
persistentdatatools v2.2.13

Release 2.2.13

Aug 31, 2021

Contents:

1	persistentdatatools	1
1.1	persistentdatatools package	1
2	Indices and tables	7
	Python Module Index	9
	Index	11

1.1 persistentsdatatools package

1.1.1 Submodules

1.1.2 persistentsdatatools.persistentsdatatools module

Updated By: Benjamin P. Trachtenberg Date Written 9/17/2015

Description: Some quick tools to make persistent data quicker

```
persistentsdatatools.persistentsdatatools.collect_and_zip_files(dir_list,      out-  
                                                                put_dir,  
                                                                zip_file_name,  
                                                                file_extension_list=None,  
                                                                file_name_list=None)
```

Function to collect files and make a zip file

Parameters

- **dir_list** (*List*) – A list of directories
- **output_dir** (*String*) – The output directory
- **zip_file_name** (*String*) – Zip file name
- **file_extension_list** (*List*) – A list of extensions of files to find
- **file_name_list** (*List*) – A list of file names to find

Return type None

Returns None Outputs a zip file

Note: If no file_extension_list and file_name_list are provided it will zip the entire directory.

```
persistentsdatatools.persistentsdatatools.csv_to_dict(file_name,file_location)
```

Function to import a csv as a dictionary

Parameters

- **file_name** (*String*) – The name of file to be import
- **file_location** (*String*) – The location of the file, derive from the os module

Return type Dict

Returns A dictionary

```
persistentdatatools.persistentdatatools.dict_to_csv(orig_dict, file_name,
                                                    field_names_tuple,
                                                    file_location)
```

Function to export a dictionary to a csv file

Parameters

- **orig_dict** (*Dict*) – The dictionary you want exported
- **file_name** (*String*) – The name of the exported file
- **field_names_tuple** (*Tuple*) – The fieldnames in a tuple
- **file_location** (*String*) – The location of the file, derive from the os module

Return type String

Returns Filename info

```
persistentdatatools.persistentdatatools.file_name_increase(file_name,
                                                            file_location)
```

Function to increase a filename by a number 1

Parameters

- **file_name** (*String*) – The name of file to check
- **file_location** (*String*) – The location of the file, derive from the os module

Return type String

Returns a good filename.

Raises **Exception** – If any errors happen

```
persistentdatatools.persistentdatatools.file_to_list(file_name, file_location)
```

Function to import a text file to a list

Parameters

- **file_name** (*String*) – The name of file to be import
- **file_location** (*String*) – The location of the file, derive from the os module

Return type List

Returns A list created from file data

```
persistentdatatools.persistentdatatools.join_split_string(split_string)
```

Function to join a split string

Parameters **split_string** (*List*) – A Split String

Return type String

Returns A joined string

```
persistentdatatools.persistentdatatools.list_directories_in_directory(full_directory_path)
```

List the directories in a specified directory

Parameters `full_directory_path (String)` – The full directory path to check, derive from the os module

Return type List

Returns A list of directories

`persistentdatatools.persistentdatatools.list_files_in_directory (full_directory_path)`

List the files in a specified directory

Parameters `full_directory_path (String)` – The full directory path to check, derive from the os module

Return type List

Returns A list of files

`persistentdatatools.persistentdatatools.list_to_file (orig_list, file_name, file_location)`

Function to export a list to a text file

Parameters

- **orig_list** (*List*) – The list you want exported
- **file_name** (*String*) – The name of the exported file
- **file_location** (*String*) – The location of the file, derive from the os module

Return type String

Returns Filename info

`persistentdatatools.persistentdatatools.random_data (line_count=1, chars_per_line=80)`

Function to creates lines of random string data

Parameters

- **line_count** (*Integer*) – An integer that says how many lines to return
- **chars_per_line** (*Integer*) – An integer that says how many characters per line to return

Return type String

Returns A String

`persistentdatatools.persistentdatatools.random_line_data (chars_per_line=80)`

Function to create a line of a random string

Parameters `chars_per_line (Integer)` – An integer that says how many characters to return

