

DataPipeline

Generated by Doxygen 1.8.17

1 Usage:	1
2 Namespace Index	3
2.1 Packages	3
3 Hierarchical Index	5
3.1 Class Hierarchy	5
4 Class Index	7
4.1 Class List	7
5 File Index	9
5.1 File List	9
6 Namespace Documentation	11
6.1 cleaner Namespace Reference	11
6.1.1 Function Documentation	11
6.1.1.1 check_correlated_column()	11
6.1.1.2 check_outliers()	12
6.1.1.3 create_correlation_graphs()	12
6.1.1.4 get_correlated_descriptors()	12
6.1.1.5 get_duplicit_correlated_descriptors()	12
6.1.1.6 load_data()	12
6.1.1.7 main()	12
6.1.1.8 process_config()	12
6.1.1.9 remove_duplicits()	13
6.1.1.10 remove_useless_columns()	13
6.1.1.11 save_mask()	13
6.1.1.12 set_proper_data_type()	13
6.1.1.13 unique_count()	13
6.1.2 Variable Documentation	13
6.1.2.1 args	13
6.2 crawler_orchestrator Namespace Reference	13
6.2.1 Function Documentation	13
6.2.1.1 main()	13
6.3 dependencies Namespace Reference	14
6.4 dependencies.pubchem Namespace Reference	14
6.4.1 Function Documentation	14
6.4.1.1 crawl()	14
6.5 dependencies.swissadme Namespace Reference	14
6.5.1 Function Documentation	14
6.5.1.1 crawl()	14
6.5.1.2 is_file_downloaded()	14
6.6 dependencies.swisstarget Namespace Reference	14

6.6.1 Function Documentation	14
6.6.1.1 crawl()	15
6.6.1.2 is_file_downloaded()	15
6.7 merger Namespace Reference	15
6.7.1 Function Documentation	15
6.7.1.1 merge()	15
6.8 pipe_handler Namespace Reference	15
6.9 populate_chem Namespace Reference	15
6.9.1 Function Documentation	15
6.9.1.1 populate()	15
6.10 run_pipeline Namespace Reference	15
6.10.1 Variable Documentation	16
6.10.1.1 args	16
6.10.1.2 help	16
6.10.1.3 parser	16
6.10.1.4 str	16
6.10.1.5 type	16
7 Class Documentation	17
7.1 pipe_handler.IgnoreBrokenPipe Class Reference	17
7.1.1 Constructor & Destructor Documentation	18
7.1.1.1 __init__()	18
7.1.2 Member Data Documentation	18
7.1.2.1 flush	18
7.1.2.2 stream	18
7.1.2.3 write	18
8 File Documentation	19
8.1 cleaner.py File Reference	19
8.2 crawlers/crawler_orchestrator.py File Reference	19
8.3 crawlers/dependencies/__init__.py File Reference	19
8.4 crawlers/dependencies/pubchem.py File Reference	20
8.5 crawlers/dependencies/swissadme.py File Reference	20
8.6 crawlers/dependencies/swisstarget.py File Reference	20
8.7 merger.py File Reference	20
8.8 pipe_handler.py File Reference	20
8.9 populate_chem.py File Reference	20
8.10 README.md File Reference	21
8.11 run_pipeline.py File Reference	21
Index	23

Chapter 1

Usage:

`merger.py` automatically attempts to read `**new_pubchem.csv**` and `**new_swiss.csv**` from `**./data**` and then outputs `**./merger/merge.csv**`

Alternatively, use `merger.py -h` to show arguments that allow passing of specific files on input.
`populate_chem.py` attempts to read data from `**./merger/merge.csv**` then output it to `**./chem/new_attrib.csv**`

Chapter 2

Namespace Index

2.1 Packages

Here are the packages with brief descriptions (if available):

cleaner	11
crawler_orchestrator	13
dependencies	14
dependencies.pubchem	14
dependencies.swissadme	14
dependencies.swisstarget	14
merger	15
pipe_handler	15
populate_chem	15
run_pipeline	15

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

object
 pipe_handler.IgnoreBrokenPipe 17

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

pipe_handler.IgnoreBrokenPipe	17
---	----

Chapter 5

File Index

5.1 File List

Here is a list of all files with brief descriptions:

cleaner.py	19
merger.py	20
pipe_handler.py	20
populate_chem.py	20
run_pipeline.py	21
crawlers/crawler_orchestrator.py	19
crawlers/dependencies/__init__.py	19
crawlers/dependencies/pubchem.py	20
crawlers/dependencies/swissadme.py	20
crawlers/dependencies/swisstarget.py	20

