

---

# **Read the Docs Template Documentation**

***Release 1.0***

**Read the Docs**

**Jan 16, 2023**



# CONTENTS

<b>1</b>	<b>Getting started</b>	<b>3</b>
<b>2</b>	<b>Authors</b>	<b>5</b>
<b>3</b>	<b>Contribution</b>	<b>7</b>
<b>4</b>	<b>Glossary</b>	<b>9</b>
4.1	SWAT . . . . .	9
4.2	HRU . . . . .	9
4.3	PEST . . . . .	9
<b>5</b>	<b>History</b>	<b>11</b>
<b>6</b>	<b>API Reference</b>	<b>13</b>
<b>7</b>	<b>Indices and tables</b>	<b>39</b>
	<b>Index</b>	<b>41</b>



Contents:



## **GETTING STARTED**

SWATy is a Python package to support the Soil & Water Assessment Tool (SWAT) model simulation and calibration.





**AUTHORS**

- Chang Liao



## CONTRIBUTION

Swaty was developed and maintained by

- Chang Liao (Pacific Northwest National Laboratory)



## GLOSSARY

### 4.1 SWAT

Soil & Water Assessment Tool

### 4.2 HRU

Hydrologic Response Units

### 4.3 PEST

Parameter Estimation (PEST), the software package, automates calibration, and calibration-constrained uncertainty analysis of any numerical model.



## HISTORY

- 2017-05-12: Design
- 2020-04-12: Publish





## API REFERENCE

`swaty.classes.pycase.CaseClassEncoder : public JSONEncoder`

The JSON encoder **for** the pycase **class**

**Args:**

`JSONEncoder (_type_):` The json encoder **for** hru **class**

**Public Functions**

`default(self, obj)`

`swaty.classes.hru.HruClassEncoder : public JSONEncoder`

The JSON encoder **for** the hru **class**

**Args:**

`JSONEncoder (_type_):` The json encoder **for** hru **class**

**Public Functions**

`default(self, obj)`

`swaty.classes.swatpara.ParaClassEncoder : public JSONEncoder`

The general parameter JSON encoder

**Args:**

`JSONEncoder (_type_):` `_description_`

## Public Functions

**default**(*self*, *obj*)

**swaty.classes.hru.pyhru** : public object

The HRU **class**

### Args:

**object** (*\_type\_*): *\_description\_*

### Returns:

*\_type\_*: *\_description\_*

## Public Functions

**\_\_init\_\_**(*self*, *aConfig\_in*=None)

**setup\_parameter\_by\_dict**(*self*, *aPara\_in*=None)

Set up the hru **class object** parameter

### Args:

*aPara\_in* (**dict**, optional): The dictionary that stores parameters. Defaults to **None**.

**setup\_parameter\_by\_list**(*self*, *aPara\_in*=None)

Another function to **set** up the hru **class object** parameter

### Args:

*aPara\_in* (**list**, optional): The **list** that stores parameters. Defaults to **None**.

**tojson**(*self*)

Convert a hru **object** to a JSON **object**

### Returns:

*\_type\_*: *\_description\_*

## Public Members

**nParameter\_hru**

## Public Static Attributes

`lIndex_hru = -1`

`iFlag_hru = 0`

`nSoil_layer = 1`

`nParameter_hru = 0`

`aParameter_hru = None`

`aParameter_hru_name = None`

`aSoil = None`

`sSoil_type = ''`

`swaty.classes.soil.pysoil` : public object

The soil **class**

**Args:**

`object (_type_): _description_`

**Returns:**

`_type_: _description_`

## Public Functions

`__init__(self, aConfig_in=None)`

`setup_parameter_by_dict(self, aPara_in=None)`

Set up the soil **class object** parameter

**Args:**

`aPara_in (dict, optional): The dictionary that stores parameters. Defaults to ↪ None.`

`setup_parameter_by_list(self, aPara_in=None)`

Another function to **set** up the soil **class object** parameter

**Args:**

`aPara_in (list, optional): The list that stores parameters. Defaults to ↪ None.`

`tojson(self)`

Convert the `object` to a JSON `object`

Returns:

`_type_`: `_description_`

## Public Members

`nParameter_soil`

## Public Static Attributes

`lIndex_hru = -1`

`lIndex_soil_layer = -1`

`iFlag_soil = 0`

`sSoil_type = ''`

`nParameter_soil = 0`

`aParameter_soil = None`

`aParameter_soil_name = None`

`swaty.classes.subbasin.pysubbasin` : public object

The subbasin `class`

**Args:**

`object (_type_)`: `_description_`

Returns:

`_type_`: `_description_`

## Public Functions

`__init__(self, aConfig_in=None)`

`setup_parameter_by_dict(self, aPara_in=None)`

Set up the subbasin **class object** parameter

Args:

    aPara\_in (**dict**, optional): The dictionary that stores parameters. Defaults to ↵  
↵to **None**.

`setup_parameter_by_list(self, aPara_in=None)`

Another function to **set** up the subbasin **class object** parameter

Args:

    aPara\_in (**list**, optional): The **list** that stores parameters. Defaults to ↵  
↵**None**.

