
fishbase Documentation

Release 1.0.12

David Yi

Oct 26, 2018

Contents

1	3
2 API	7
Python Module Index	17

fishbase Python Python

2016 Python fishbase

fishbase Python

1.1 2018.5.30 v1.0.13

- #9037, common and system, function `check_platform()` move to `fish_system` rename to `fish_system.get_platform()`;
- 19038, common, add function `fish_common.get_uuid()`, edit `fish_common.get_time_uuid`, add doc and unittest;
- 19039, logger, edit function `fish_logger.set_log_file()` by class `SafeFileHandler()`, prevent the multi process delete log file error;
- 19040, file, edit function `fish_file.get_abs_filename_with_sub_path()`, thanks to Wu Yanan;
- 19041, file, delete function `check_kind_path_file()`;
- 19042, file, edit function `fish_file.check_sub_path_create()`, optimize, doc and unittest;
- 19043, common, edit function `fish_common.sorted_list_from_dict()`, optimize, doc and unittest;
- 19044, file, remove `auto_add_file_ext()`;
- 19045, file, remove `get_abs_filename_with_sub_path_module()`;

1.2 2018.5.21 v1.0.12

- 19035, rename package 'fish_base' to 'fishbase'

1.3 2018.5.18 v1.0.11

- 19011, 19011ok
- 19015, common `conf_as_dict()` docstring ok
- 19016, sphinx api ok

- 19017, `conf_as_dict()` doc ok
- 19018, `__init__.py` `get_ver()` ok
- 19019, common class `SingleTon()` docstring ok
- 19020, csv `csv_file_to_list()` docstring ok
- 19021, common `fish_common`, csv `fish_csv`, `fish` ok
- 19022, sphinx doc theme `rtd theme`<https://sphinx-rtd-theme.readthedocs.io/en/latest/>, ok
- 19023, logger `set_log_file()` docstring ok
- 19024, `fish_file` docstring ok
- 19025, common, `get_md5()` ok
- 19026, common, class `GetMD5` `md5` ok
- 19027, test, unittest common ok
- 19028, common, `conf_as_dict()` ok
- 19029, common, `json_contained()` json ok
- 19030, common, `splice_url_params()` ok
- 19031, `requirements.txt`; ok
- 19032, `.travis.yml`; ok
- 19033, `coveralls.io` ut ; python 2.7.15 ; ok
- 19034, `__init__.py` `setup.py` `__version__` ; ok

1.4 2018.3.20 v1.0.10

- 19006, `get_time_uuid()`, ok
- 19007, `if_any_elements_is_space()`, None ok
- 19008, common `conf_as_dict()`; ok
- 11001,
- 11002, csv `csv_file_to_list()`; ok
- 11003, `fish_file` file `fish_file`; ok
- 11004, file `get_abs_filename_with_sub_path()` ok
- 11005, `fish_date` date, `demo/demo_date.py` ok
- 11006, `setup.py` `setup.py` ok
- 11007, pip `python-dateutil`; ok
- 11008, `check_platform()` system
- 11009, csv, `csv_file_to_list()` ok
- 11010, logger, `log`; ok
- 11011, demo, test test log demo ; ok
- 11013, demo, `common.conf_as_dict()` demo ok

- 11014, common, `conf_as_dict()` ok
- 11015, common class `SingleTonok`

2.1 fish_common

<i>fish_common.conf_as_dict(conf_filename)</i>	ini
<i>fish_common.get_uuid(kind)</i>	uuid uuid Python uuid
<i>fish_common.GetMD5.string(s)</i>	MD5
<i>fish_common.GetMD5.file(filename)</i>	MD5
<i>fish_common.GetMD5.big_file(filename)</i>	MD5
<i>fish_common.if_json_contain(left_json,...)</i>	json json key value
<i>fish_common.Singleton</i>	
<i>fish_common.sorted_list_from_dict(p_dict[, value ...])</i>	
<i>fish_common.splice_url_params(dic)</i>	url ? ?key1=value1&key2=value2

fish_common

class fish_common.**GetMD5**

Python hashlib.md5()

:

```
print('--- md5 demo ---')
print('string md5:', GetMD5.string('hello world!'))
print('file md5:', GetMD5.file(get_abs_filename_with_sub_path('test_conf', 'test_
↪conf.ini')[1]))
print('big file md5:', GetMD5.big_file(get_abs_filename_with_sub_path('test_conf',
↪ 'test_conf.ini')[1]))
print('---')
```