Chapter 6

Namespace Documentation

6.1 cleaner Namespace Reference

Functions

- def `load_data` (csv_file_name)
- def `set_proper_data_type` (df)
- def `unique_count` (column)
- def `remove_useless_columns` (df)
- list `get_duplicity_correlated_descriptors` (graphs, descriptors)
- def `get_correlated_descriptors` (df, threshold, fig_location)
- def `create_correlation_graphs` (correlations, fig_location, threshold)
- def `save_mask` (df, mask_name, [None, list] columns=None, [None, list] rows=None)
- pd.DataFrame `check_correlated_column` (df, threshold=0.9, remove=False, preserve_columns=[], graph_location='./plot/correlations_grapgs_{}.jpeg', heatmap_location='./plot/correlations_heatmap_{}.jpeg')
- def `remove_duplicity` (pd.DataFrame df, subset, keep='first')
- def `check_outliers` (df, threshold=4.2, remove=False)
- def `main` (input_file, output_file, correlation_threshold=0.95, outliers_threshold=4.2, preserve_columns=None, remove=False)
- dict `process_config` (config_file='conf/cleaner.ini')

Variables

- dict `args` = `process_config`('conf/cleaner.ini')

6.1.1 Function Documentation

6.1.1.1 `check_correlated_column()`

```
pd.DataFrame cleaner.check_correlated_column (  
    df,  
    threshold = 0.9,  
    remove = False,  
    preserve_columns = [],  
    graph_location = './plot/correlations_grapgs_{}.jpeg',  
    heatmap_location = './plot/correlations_heatmap_{}.jpeg' )
```

6.1.1.2 check_outliers()

```
def cleaner.check_outliers (
    df,
    threshold = 4.2,
    remove = False )
```

6.1.1.3 create_correlation_graphs()

```
def cleaner.create_correlation_graphs (
    correlations,
    fig_location,
    threshold )
```

6.1.1.4 get_correlated_descriptors()

```
def cleaner.get_correlated_descriptors (
    df,
    threshold,
    fig_location )
```

6.1.1.5 get_duplicity_correlated_descriptors()

```
list cleaner.get_duplicity_correlated_descriptors (
    graphs,
    descriptors )
```

6.1.1.6 load_data()

```
def cleaner.load_data (
    csv_file_name )
```

6.1.1.7 main()

```
def cleaner.main (
    input_file,
    output_file,
    correlation_threshold = 0.95,
    outliers_threshold = 4.2,
    preserve_columns = None,
    remove = False )
```

6.1.1.8 process_config()

```
dict cleaner.process_config (
    config_file = 'conf/cleaner.ini' )
```

Process input config and extract
:param config_file: config file name location
:return: configuration

6.1.1.9 remove_duplicits()

```
def cleaner.remove_duplicits (
    pd.DataFrame df,
    subset,
    keep = 'first' )
```

6.1.1.10 remove_useless_columns()

```
def cleaner.remove_useless_columns (
    df )
```

6.1.1.11 save_mask()

```
def cleaner.save_mask (
    df,
    mask_name,
    [None, list] columns = None,
    [None, list] rows = None )
```

6.1.1.12 set_proper_data_type()

```
def cleaner.set_proper_data_type (
    df )
```

6.1.1.13 unique_count()

```
def cleaner.unique_count (
    column )
```

6.1.2 Variable Documentation

6.1.2.1 args

```
dict cleaner.args = process\_config('conf/cleaner.ini')
```

6.2 crawler_orchestrator Namespace Reference

Functions

- def [main](#) (file)

6.2.1 Function Documentation

6.2.1.1 main()

```
def crawler_orchestrator.main (
    file )
```

6.3 dependencies Namespace Reference

Namespaces

- [pubchem](#)
- [swissadme](#)
- [swisstarget](#)

6.4 dependencies.pubchem Namespace Reference

Functions

- def [crawl](#) (prefs)

6.4.1 Function Documentation

6.4.1.1 [crawl\(\)](#)

```
def dependencies.pubchem.crawl (  
    prefs )
```

6.5 dependencies.swissadme Namespace Reference

Functions

- def [is_file_downloaded](#) (filename, timeout=5)
- def [crawl](#) (prefs)

6.5.1 Function Documentation

6.5.1.1 [crawl\(\)](#)

```
def dependencies.swissadme.crawl (  
    prefs )
```

6.5.1.2 [is_file_downloaded\(\)](#)

```
def dependencies.swissadme.is_file_downloaded (  
    filename,  
    timeout = 5 )
```

6.6 dependencies.swisstarget Namespace Reference

Functions

- def [is_file_downloaded](#) (filename, timeout=5)
- def [crawl](#) (prefs)

6.6.1 Function Documentation

6.6.1.1 crawl()

```
def dependencies.swisstarget.crawl (
    prefs )
```

6.6.1.2 is_file_downloaded()

```
def dependencies.swisstarget.is_file_downloaded (
    filename,
    timeout = 5 )
```