`tojson(self)`

Convert a subbasin **object** to a JSON **object**

Returns:

    \_type\_: \_description\_

## Public Members

`nParameter_subbasin`

## Public Static Attributes

`lIndex_subbasin = -1`

`iFlag_subbasin = 0`

`nSoil_layer = 1`

`nParameter_subbasin = 0`

`aParameter_subbasin = None`

`aParameter_subbasin_name = None`

`swaty.classes.watershed.pywatershed` : public object

The watershed **class**

**Args:**

`object (_type_): _description_`

**Returns:**

`_type_: _description_`

## Public Functions

`__init__(self, aConfig_in=None)`

`setup_parameter_by_dict(self, aPara_in)`

Set up the watershed **class object** parameter

**Args:**

`aPara_in (dict, optional):` The dictionary that stores parameters. Defaults\_  
↪ to **None**.

`tojson(self)`

Convert a watershed **object** to a JSON **object**

**Returns:**

`_type_: _description_`

## Public Members

`nParameter_watershed`

## Public Static Attributes

`lIndex_watershed = -1`

`iFlag_watershed = 0`

`nSoil_layer = 1`

`nParameter_watershed = 0`

`aParameter_watershed = None`

```
aParameter_watershed_name = None
```

```
swaty.classes.soil.SoilClassEncoder : public JSONEncoder
```

The JSON encoder **for** the soil **class**

**Args:**

```
JSONEncoder (_type_): _description_
```

### Public Functions

```
default(self, obj)
```

```
swaty.classes.subbasin.SubbasinClassEncoder : public JSONEncoder
```

The subbasin **class json** encoder

**Args:**

```
JSONEncoder (_type_): _description_
```

### Public Functions

```
default(self, obj)
```

```
swaty.classes.pycase.swatcase : public object
```

The swat case **class**

**Args:**

```
object (_type_): _description_
```

**Returns:**

```
_type_: _description_
```

### Public Functions

```
__init__(self, aConfig_in, iFlag_read_discretization_in=None, iFlag_standalone_in=None,
          sDate_in=None, sWorkspace_output_in=None, aParameter_in=None)
```

```
_summary_
```

```
:param aConfig_in: _description_
```

```
:type aConfig_in: _type_
```

```
:param iFlag_read_discretization_in: _description_, defaults to None
```

```
:type iFlag_read_discretization_in: _type_, optional
```

```
:param iFlag_standalone_in: _description_, defaults to None
```

```
:type iFlag_standalone_in: _type_, optional
```

(continues on next page)

(continued from previous page)

```
:param sDate_in: _description_, defaults to None
:type sDate_in: _type_, optional
:param sWorkspace_output_in: _description_, defaults to None
:type sWorkspace_output_in: _type_, optional
:param aParameter_in: _description_, defaults to None
:type aParameter_in: _type_, optional
```

**copy TxtInOut\_files**(self)

Copy the raw SWAT **input** files

**prepare\_pest\_template\_files**(self)

Prepare **all** the PEST calibration template files

**setup**(self)

Set up a SWAT case

```
convert_pest_parameter_to_model_input(self, sFilename_pest_parameter_watershed_in=None,
                                       sFilename_watershed_parameter_default_in=None,
                                       sFilename_watershed_parameter_bounds_in=None,
                                       sFilename_pest_parameter_subbasin_in=None,
                                       sFilename_subbasin_parameter_default_in=None,
                                       sFilename_subbasin_parameter_bounds_in=None,
                                       sFilename_pest_parameter_hru_in=None,
                                       sFilename_hru_parameter_default_in=None,
                                       sFilename_hru_parameter_bounds_in=None,
                                       sFilename_pest_parameter_soil_in=None,
                                       sFilename_soil_parameter_bounds_in=None,
                                       sWorkspace_soil_parameter_default_in=None)
```

Convert the pest parameters into SWAT **input** file

Args:

```
sFilename_pest_parameter_watershed_in (str, optional): _description_.
↳ Defaults to None.
sFilename_watershed_parameter_default_in (str, optional): _description_.
↳ Defaults to None.
sFilename_watershed_parameter_bounds_in (str, optional): _description_.
↳ Defaults to None.
sFilename_pest_parameter_subbasin_in (str, optional): _description_.
↳ Defaults to None.
sFilename_subbasin_parameter_default_in (str, optional): _description_.
↳ Defaults to None.
sFilename_subbasin_parameter_bounds_in (str, optional): _description_.
↳ Defaults to None.
sFilename_pest_parameter_hru_in (str, optional): _description_. Defaults to
↳ None.
sFilename_hru_parameter_default_in (str, optional): _description_. Defaults
↳ to None.
sFilename_hru_parameter_bounds_in (str, optional): _description_. Defaults
```

(continues on next page)



(continued from previous page)

```

↳to None.
    sFilename_pest_parameter_soil_in (str, optional): _description_. Defaults_
↳to None.
    sFilename_soil_parameter_bounds_in (str, optional): _description_. Defaults_
↳to None.
    sWorkspace_soil_parameter_default_in (str, optional): _description_.
Defaults to None.

