rdbtools3 Documentation

Release 0.1.2

Alexey Popravka

October 02, 2015

Contents

1 Installation **3**

2 Source code **5**

3 Getting started **7**

4 Indices and tables **9**

4.1 `rdbtools3` — Parser API 9

4.2 `rdbtools3.cli` — Command-line interface (cli) 10

Python Module Index **11**

Redis dumps parser and cli tool.

CHAPTER 1

Installation

The easiest way to install rdbtools3 is by using the package on PyPi:

```
pip install rdbtools3
```

CHAPTER 2

Source code

The project is hosted on GitHub and can be installed running following commands:

```
git clone https://github.com/popravich/rdbtools3.git
cd rdbtools3
python3 setup.py install

# optionally, run tests with
make test
```

Fill free to fork it and/or to report any found issues/ideas on bug tracker

CHAPTER 3

Getting started

The simpliest way to start using tool:

```
from rdbtools3 import parse_rdb_stream

with open('/path/to/redis/dump.rdb', 'rb') as f:
   for item in parse_rdb_stream(f):
      print(item.key, item.value)
```

And on command line:

```
python3 -m rdbtools3 /path/to/redis/dump.rdb
```

CHAPTER 4

Indices and tables

- genindex
- modindex
- search

4.1 rdbtools3 — Parser API

4.1.1 Parser

rdbtools3.**parse_rdb_stream**(*f*, *skip_db=lambda dbnum: False*, *skip_key_type=lambda dbnum, key_type: False*, *skip_key=lambda dbnum, key_type, key: False*)

Parameters

- **f** (*file*) – File stream
- **skip_db** (*function*) – callback to check if db should be skipped.
- **skip_key_type** (*function*) – callback to check if key type should be skipped.
- **skip_key** (*function*) – callback to check if key should be skipped.

Returns generator that yields *RDBItem*'s

Return type generator

Raises

- ***rdbtools3.FileFormatError*** – if read invalid magic string, unsupported RDB version or no "select db" code found.
- ***rdbtools3.RDBValueError*** – if parser encounters unexpected data value.

Parses Redis dump file stream.

4.1.2 RDBItem

class rdbtools3.**RDBItem**(*dbnum*, *key_type*, *key*, *value*, *expire*, *info*)

Parameters

- **dbnum** (*int*) – database number
- **key_type** (*str*) – key type; possible values *string*, *list*, *set*, *zset*, *hash*

- **key** (*bytes*) – Key name
- **value** (*bytes*) – Value itself
- **expire** (*int or None*) – TTL if set
- **info** (*dict*) – additional key info

Named tuple representing Redis DB item.

4.1.3 Exceptions

exception rdbtools3.**ParserError**
Base parser exception type.

Subclass of ValueError.

exception rdbtools3.**FileFormatError**(*msg*)
Raised if magic string is invalid, RDB version number invalid or not supported or no "select db" control code found.

exception rdbtools3.**RDBValueError**(*msg*)
Raised if unexpected value received.

4.2 rdbtools3.cli — Command-line interface (cli)

Note: CLI is still being developed so documentation yet to come.

You can always get help running *python3 -m rdbtools3.cli –help*

r

rdbtools3, 9

F

FileFormatError, 10

P

parse_rdb_stream() (in module rdbtools3), 9
ParserError, 10

R

RDBItem (class in rdbtools3), 9
rdbtools3 (module), 9
RDBValueError, 10