6.7 merger Namespace Reference

Functions

- def [merge](#) ()

6.7.1 Function Documentation

6.7.1.1 merge()

```
def merger.merge ( )
```

6.8 pipe_handler Namespace Reference

Classes

- class [IgnoreBrokenPipe](#)

6.9 populate_chem Namespace Reference

Functions

- def [populate](#) ()

6.9.1 Function Documentation

6.9.1.1 populate()

```
def populate_chem.populate ( )
```

6.10 run_pipeline Namespace Reference

Variables

- [parser](#) = argparse.ArgumentParser()
- [type](#)
- [str](#)
- [help](#)
- [args](#) = parser.parse_args()

6.10.1 Variable Documentation

6.10.1.1 args

```
run_pipeline.args = parser.parse_args()
```

6.10.1.2 help

```
run_pipeline.help
```

6.10.1.3 parser

```
run_pipeline.parser = argparse.ArgumentParser()
```

6.10.1.4 str

```
run_pipeline.str
```

6.10.1.5 type

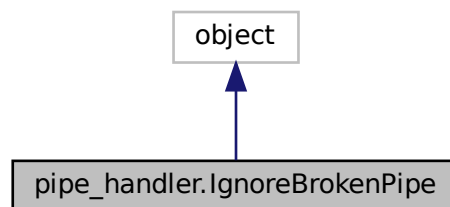
```
run_pipeline.type
```

Chapter 7

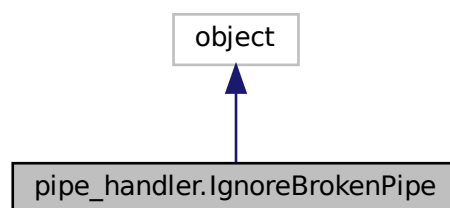
Class Documentation

7.1 pipe_handler.IgnoreBrokenPipe Class Reference

Inheritance diagram for pipe_handler.IgnoreBrokenPipe:



Collaboration diagram for pipe_handler.IgnoreBrokenPipe:



Public Member Functions

- `def __init__(self, stream)`

Public Attributes

- `stream`

- [write](#)
- [flush](#)

7.1.1 Constructor & Destructor Documentation

7.1.1.1 `__init__()`

```
def pipe_handler.IgnoreBrokenPipe.__init__ (
    self,
    stream )
```

7.1.2 Member Data Documentation

7.1.2.1 `flush`

```
pipe_handler.IgnoreBrokenPipe.flush
```

7.1.2.2 `stream`

```
pipe_handler.IgnoreBrokenPipe.stream
```

7.1.2.3 `write`

```
pipe_handler.IgnoreBrokenPipe.write
```

The documentation for this class was generated from the following file:

- [pipe_handler.py](#)

Chapter 8

File Documentation

8.1 cleaner.py File Reference

Namespaces

- [cleaner](#)

Functions

- def [cleaner.load_data](#) (csv_file_name)
- def [cleaner.set_proper_data_type](#) (df)
- def [cleaner.unique_count](#) (column)
- def [cleaner.remove_useless_columns](#) (df)
- list [cleaner.get_duplicat_correlated_descriptors](#) (graphs, descriptors)
- def [cleaner.get_correlated_descriptors](#) (df, threshold, fig_location)
- def [cleaner.create_correlation_graphs](#) (correlations, fig_location, threshold)
- def [cleaner.save_mask](#) (df, mask_name, [None, list] columns=None, [None, list] rows=None)
- pd.DataFrame [cleaner.check_correlated_column](#) (df, threshold=0.9, remove=False, preserve_columns=[], graph_location='./plot/correlations_grapgs_{}.jpeg', heatmap_location='./plot/correlations_heatmap_{}.jpeg')
- def [cleaner.remove_duplicits](#) (pd.DataFrame df, subset, keep='first')
- def [cleaner.check_outliers](#) (df, threshold=4.2, remove=False)
- def [cleaner.main](#) (input_file, output_file, correlation_threshold=0.95, outliers_threshold=4.2, preserve_↵ columns=None, remove=False)
- dict [cleaner.process_config](#) (config_file='conf/cleaner.ini')

Variables

- dict [cleaner.args](#) = process_config('conf/cleaner.ini')

8.2 crawlers/crawler_orchestrator.py File Reference

Namespaces

- [crawler_orchestrator](#)

Functions

- def [crawler_orchestrator.main](#) (file)

8.3 crawlers/dependencies/__init__.py File Reference

Namespaces

- [dependencies](#)

8.4 crawlers/dependencies/pubchem.py File Reference

Namespaces

- [dependencies.pubchem](#)

Functions

- def [dependencies.pubchem.crawl](#) (prefs)

8.5 crawlers/dependencies/swissadme.py File Reference

Namespaces

- [dependencies.swissadme](#)

