- **identifiers** (array[string]) – Identifiers of the timeseries to read into a DataFrame.
- **freq** (string) – Timeseries data frequency.
- **kwargs** (kwargs) – Attributes to match against timeseries instances (e.g. source, measurand).

Returns pandas.DataFrame – Timeseries data.

read_log (identifier, freq, as_at_datetime, **kwargs)

Read timeseries record for the requested timeseries instance as it was at specified datetime in the log.

Parameters

- **identifier** (*string*) – Identifier of the timeseries.
- **freq** (*string*) – Timeseries data frequency.
- **as_at_datetime** (*datetime*) – Filter to a timeseries, as available at this specified datetime, from the log.
- **kwargs** (*kwargs*) – Attributes to match against timeseries instances (e.g. source, measurand).

Returns pandas.DataFrame – Timeseries data.

read_metadata (*ts_id, freq, **kwargs*)

Returns the metadata that was associated with an initial TimeseriesInstance.

Parameters **identifier** (*string*) – Identifier of the timeseries.

Returns string – The initial metadata that was recorded on instance creation.

ts_list (***kwargs*)

Returns list of primary ID for all timeseries records.

Parameters **kwargs** (*kwargs*) – Restrict to records associated with this the kwargs attributes supplied. (Optional).

Returns list(string) – Sorted list of timeseries identifiers.

version ()

Returns the version number of the database schema.

Returns string – Schema version.

write (*identifier, freq, ts, **kwargs*)

Write/update timeseries data for existing timeseries.

Parameters

- **identifier** (*string*) – Identifier of the timeseries.
- **freq** (*string*) – Data frequency (e.g. 'D' for day, as supported by pandas.)
- **ts** (*pd.Series*) – Timeseries data to write into the database.

phildb.exceptions module

exception phildb.exceptions.**AlreadyExistsError**

Bases: exceptions.Exception

exception phildb.exceptions.**DataError**

Bases: exceptions.Exception

exception phildb.exceptions.**DuplicateError**

Bases: exceptions.Exception

exception phildb.exceptions.**MissingAttributeError**

Bases: exceptions.Exception

exception phildb.exceptions.**MissingDataError**

Bases: exceptions.Exception

phildb.log_handler module

`class phildb.log_handler.LogHandler(filename, mode)`

FILTERS = <Mock name='mock.Filters()' id='140005188462288'>

create_skeleton ()

Create the skeleton of the log self.hdf5.

read (as_at_datetime)

write (log_entries, operation_datetime)

phildb.reader module

`phildb.reader.read(filename)`

`phildb.reader.read_log(log_file, as_at_datetime)`

phildb.writer module

`phildb.writer.write(tsd_file, ts, freq)`

Smart write.

Will only update existing values where they have changed. Changed existing values are returned in a list.

Parameters

- **tsdb_file** (*string*) – File to write timeseries data into.
- **ts** (*pd.Series*) – Timeseries data to write.
- **freq** (*string*) – Frequency of the data. (e.g. 'D' for daily, '1Min' for minutely). Accepts any string that `pandas.TimeSeries.asfreq` does or 'IRR' for irregular data.

`phildb.writer.write_irregular_data(tsd_file, series)`

Smart write of irregular data.

Will only update existing values where they have changed. Changed existing values are returned in a list.

Parameters

- **tsdb_file** (*string*) – File to write timeseries data into.
- **series** (*pandas.Series*) – Pandas Series of irregular data to write.

`phildb.writer.write_log(log_file, modified, replacement_datetime)`

`phildb.writer.write_regular_data(tsd_file, series)`

Smart write. Expects continuous time series.

Will only update existing values where they have changed. Changed existing values are returned in a list.

Parameters

- **tsdb_file** (*string*) – File to write timeseries data into.
- **series** (*pandas.Series*) – Pandas Series of regular data to write.

Module contents

CHAPTER 2

Indices and tables

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

p

phildb, 8
phildb.constants, 3
phildb.create, 3
phildb.database, 3
phildb.exceptions, 6
phildb.log_handler, 7
phildb.reader, 7
phildb.writer, 7

A

`add_attribute()` (phildb.database.PhilDB method), 3
`add_attribute_value()` (phildb.database.PhilDB method), 3
`add_measurand()` (phildb.database.PhilDB method), 4
`add_source()` (phildb.database.PhilDB method), 4
`add_timeseries()` (phildb.database.PhilDB method), 4
`add_timeseries_instance()` (phildb.database.PhilDB method), 4
`AlreadyExistsError`, 6

C

`create()` (in module `phildb.create`), 3
`create_skeleton()` (phildb.log_handler.LogHandler method), 7

D

`DataError`, 6
`DuplicateError`, 6

F

`FILTERS` (phildb.log_handler.LogHandler attribute), 7

G

`get_file_path()` (phildb.database.PhilDB method), 4

H

`help()` (phildb.database.PhilDB method), 4

L

`list_ids()` (phildb.database.PhilDB method), 4
`list_measurands()` (phildb.database.PhilDB method), 5
`list_sources()` (phildb.database.PhilDB method), 5
`list_timeseries_instances()` (phildb.database.PhilDB method), 5
`LogHandler` (class in `phildb.log_handler`), 7

M

`main()` (in module `phildb.create`), 3

`MissingAttributeError`, 6
`MissingDataError`, 6

P

`PhilDB` (class in `phildb.database`), 3
`phildb` (module), 8
`phildb.constants` (module), 3
`phildb.create` (module), 3
`phildb.database` (module), 3
`phildb.exceptions` (module), 6
`phildb.log_handler` (module), 7
`phildb.reader` (module), 7
`phildb.writer` (module), 7

R

`read()` (in module `phildb.reader`), 7
`read()` (phildb.database.PhilDB method), 5
`read()` (phildb.log_handler.LogHandler method), 7
`read_all()` (phildb.database.PhilDB method), 5
`read_dataframe()` (phildb.database.PhilDB method), 5
`read_log()` (in module `phildb.reader`), 7
`read_log()` (phildb.database.PhilDB method), 5
`read_metadata()` (phildb.database.PhilDB method), 6

T

`ts_list()` (phildb.database.PhilDB method), 6

V

`version()` (phildb.database.PhilDB method), 6

W

`write()` (in module `phildb.writer`), 7
`write()` (phildb.database.PhilDB method), 6
`write()` (phildb.log_handler.LogHandler method), 7
`write_irregular_data()` (in module `phildb.writer`), 7
`write_log()` (in module `phildb.writer`), 7
`write_regular_data()` (in module `phildb.writer`), 7