```

```

convert_pest_parameter_to_actual_parameter(self, sFilename_pest_parameter_watershed_in=None,
                                           sFilename_watershed_parameter_default_in=None,
                                           sFilename_watershed_parameter_bounds_in=None,
                                           sFilename_pest_parameter_subbasin_in=None,
                                           sFilename_subbasin_parameter_default_in=None,
                                           sFilename_subbasin_parameter_bounds_in=None,
                                           sFilename_pest_parameter_hru_in=None,
                                           sFilename_hru_parameter_default_in=None,
                                           sFilename_hru_parameter_bounds_in=None,
                                           sFilename_pest_parameter_soil_in=None,
                                           sFilename_soil_parameter_bounds_in=None,
                                           sWorkspace_soil_parameter_default_in=None)

```

Convert PEST parameters to actual SWAT parameters

Args:

```

    sFilename_pest_parameter_watershed_in (_type_, optional): _description_.
Defaults to None.
    sFilename_watershed_parameter_default_in (_type_, optional): _description_.
Defaults to None.
    sFilename_watershed_parameter_bounds_in (_type_, optional): _description_.
Defaults to None.
    sFilename_pest_parameter_subbasin_in (_type_, optional): _description_.
Defaults to None.
    sFilename_subbasin_parameter_default_in (_type_, optional): _description_.
Defaults to None.
    sFilename_subbasin_parameter_bounds_in (_type_, optional): _description_.
Defaults to None.
    sFilename_pest_parameter_hru_in (_type_, optional): _description_. Defaults_
↳to None.
    sFilename_hru_parameter_default_in (_type_, optional): _description_.
Defaults to None.
    sFilename_hru_parameter_bounds_in (_type_, optional): _description_.
Defaults to None.
    sFilename_pest_parameter_soil_in (_type_, optional): _description_.
Defaults to None.
    sFilename_soil_parameter_bounds_in (_type_, optional): _description_.
Defaults to None.
    sWorkspace_soil_parameter_default_in (_type_, optional): _description_.
Defaults to None.

```

```

convert_pest_watershed_parameter_to_actual_parameter(self, sFile-
                                                    name_pest_parameter_watershed_in=None,
                                                    sFile-
                                                    name_watershed_parameter_default_in=None,
                                                    sFile-
                                                    name_watershed_parameter_bounds_in=None)

```

Convert PEST watershed parameter to actual parameter

```

:param sFilename_pest_parameter_watershed_in: _description_, defaults to None
:type sFilename_pest_parameter_watershed_in: _type_, optional
:param sFilename_watershed_parameter_bounds_in: _description_, defaults to None
:type sFilename_watershed_parameter_bounds_in: _type_, optional

```

```

convert_pest_subbasin_parameter_to_actual_parameter(self, sFile-
                                                    name_pest_parameter_subbasin_in=None,
                                                    sFile-
                                                    name_subbasin_parameter_default_in=None,
                                                    sFile-
                                                    name_subbasin_parameter_bounds_in=None)

```

Convert PEST subbasin parameter to actual parameter

Args:

```

sFilename_pest_parameter_subbasin_in (_type_, optional): _description_.
↳ Defaults to None.
sFilename_subbasin_parameter_default_in (_type_, optional): _description_.
↳ Defaults to None.
sFilename_subbasin_parameter_bounds_in (_type_, optional): _description_.
↳ Defaults to None.

```

```

convert_pest_hru_parameter_to_actual_parameter(self, sFilename_pest_parameter_hru_in=None,
                                                  sFilename_hru_parameter_default_in=None,
                                                  sFilename_hru_parameter_bounds_in=None)

```

Convert PEST hru parameter to actual parameter

Args:

```

sFilename_pest_parameter_hru_in (_type_, optional): _description_. Defaults
↳ to None.
sFilename_hru_parameter_default_in (_type_, optional): _description_.
↳ Defaults to None.
sFilename_hru_parameter_bounds_in (_type_, optional): _description_.
↳ Defaults to None.

```

```

convert_pest_soil_parameter_to_actual_parameter(self, sFilename_pest_parameter_soil_in=None,
                                                  sFilename_soil_parameter_bounds_in=None,
                                                  sWorkspace_soil_parameter_default_in=None)

```

Convert PEST soil parameter to actual parameter

Args:

```

sFilename_pest_parameter_soil_in (_type_, optional): _description_.

```

(continues on next page)

(continued from previous page)

```

↪ Defaults to None.
    sFilename_soil_parameter_bounds_in (_type_, optional): _description_.
↪ Defaults to None.
    sWorkspace_soil_parameter_default_in (_type_, optional): _description_.
↪ Defaults to None.

```

**run**(*self*)

Run the SWAT simulation using a subprocess

**analyze**(*self*, *sFilename\_output\_in*=None)

Analyze the SWAT simulation

Args:

```

    sFilename_output_in (_type_, optional): _description_. Defaults to None.

