
sync

Release 2022

Kapa Kudaibergenov

Dec 27, 2022

CONTENTS:

1	sync	1
1.1	main module	1
1.2	sync package	1
1.2.1	Submodules	1
1.2.2	sync.run module	1
1.2.3	sync.syncer module	2
1.2.4	Module contents	2
2	Indices and tables	3
	Python Module Index	5
	Index	7

1.1 main module

Takes all input arguments. Schedules the synchronization with the input interval.

This is a Python package that synchronizes two folders: source and replica. The program maintains a full, identical copy of source folder at replica folder.

To implement it you should be in the folder ‘veeam_task’ and write into console command “python main.py “sourcedir” “targetdir” interval “log_file_path” “.

positional arguments:

param sourcedir

Location of the source folder.

param targetdir

Location of the replica folder. If the folder doesn’t exist at this directory, it will be created.

param interval

Synchronization interval. Stays for with what frequency synchronization should be repeated.
Float argument.

param log_file_path

Log file path. If the file doesn’t exist, it will be created.

optional arguments

- param -h, -help: Show this help message and exit.
- param time_units {s,m,h}: Time measurement units. Possible values: s for seconds, m for minutes,h for hours. Default value is seconds.

1.2 sync package

1.2.1 Submodules

1.2.2 sync.run module

Run function to call to the Syncer class and to make report.

`sync.run.args_parser()`

Parser of input arguments. Returns all input arguments from command line.

`sync.run.synchronize(sourcedir, targetdir, interval, log_file_path)`

Calls to Syncer class, makes comparison, copies content and outputs logs.

Parameters

- **sourcedir** – the source folder location
- **targetdir** – the replica folder location. Will be created if doesn't exist.
- **interval** – synchronization interval
- **log_file_path** – the log file location. Will be created if doesn't exist.

1.2.3 sync.syncer module

class `sync.syncer.DCMP(l, r, c)`

Bases: `object`

Class to store unique values in source and replica folders, and common files in these folders

class `sync.syncer.Syncer(dir1, dir2, interval, log_file_path)`

Bases: `object`

Modifies the replica folder to exactly match content of the source folder.

Parameters

- **sourcedir** – the source folder location
- **targetdir** – the replica folder location. Will be created if doesn't exist.
- **interval** – synchronization interval
- **log_file_path** – the log file location. Will be created if doesn't exist.

log(msg="")

Reference to the logger.

print_logs()

Outputs logs.

sync()

Sync function.

sync_work()

Creates new directory if it doesn't exist. Calls to synchronization function. Takes starting and ending time.

1.2.4 Module contents

INDICES AND TABLES

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

PYTHON MODULE INDEX

m

`main`, 1

s

`sync`, 2

`sync.run`, 1

`sync.syncer`, 2

INDEX

A

`args_parser()` (*in module sync.run*), 1

D

`DCMP` (*class in sync.syncer*), 2

L

`log()` (*sync.syncer.Syncer method*), 2

M

`main`

module, 1

`module`

main, 1

sync, 2

sync.run, 1

sync.syncer, 2

P

`print_logs()` (*sync.syncer.Syncer method*), 2

S

`sync`

module, 2

`sync()` (*sync.syncer.Syncer method*), 2

`sync.run`

module, 1

`sync.syncer`

module, 2

`sync_work()` (*sync.syncer.Syncer method*), 2

`Syncer` (*class in sync.syncer*), 2

`synchronize()` (*in module sync.run*), 1