Functions

- def [dependencies.swissadme.is_file_downloaded](#) (filename, timeout=5)
- def [dependencies.swissadme.crawl](#) (prefs)

8.6 crawlers/dependencies/swisstarget.py File Reference

Namespaces

- [dependencies.swisstarget](#)

Functions

- def [dependencies.swisstarget.is_file_downloaded](#) (filename, timeout=5)
- def [dependencies.swisstarget.crawl](#) (prefs)

8.7 merger.py File Reference

Namespaces

- [merger](#)

Functions

- def [merger.merge](#) ()

8.8 pipe_handler.py File Reference

Classes

- class [pipe_handler.IgnoreBrokenPipe](#)

Namespaces

- [pipe_handler](#)

8.9 populate_chem.py File Reference

Namespaces

- [populate_chem](#)

Functions

- `def populate_chem.populate ()`

8.10 README.md File Reference

8.11 run_pipeline.py File Reference

Namespaces

- `run_pipeline`

Variables

- `run_pipeline.parser = argparse.ArgumentParser()`
- `run_pipeline.type`
- `run_pipeline.str`
- `run_pipeline.help`
- `run_pipeline.args = parser.parse_args()`

Index

- `__init__`
 - `pipe_handler.IgnoreBrokenPipe`, 18
- `args`
 - `cleaner`, 13
 - `run_pipeline`, 16
- `check_correlated_column`
 - `cleaner`, 11
- `check_outliers`
 - `cleaner`, 11
- `cleaner`, 11
 - `args`, 13
 - `check_correlated_column`, 11
 - `check_outliers`, 11
 - `create_correlation_graphs`, 12
 - `get_correlated_descriptors`, 12
 - `get_duplicat_correlated_descriptors`, 12
 - `load_data`, 12
 - `main`, 12
 - `process_config`, 12
 - `remove_duplicits`, 12
 - `remove_useless_columns`, 13
 - `save_mask`, 13
 - `set_proper_data_type`, 13
 - `unique_count`, 13
- `cleaner.py`, 19
- `crawl`
 - `dependencies.pubchem`, 14
 - `dependencies.swissadme`, 14
 - `dependencies.swisstarget`, 14
- `crawler_orchestrator`, 13
 - `main`, 13
- `crawlers/crawler_orchestrator.py`, 19
- `crawlers/dependencies/__init__.py`, 19
- `crawlers/dependencies/pubchem.py`, 20
- `crawlers/dependencies/swissadme.py`, 20
- `crawlers/dependencies/swisstarget.py`, 20
- `create_correlation_graphs`
 - `cleaner`, 12
- `dependencies`, 14
 - `dependencies.pubchem`, 14
 - `crawl`, 14
 - `dependencies.swissadme`, 14
 - `crawl`, 14
 - `is_file_downloaded`, 14
 - `dependencies.swisstarget`, 14
 - `crawl`, 14
 - `is_file_downloaded`, 15
- `flush`
 - `pipe_handler.IgnoreBrokenPipe`, 18
- `get_correlated_descriptors`
 - `cleaner`, 12
- `get_duplicat_correlated_descriptors`
 - `cleaner`, 12
- `help`
 - `run_pipeline`, 16
- `is_file_downloaded`
 - `dependencies.swissadme`, 14
 - `dependencies.swisstarget`, 15
- `load_data`
 - `cleaner`, 12
- `main`
 - `cleaner`, 12
 - `crawler_orchestrator`, 13
- `merge`
 - `merger`, 15
- `merger`, 15
 - `merge`, 15
- `merger.py`, 20
- `parser`
 - `run_pipeline`, 16
- `pipe_handler`, 15
 - `pipe_handler.IgnoreBrokenPipe`, 17
 - `__init__`, 18
 - `flush`, 18
 - `stream`, 18
 - `write`, 18
- `pipe_handler.py`, 20
- `populate`
 - `populate_chem`, 15
- `populate_chem`, 15
 - `populate`, 15
- `populate_chem.py`, 20
- `process_config`
 - `cleaner`, 12
- `README.md`, 21
- `remove_duplicits`
 - `cleaner`, 12
- `remove_useless_columns`
 - `cleaner`, 13
- `run_pipeline`, 15
 - `args`, 16

- help, [16](#)
 - parser, [16](#)
 - str, [16](#)
 - type, [16](#)
- run_pipeline.py, [21](#)
- save_mask
 - cleaner, [13](#)
- set_proper_data_type
 - cleaner, [13](#)
- str
 - run_pipeline, [16](#)
- stream
 - pipe_handler.IgnoreBrokenPipe, [18](#)
- type
 - run_pipeline, [16](#)
- unique_count
 - cleaner, [13](#)
- write
 - pipe_handler.IgnoreBrokenPipe, [18](#)