```

**evaluate**(*self*)

Evaluate the SWAT model simulation

**swaty\_generate\_model\_structure\_files**(*self*)

Generate the SWAT model spatial discretization configuration

**generate\_parameter\_bounds**(*self*, *sFilename\_watershed\_parameter\_bounds\_in*=None,  
*sFilename\_subbasin\_parameter\_bounds\_in*=None,  
*sFilename\_hru\_parameter\_bounds\_in*=None,  
*sFilename\_soil\_parameter\_bounds\_in*=None)

Generate the upper **and** lower bound of SWAT parameters

Args:

```

    sFilename_watershed_parameter_bounds_in (str, optional): _description_.
↪ Defaults to None.
    sFilename_subbasin_parameter_bounds_in (str, optional): _description_.
↪ Defaults to None.
    sFilename_hru_parameter_bounds_in (str, optional): _description_. Defaults
↪ to None.
    sFilename_soil_parameter_bounds_in (str, optional): _description_. Defaults
↪ to None.

```

**extract\_default\_parameter\_value**(*self*, *aParameter\_in*, *sFilename\_watershed\_in*=None,  
*sFilename\_subbasin\_in*=None, *sFilename\_hru\_in*=None,  
*sWorkspace\_soil\_in*=None)

Extract the default SWAT model parameters

Args:

```

    aParameter_in (_type_): _description_
    sFilename_watershed_in (str, optional): _description_. Defaults to None.
    sFilename_subbasin_in (str, optional): _description_. Defaults to None.

```

(continues on next page)

(continued from previous page)

```
sFilename_hru_in (str, optional): _description_. Defaults to None.  
sWorkspace_soil_in (str, optional): _description_. Defaults to None.
```

```
extract_default_parameter_value_watershed(self, aParameter_watershed,  
                                           sFilename_watershed_in=None)
```

```
Extract the default watershed parameter
```

```
Args:
```

```
aParameter_watershed (_type_): _description_  
sFilename_watershed_in (str, optional): _description_. Defaults to None.
```

```
extract_default_parameter_value_subbasin(self, aParameter_subbasin,  
                                           sFilename_subbasin_in=None)
```

```
Extract the default subbaisn parameter
```

```
Args:
```

```
aParameter_subbasin (_type_): _description_  
sFilename_subbasin_in (_type_, optional): _description_. Defaults to None.
```

```
extract_default_parameter_value_hru(self, aParameter_hru, sFilename_hru_in=None)
```

```
Extract the default hru parameters
```

```
Args:
```

```
aParameter_hru (_type_): _description_  
sFilename_hru_in (str, optional): _description_. Defaults to None.
```

```
extract_default_parameter_value_soil(self, aParameter_soil, sWorkspace_soil_in=None)
```

```
Extract the default soil parameter
```

```
Args:
```

```
aParameter_soil (_type_): _description_  
sWorkspace_soil_in (str, optional): _description_. Defaults to None.
```

```
swaty_prepare_watershed_configuration(self)
```

```
Prepare the watershed configuration
```

```
swaty_retrieve_soil_info(self)
```

```
Retrieve the soil information from the existing files
```

```
swaty_prepare_watershed_parameter_file(self)
```

```
prepare the pest control file
```

```
swaty_prepare_watershed_template_file(self, sFilename_watershed_template_in=None)
```

```
#prepare the pest control file
```

**swaty\_prepare\_subbasin\_parameter\_file**(*self*)

```
#prepare the pest control file
```

**swaty\_prepare\_subbasin\_template\_file**(*self*, *sFilename\_subbasin\_template\_in*=None)

```
#prepare the pest control file
```

**swaty\_prepare\_hru\_parameter\_file**(*self*)

```
#prepare the pest control file
```

**swaty\_prepare\_hru\_template\_file**(*self*, *sFilename\_hru\_template\_in*=None)

Prepare the hru template file **for** PEST

Args:

*sFilename\_hru\_template\_in* (**str**, optional): *\_description\_*. Defaults to **None**.

**swaty\_prepare\_soil\_parameter\_file**(*self*)

```
Prepare the soil parameter file
```

**swaty\_prepare\_soil\_template\_file**(*self*, *sFilename\_soil\_template\_in*=None)

Prepare the soil template file **for** PEST

Args:

*sFilename\_soil\_template\_in* (**str**, optional): *\_description\_*. Defaults to **None**.

**swaty\_create\_pest\_instruction\_file**(*self*, *sFilename\_instruction*)

Prepare pest instruction file

Args:

*sFilename\_instruction* (**str**): The pest instruction filename

**swaty\_write\_watershed\_input\_file**(*self*)

```
write the input files from the new parameter generated by PEST to each hru file
```

**swaty\_write\_subbasin\_input\_file**(*self*)

```
write the input files from the new parameter generated by PEST to each hru file
```

**swaty\_write\_hru\_input\_file**(*self*)

```
write the input files from the new parameter generated by PEST to each hru file
```

**swaty\_copy\_executable\_file**(*self*)

Prepare executable file to the workspace

**swaty\_prepare\_simulation\_bash\_file**(*self*)

Generate a swat simulation bash file.