Return type String

Returns A String

`persistentdatatools.persistentdatatools.remove_extra_spaces (string_item)`

Remove all extra spaces from a string leaving single spaces

Parameters `string_item (String)` – String that you want to remove spaces from

Return type String

Returns Corrected string without any extra spaces

`persistentdatatools.persistentdatatools.remove_spaces (string_item)`

Remove all spaces from a string

Parameters **string_item** (*String*) – String that you want to remove spaces from

Return type String

Returns Corrected string without any spaces

`persistentdatatools.persistentdatatools.remove_spaces_add_hyphen(string_item)`

Remove all spaces from a string and replace them with a hyphen

Parameters **string_item** (*String*) – String that you want to remove hyphens from

Return type String

Returns Corrected string without any hyphens

`persistentdatatools.persistentdatatools.remove_symbol_add_symbol(string_item,
re-
move_symbol,
add_symbol)`

Remove a symbol from a string, and replace it with a different one

Parameters

- **string_item** (*String*) – String that you want to replace symbols in
- **remove_symbol** (*String*) – Symbol to remove
- **add_symbol** (*String*) – Symbol to add

Return type String

Returns Corrected string with symbols swapped

`persistentdatatools.persistentdatatools.split_string_retain_spaces(string)`

Function to split a string, and retain spaces to rejoin

Parameters **string** (*String*) – A String

Return type List

Returns A split string

`persistentdatatools.persistentdatatools.split_strings_in_list_retain_spaces(orig_list)`

Function to split every line in a list, and retain spaces for a rejoin

Parameters **orig_list** (*List*) – Original list

Return type List

Returns A List with split lines

`persistentdatatools.persistentdatatools.verify_directory(directory_name, direc-
tory_location, direc-
tory_create=False)`

Function to verify if a directory exists

Parameters

- **directory_name** (*String*) – The name of directory to check
- **directory_location** (*String*) – The location of the directory, derive from the os module
- **directory_create** (*Boolean*) – If you want to create the directory

Return type Boolean

Returns Boolean True or False, but if you set `directory_create` to True it will create the directory

`persistentsdatatools.persistentsdatatools.verify_file_exists` (*file_name*,
file_location)

Function to verify if a file exists

Parameters

- **file_name** (*String*) – The name of file to check
- **file_location** (*String*) – The location of the file, derive from the os module

Return type Boolean

Returns returns boolean True or False

1.1.3 Module contents

CHAPTER 2

Indices and tables

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

p

`persistentdatatools`, [5](#)

`persistentdatatools.persistentdatatools`,

[1](#)

C

`collect_and_zip_files()` (in module `persistentdatatools.persistentdatatools`), 1

`csv_to_dict()` (in module `persistentdatatools.persistentdatatools`), 1

D

`dict_to_csv()` (in module `persistentdatatools.persistentdatatools`), 2

F

`file_name_increase()` (in module `persistentdatatools.persistentdatatools`), 2

`file_to_list()` (in module `persistentdatatools.persistentdatatools`), 2

J

`join_split_string()` (in module `persistentdatatools.persistentdatatools`), 2

L

`list_directories_in_directory()` (in module `persistentdatatools.persistentdatatools`), 2

`list_files_in_directory()` (in module `persistentdatatools.persistentdatatools`), 3

`list_to_file()` (in module `persistentdatatools.persistentdatatools`), 3

P

`persistentdatatools` (module), 5

`persistentdatatools.persistentdatatools` (module), 1

R

`random_data()` (in module `persistentdatatools.persistentdatatools`), 3

`random_line_data()` (in module `persistentdatatools.persistentdatatools`), 3

`remove_extra_spaces()` (in module `persistentdatatools.persistentdatatools`), 3

`remove_spaces()` (in module `persistentdatatools.persistentdatatools`), 3

`remove_spaces_add_hyphen()` (in module `persistentdatatools.persistentdatatools`), 4

`remove_symbol_add_symbol()` (in module `persistentdatatools.persistentdatatools`), 4

S

`split_string_retain_spaces()` (in module `persistentdatatools.persistentdatatools`), 4

`split_strings_in_list_retain_spaces()` (in module `persistentdatatools.persistentdatatools`), 4

V

`verify_directory()` (in module `persistentdatatools.persistentdatatools`), 4

`verify_file_exists()` (in module `persistentdatatools.persistentdatatools`), 4