:

```
string md5: fc3ff98e8c6a0d3087d515c0473f8677
```

(continues on next page)

(continued from previous page)

```
file md5: fb7528c9778b2377e30b0f7e4c26fef0
big file md5: fb7528c9778b2377e30b0f7e4c26fef0
```

static big_file (*filename*)
MD5

Param

- (string) filename hash

Returns

- (string) result 32 MD5

static file (*filename*)
MD5

Param

- (string) filename hash

Returns

- (string) result 32 MD5

static string (*s*)
MD5

Param

- (string) str hash

Returns

- (string) result 32 MD5

class fish_common.Singleton

Param**Returns**

:

```
print('--- class singleton demo ---')
t1 = Singleton()
t1.x = 2
print('t1.x:', t1.x)

t2 = Singleton()

t1.x += 1

print('t1.x:', t1.x)
print('t2.x:', t2.x)
print('---')
```

:

```
--- class singleton demo ---
t1.x: 2
t1.x: 3
t2.x: 3
---
```

`fish_common.conf_as_dict(conf_filename)`
 ini

Param

- `conf_filename`: (string) ini

Returns

- `flag`: (bool) True False
- `d`: (dict)
- `count`: (int) key

:

```
print('--- conf_as_dict demo---')
#
conf_filename = 'test_conf.ini'
#
ds = conf_as_dict(conf_filename)
# dict dict key
print('flag:', ds[0])
print('dict:', ds[1])
print('length:', ds[2])

d = ds[1]

# section
print('section show_opt:', d['show_opt'])
# section key value
print('section show_opt, key short_opt:', d['show_opt']['short_opt'])

# section key count key value
i = int(d['get_extra_rules']['erule_count'])
print('section get_extra_rules, key erule_count:', i)
for j in range(i):
    print('section get_extra_rules, key erule_type:', d['get_extra_rules']['erule_
↪'+str(j)])
print('---')
```

:

```
--- conf_as_dict demo---
flag: True
dict: (omit)
length: 7
section show_opt: {'short_opt': 'b:d:v:p:f:', 'long_opt': 'region=,prov=,mer_id=,
↪mer_short_name=,web_status='}
section show_opt, key short_opt: b:d:v:p:f:
section get_extra_rules, key erule_count: 2
section get_extra_rules, key erule_type: extra_rule_1
section get_extra_rules, key erule_type: extra_rule_2
---
```

`fish_common.get_time_uuid()`
 uuid uuid Python uuid
 get_time_uuid() uuid v1.0.12

Param

- kind: (int) uuid udTime udRandom

Returns

- result: (string) 66b438e3-200d-4fe3-8c9e-2bc431bb3000 uuid

:

```
print('--- uuid demo ---')
# uuid
for i in range(2):
    print(get_uuid(udTime))

print('---')

# uuid
for i in range(2):
    print(get_time_uuid())

print('---')

# uuid
for i in range(2):
    print(get_uuid(udRandom))

print('---')
```

:

```
--- uuid demo ---
c8aa92cc-60ef-11e8-aa87-acbf52d15413
c8ab7194-60ef-11e8-b7bd-acbf52d15413
---
c8ab7368-60ef-11e8-996c-acbf52d15413
c8ab741e-60ef-11e8-959d-acbf52d15413
---
8e108777-26a1-42d6-9c4c-a0c029423eb0
8175a81a-f346-46af-9659-077ad52e3e8f
---
```

fish_common.get_uuid(kind)

uuid uuid Python uuid

get_time_uuid() uuid v1.0.12

Param

- kind: (int) uuid udTime udRandom

Returns

- result: (string) 66b438e3-200d-4fe3-8c9e-2bc431bb3000 uuid

:

```
print('--- uuid demo ---')
# uuid
for i in range(2):
    print(get_uuid(udTime))

print('---')
```

(continues on next page)

(continued from previous page)

```
# uuid
for i in range(2):
    print(get_time_uuid())

print('---')

# uuid
for i in range(2):
    print(get_uuid(udRandom))

print('---')
```

:

```
--- uuid demo ---
c8aa92cc-60ef-11e8-aa87-acbf52d15413
c8ab7194-60ef-11e8-b7bd-acbf52d15413
---
c8ab7368-60ef-11e8-996c-acbf52d15413
c8ab741e-60ef-11e8-959d-acbf52d15413
---
8e108777-26a1-42d6-9c4c-a0c029423eb0
8175a81a-f346-46af-9659-077ad52e3e8f
---
```