Returns:

    \_type\_: \_description\_

**swaty\_prepare\_simulation\_job\_file**(*self*)

Generate a HPC job file **for** the SWAT simulation

Returns:

    \_type\_: \_description\_

**swaty\_prepare\_observation\_discharge\_file**(*self*)

Pre-process the observed stream discharge

**swaty\_extract\_stream\_discharge**(*self*, *sFilename\_output\_in=None*)

Extract discharge **from swat** model simulation

Args:

*sFilename\_output\_in* (**str**, optional): the destination filename. Defaults to **None**.

**swaty\_tsplot\_stream\_discharge**(*self*)

Plot the time series swat simulated stream discharge

**export\_config\_to\_json**(*self*, *sFilename\_output*)

Export the configuration to a JSON **object**

Returns:

    \_type\_: \_description\_

**tojson**(*self*)

Convert a swat case **object** to a JSON **object**

Returns:

    \_type\_: \_description\_

## Public Members

iFlag\_run

iFlag\_standalone

iFlag\_read\_discretization

iFlag\_initialization

iFlag\_calibration

iFlag\_simulation

iFlag\_watershed

iFlag\_subbasin

iFlag\_hru

iFlag\_soil

iFlag\_mode

iFlag\_replace\_parameter

iYear\_start

iYear\_end

iMonth\_start

iMonth\_end

iDay\_start

iDay\_end

nstress

sRegion

sModel

sPython

sFilename\_model\_configuration

sWorkspace\_input

sWorkspace\_output

sWorkspace\_bin

sDate

iCase\_index

sCase

sJob

sWorkspace\_simulation\_copy

sFilename\_LandUseSoilsReport

sFilename\_HRULandUseSoilsReport

sFilename\_parameter\_bounds

sFilename\_hru\_combination

sFilename\_watershed\_configuration

sFilename\_hru\_info

sFilename\_soil\_combination

sFilename\_soil\_info

nsubbasin

nsegment



nhru\_combination

nhru

nsoil\_combination

sFilename\_observation\_discharge

sTime\_step\_calibration

sFilename\_swat

nstress\_month

nParameter\_watershed

nParameter\_subbasin

nParameter\_hru

nParameter\_soil

sFilename\_swat\_current

### Public Static Attributes

iCase\_index = 0

iSiteID = 0

iFlag\_run = 0

iFlag\_standalone = 1

iFlag\_simulation = 1

iFlag\_initialization = 1

iFlag\_calibration = 0

iFlag\_watershed = 0

```
iFlag_subbasin = 0

iFlag_hru = 0

iFlag_soil = 0

iFlag_mode = 0

iYear_start = 0

iYear_end = 0

iMonth_start = 0

iMonth_end = 0

iDay_start = 0

iDay_end = 0

nstress = 0

nsegment = 0

nhru = 0

nhru_combination = 0

nsoil_combination = 0

aConfig_in = None

aParameter_watershed_name = None

aParameter_subbasin_name = None

aParameter_hru_name = None

aParameter_soil_name = None

pWatershed = None
```

```
aSubbasin = None

aHru = None

aHru_combination = None

aSoil_combinaiton = None

nParameter = 0

nParameter_watershed = 0

nParameter_subbasin = 0

nParameter_hru = 0

nParameter_soil = 0

sFilename_swat_current = ''

sFilename_model_configuration = ''

sWorkspace_input = ''

sWorkspace_output = ''

sTime_step_calibration = ''

sFilename_observation_discharge = ''

sFilename_LandUseSoilsReport = ''

sFilename_HRULandUseSoilsReport = ''

sRegion = ''

sModel = ''

sCase = ''

sDate = ''
```

```
sSiteID = ''
```

```
sDate_start = ''
```

```
sDate_end = ''
```

```
swaty.classes.swatpara.swatpara : public object
```

The parameter **class**

**Args:**

```
object (_type_): _description_
```

**Returns:**

```
_type_: _description_
```

## Public Functions

```
__init__(self, aConfig_in)
```

Initialize a parameter **object** through a dictionary

**Args:**

```
aConfig_in (dict): The dictionary that stores parameters
```

```
tojson(self)
```

Convert a parameter **object** to a JSON **object**

**Returns:**

```
_type_: _description_
```

## Public Members

```
iParameter_type
```

```
iFlag_pseudo
```

```
lIndex_subbasin
```

```
lIndex_hru
```

```
lIndex_soil_layer
```

```
sName
```

dValue\_init

dValue\_current

dValue\_lower

dValue\_upper

### Public Static Attributes

sName = ''

iParameter\_type = 1

lIndex\_subbasin = -1

lIndex\_hru = -1

lIndex\_soil\_layer = -1

iFlag\_pseudo = 0

dValue\_init = 0.0

dValue\_current = 0.5

dValue\_lower = -1

dValue\_upper = 1

swaty.classes.watershed.WatershedClassEncoder : public JSONEncoder

The watershed `class json` encoder

Args:

JSONEncoder (\_type\_): \_description\_

## Public Functions

`default(self, obj)`

`module define_global_variables`

`module swaty`

`module auxiliary`

`module line_count`

## Functions

`line_count(sFilename_in)`

Count the line number of a text-based file

Args:

`sFilename_in (string):` text filename

Returns:

`int:` line number

`module text_reader_string`

## Functions

`text_reader_string(sFilename_in, ncolumn_in=None, nrow_in=None, cDelimiter_in=None, iFlag_remove_quota=None, iSkipline_in=None)`

Read a text based file

`sFilename_in,`

`ncolumn_in = None,`

`nrow_in = None,`

`cDelimiter_in = None,`

`iSkipline_in = None`

`module classes`

`module hru`

`module pycase`

## Variables

```
pDate = datetime.datetime.today()
```

```
sDate_default = "{:04d}".format(pDate.year) + "{:02d}".format(pDate.month) +  
"{:02d}".format(pDate.day)
```

```
module soil
```

```
module subbasin
```

```
module swatpara
```

```
module watershed
```

```
module swaty_create_template_configuration_file
```

## Functions

```
swaty_create_template_configuration_file(sFilename_json, sPath_bin, sWorkspace_input,  
                                         sWorkspace_output, iFlag_standalone_in=None,  
                                         iCase_index_in=None,  
                                         iFlag_read_discretization_in=None, sDate_in=None,  
                                         aParameter_in=None)
```

Generat a template configuration file **for** users

### Args:

sFilename\_json (string): The output json file  
sPath\_bin (string): The swat binary file path  
sWorkspace\_input (string): The **input** workspace  
sWorkspace\_output (string): The output workspace  
iCase\_index\_in (**int**, optional): The case study ID. Defaults to **None**.  
aParameter\_in (**dict**, optional): The dictionary that stores **all** the **parameters**. Defaults to **None**.

### Returns:

**int: None**

```
module swaty_read_model_configuration_file
```

## Functions

```
swaty_read_model_configuration_file(sFilename_configuration_in,  
                                   iFlag_read_discretization_in=None,  
                                   iFlag_standalone_in=None, iCase_index_in=None,  
                                   sDate_in=None, iYear_start_in=None, iMonth_start_in=None,  
                                   iDay_start_in=None, iYear_end_in=None,  
                                   iMonth_end_in=None, iDay_end_in=None,  
                                   sWorkspace_input_in=None, sWorkspace_output_in=None,  
                                   aParameter_in=None)
```

`_summary_`

### Args:

```
sFilename_configuration_in (str): Filename of the configuration json file  
iFlag_read_discretization_in (int, optional): _description_. Defaults to None.  
iFlag_standalone_in (int, optional): _description_. Defaults to None.  
iCase_index_in (int, optional): _description_. Defaults to None.  
sDate_in (str, optional): _description_. Defaults to None.  
iYear_start_in (int, optional): _description_. Defaults to None.  
iMonth_start_in (int, optional): _description_. Defaults to None.  
iDay_start_in (int, optional): _description_. Defaults to None.  
iYear_end_in (int, optional): _description_. Defaults to None.  
iMonth_end_in (int, optional): _description_. Defaults to None.  
iDay_end_in (int, optional): _description_. Defaults to None.  
sWorkspace_input_in (str, optional): _description_. Defaults to None.  
sWorkspace_output_in (str, optional): _description_. Defaults to None.  
aParameter_in (dict, optional): _description_. Defaults to None.
```

### Returns:

```
_type_: _description_
```

## Variables

```
pDate = datetime.datetime.today()
```

```
sDate_default = "{:04d}".format(pDate.year) + "{:02d}".format(pDate.month) +  
"{:02d}".format(pDate.day)
```

```
module tools
```

```
file __init__.py
```

```
file __init__.py
```

```
file __init__.py
```



*file* **\_\_init\_\_.py**

*file* **line\_count.py**

*file* **text\_reader\_string.py**

*file* **hru.py**

*file* **pycase.py**

*file* **soil.py**

*file* **subbasin.py**

*file* **swatpara.py**

*file* **watershed.py**

*file* **swaty\_create\_template\_configuration\_file.py**

*file* **swaty\_read\_model\_configuration\_file.py**

*dir*

**/home/docs/checkouts/readthedocs.org/user\_builds/swaty/checkouts/latest/swaty/auxiliary**

*dir* **/home/docs/checkouts/readthedocs.org/user\_builds/swaty/checkouts/latest/swaty/classes**

*dir* **/home/docs/checkouts/readthedocs.org/user\_builds/swaty/checkouts/latest/swaty**

*dir* **/home/docs/checkouts/readthedocs.org/user\_builds/swaty/checkouts/latest/swaty/tools**



## INDICES AND TABLES

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



## Symbols

`__init__()`  
built-in function, 14, 15, 17–19, 32

## A

`aConfig_in`, 30  
`aHru`, 31  
`aHru_combination`, 31  
`analyze()`  
built-in function, 23  
`aParameter_hru`, 15  
`aParameter_hru_name`, 15, 30  
`aParameter_soil`, 16  
`aParameter_soil_name`, 16, 30  
`aParameter_subbasin`, 17  
`aParameter_subbasin_name`, 17, 30  
`aParameter_watershed`, 18  
`aParameter_watershed_name`, 18, 30  
`aSoil`, 15  
`aSoil_combinaiton`, 31  
`aSubbasin`, 30

## B

built-in function  
`__init__()`, 14, 15, 17–19, 32  
`analyze()`, 23  
`convert_pest_hru_parameter_to_actual_parameter()`, 22  
`convert_pest_parameter_to_actual_parameter()`, 21  
`convert_pest_parameter_to_model_input()`, 20  
`convert_pest_soil_parameter_to_actual_parameter()`, 22  
`convert_pest_subbasin_parameter_to_actual_parameter()`, 22  
`convert_pest_watershed_parameter_to_actual_parameter()`, 21  
`copy_TxtInOut_files()`, 20  
`default()`, 13, 14, 19, 34  
`evaluate()`, 23  
`export_config_to_json()`, 26

`extract_default_parameter_value()`, 23  
`extract_default_parameter_value_hru()`, 24  
`extract_default_parameter_value_soil()`, 24  
`extract_default_parameter_value_subbasin()`, 24  
`extract_default_parameter_value_watershed()`, 24  
`generate_parameter_bounds()`, 23  
`prepare_pest_template_files()`, 20  
`run()`, 23  
`setup()`, 20  
`setup_parameter_by_dict()`, 14, 15, 17, 18  
`setup_parameter_by_list()`, 14, 15, 17  
`swaty.auxiliary.line_count.line_count()`, 34  
`swaty.auxiliary.text_reader_string.text_reader_string()`, 34  
`swaty.swaty_create_template_configuration_file.swaty_c`, 35  
`swaty.swaty_read_model_configuration_file.swaty_read_m`, 36  
`swaty_copy_executable_file()`, 25  
`swaty_create_pest_instruction_file()`, 25  
`swaty_extract_stream_discharge()`, 26  
`swaty_generate_model_structure_files()`, 23  
`swaty_prepare_hru_parameter_file()`, 25  
`swaty_prepare_hru_template_file()`, 25  
`swaty_prepare_observation_discharge_file()`, 26  
`swaty_prepare_simulation_bash_file()`, 26  
`swaty_prepare_simulation_job_file()`, 26  
`swaty_prepare_soil_parameter_file()`, 25  
`swaty_prepare_soil_template_file()`, 25  
`swaty_prepare_subbasin_parameter_file()`, 25  
`swaty_prepare_subbasin_template_file()`, 25  
`swaty_prepare_watershed_configuration()`, 24  
`swaty_prepare_watershed_parameter_file()`, 24

24  
swaty\_prepare\_watershed\_template\_file(),  
24  
swaty\_retrieve\_soil\_info(), 24  
swaty\_tsplot\_stream\_discharge(), 26  
swaty\_write\_hru\_input\_file(), 25  
swaty\_write\_subbasin\_input\_file(), 25  
swaty\_write\_watershed\_input\_file(), 25  
tojson(), 14, 15, 17, 18, 26, 32

## C

convert\_pest\_hru\_parameter\_to\_actual\_parameter()  
built-in function, 22  
convert\_pest\_parameter\_to\_actual\_parameter()  
built-in function, 21  
convert\_pest\_parameter\_to\_model\_input()  
built-in function, 20  
convert\_pest\_soil\_parameter\_to\_actual\_parameter()  
built-in function, 22  
convert\_pest\_subbasin\_parameter\_to\_actual\_parameter()  
built-in function, 22  
convert\_pest\_watershed\_parameter\_to\_actual\_parameter()  
built-in function, 21  
copy\_TxtInOut\_files()  
built-in function, 20

## D

default()  
built-in function, 13, 14, 19, 34  
dValue\_current, 33  
dValue\_init, 32, 33  
dValue\_lower, 33  
dValue\_upper, 33

## E

evaluate()  
built-in function, 23  
export\_config\_to\_json()  
built-in function, 26  
extract\_default\_parameter\_value()  
built-in function, 23  
extract\_default\_parameter\_value\_hru()  
built-in function, 24  
extract\_default\_parameter\_value\_soil()  
built-in function, 24  
extract\_default\_parameter\_value\_subbasin()  
built-in function, 24  
extract\_default\_parameter\_value\_watershed()  
built-in function, 24

## G

generate\_parameter\_bounds()  
built-in function, 23

## I

iCase\_index, 28, 29  
iDay\_end, 27, 30  
iDay\_start, 27, 30  
iFlag\_calibration, 27, 29  
iFlag\_hru, 15, 27, 30  
iFlag\_initialization, 27, 29  
iFlag\_mode, 27, 30  
iFlag\_pseudo, 32, 33  
iFlag\_read\_discretization, 27  
iFlag\_replace\_parameter, 27  
iFlag\_run, 27, 29  
iFlag\_simulation, 27, 29  
iFlag\_soil, 16, 27, 30  
iFlag\_standalone, 27, 29  
iFlag\_subbasin, 17, 27, 29  
iFlag\_watershed, 18, 27, 29  
iMonth\_end, 27, 30  
iMonth\_start, 27, 30  
iParameter\_type, 32, 33  
iSiteID, 29  
iTime\_end, 27, 30  
iYear\_start, 27, 30

## L

lIndex\_hru, 15, 16, 32, 33  
lIndex\_soil\_layer, 16, 32, 33  
lIndex\_subbasin, 17, 32, 33  
lIndex\_watershed, 18

## N

nhru, 29, 30  
nhru\_combination, 28, 30  
nParameter, 31  
nParameter\_hru, 14, 15, 29, 31  
nParameter\_soil, 16, 29, 31  
nParameter\_subbasin, 17, 29, 31  
nParameter\_watershed, 18, 29, 31  
nsegment, 28, 30  
nsoil\_combination, 29, 30  
nSoil\_layer, 15, 17, 18  
nstress, 27, 30  
nstress\_month, 29  
nsubbasin, 28

## P

pDate (*swaty.classes.pycase attribute*), 35  
pDate (*swaty.swaty\_read\_model\_configuration\_file attribute*), 36  
prepare\_pest\_template\_files()  
built-in function, 20  
pWatershed, 30  
pyearth.system.define\_global\_variables (*built-in class*), 34

## R

run()

built-in function, 23

## S

sCase, 28, 31

sDate, 28, 31

sDate\_default (swaty.classes.pycase attribute), 35

sDate\_default (swaty.swaty\_read\_model\_configuration\_file attribute), 36

sDate\_end, 32

sDate\_start, 32

setup()

built-in function, 20

setup\_parameter\_by\_dict()

built-in function, 14, 15, 17, 18

setup\_parameter\_by\_list()

built-in function, 14, 15, 17

sFilename\_hru\_combination, 28

sFilename\_hru\_info, 28

sFilename\_HRULandUseSoilsReport, 28, 31

sFilename\_LandUseSoilsReport, 28, 31

sFilename\_model\_configuration, 28, 31

sFilename\_observation\_discharge, 29, 31

sFilename\_parameter\_bounds, 28

sFilename\_soil\_combination, 28

sFilename\_soil\_info, 28

sFilename\_swat, 29

sFilename\_swat\_current, 29, 31

sFilename\_watershed\_configuration, 28

sJob, 28

sModel, 27, 31

sName, 32, 33

sPython, 28

sRegion, 27, 31

sSiteID, 31

sSoil\_type, 15, 16

sTime\_step\_calibration, 29, 31

swaty (built-in class), 34

swaty.auxiliary (built-in class), 34

swaty.auxiliary.line\_count (built-in class), 34

swaty.auxiliary.line\_count.line\_count()

built-in function, 34

swaty.auxiliary.text\_reader\_string (built-in class), 34

swaty.auxiliary.text\_reader\_string.text\_reader\_string()

built-in function, 34

swaty.classes (built-in class), 34

swaty.classes.hru (built-in class), 34

swaty.classes.pycase (built-in class), 34

swaty.classes.soil (built-in class), 35

swaty.classes.subbasin (built-in class), 35

swaty.classes.swatpara (built-in class), 35

swaty.classes.watershed (built-in class), 35

swaty.swaty\_create\_template\_configuration\_file (built-in class), 35

swaty.swaty\_create\_template\_configuration\_file.swaty\_create\_template\_configuration\_file()

built-in function, 35

swaty.swaty\_read\_model\_configuration\_file

(built-in class), 35

swaty.swaty\_read\_model\_configuration\_file.swaty\_read\_model\_configuration\_file()

built-in function, 36

swaty.tools (built-in class), 36

swaty\_copy\_executable\_file()

built-in function, 25

swaty\_create\_pest\_instruction\_file()

built-in function, 25

swaty\_extract\_stream\_discharge()

built-in function, 26

swaty\_generate\_model\_structure\_files()

built-in function, 23

swaty\_prepare\_hru\_parameter\_file()

built-in function, 25

swaty\_prepare\_hru\_template\_file()

built-in function, 25

swaty\_prepare\_observation\_discharge\_file()

built-in function, 26

swaty\_prepare\_simulation\_bash\_file()

built-in function, 26

swaty\_prepare\_simulation\_job\_file()

built-in function, 26

swaty\_prepare\_soil\_parameter\_file()

built-in function, 25

swaty\_prepare\_soil\_template\_file()

built-in function, 25

swaty\_prepare\_subbasin\_parameter\_file()

built-in function, 25

swaty\_prepare\_subbasin\_template\_file()

built-in function, 25

swaty\_prepare\_watershed\_configuration()

built-in function, 24

swaty\_prepare\_watershed\_parameter\_file()

built-in function, 24

swaty\_prepare\_watershed\_template\_file()

built-in function, 24

swaty\_retrieve\_soil\_info()

built-in function, 24

swaty\_tsplot\_stream\_discharge()

built-in function, 26

swaty\_write\_hru\_input\_file()

built-in function, 25

swaty\_write\_subbasin\_input\_file()

built-in function, 25

swaty\_write\_watershed\_input\_file()

built-in function, 25

sWorkspace\_bin, 28

sWorkspace\_input, 28, 31

sWorkspace\_output, 28, 31

`sWorkspace_simulation_copy`, 28

## T

`tojson()`

built-in function, 14, 15, 17, 18, 26, 32