`fish_common.if_json_contain(left_json, right_json, op='strict')`
 json json key value

Param

- left_json: (dict) json left
- right_json: (dict) json right left right
- op: (string) strict

Returns

- result: (bool) right json left json key value True False

:

```
print('--- json contain demo ---')
json1 = {"id": "0001"}
json2 = {"id": "0001", "value": "File"}
print(if_json_contain(json1, json2))
print('---')
```

:

```
True
```

`fish_common.sorted_list_from_dict(p_dict, order=10011)`
 value

Param

- p_dict: (dict)
- order: (int) odASC odDES

Returns

- o_list: (list) list

:

```
#
dict1 = {'a_key': 'a_value', 'l_key': 'l_value', 'A_key': 'A_value', 'z_key': 'z_
↪value'}
print(dict1)
#
list1 = sorted_list_from_dict(dict1, odASC)
print('ascending order result is:', list1)
#
list1 = sorted_list_from_dict(dict1, odDES)
print('descending order result is:', list1)
```

:

```
{'a_key': 'a_value', 'A_key': 'A_value', 'l_key': 'l_value', 'z_key': 'z_value'}
ascending order result is: ['l_value', 'A_value', 'a_value', 'z_value']
descending order result is: ['z_value', 'a_value', 'A_value', 'l_value']
```

`fish_common.splice_url_params(dic)``url ? ?key1=value1&key2=value2`**Param**

- dic: (dict)

Returns

- result: (string)

:

```
dic1 = {'key1': 'value1', 'key2': 'value2'}
print(splice_url_params(dic1))
```

:

```
?key1=value1&key2=value2
```

2.2 fish_system

`fish_system``fish_system.get_platform()
, sys.platform();`**Param**

-

Returns

- platform: (string) linux, osx, win

:


```
print('current os:', get_platform())
```

```
:
```

```
current os: osx
```

2.3 fish_file

fish_file

fish_file

fish_file.**check_sub_path_create**(*sub_path*)

```
,
```

Param

- sub_path: (string)

Returns

- (tuple)
- True: False:
- False: True:

```
:
```

```
#
sub_path = 'demo_sub_dir'
#
print('check sub path:', sub_path)
result = check_sub_path_create(sub_path)
print(result)
```

```
:
```

```
check sub path: demo_sub_dir
(True, False)
```

fish_file.**get_abs_filename_with_sub_path**(*sub_path*, *filename*)

Param

- sub_path: (string)
- filename: (string)

Returns

- (tuple) flag
- flag: (bool) True False
- abs_filename: (string) filename

```
:
```

```
print('get_abs_filename_with_sub_path')
# define sub dir
path_name = 'sub_dir'
# define not exists file
filename = 'test_file.txt'

abs_filename = get_abs_filename_with_sub_path(path_name, filename)
# return False and abs filename
print(abs_filename)

# define exists file
filename = 'demo.txt'
abs_filename = get_abs_filename_with_sub_path(path_name, filename)
# return True and abs filename
print(abs_filename)
```

:

```
(False, '/Users/****/Documents/dev_python/fishbase/demo/sub_dir/test_file.txt')
(True, '/Users/****/Documents/dev_python/fishbase/demo/sub_dir/demo.txt')
```

2.4 fish_csv CSV

`fish_csv.csv_file_to_list(csv_filename, deli=',', del_blank_row=True)`
csv list

Param

- csv_filename: (string) csv
- deli: (string) csv
- del_blank_row: (string)

Returns

- csv_list: (list) list

:

```
from fishbase.fish_file import *
from fishbase.fish_csv import *

def test_csv():
    csv_filename = get_abs_filename_with_sub_path('csv', 'test_csv.csv')[1]
    print(csv_filename)
    csv_list = csv_file_to_list(csv_filename)
    print(csv_list)

if __name__ == '__main__':
    test_csv()
```

2.5 fish_logger

fish_logger

```
set_log_file()
```

```
fish_logger.set_log_file(local_file=None)  
info fish_test.log.2018-05-27
```

Param

- local_fie: (string)

Returns

:

```
from fishbase.fish_logger import *  
from fishbase.fish_file import *  
  
log_abs_filename = get_abs_filename_with_sub_path('log', 'fish_test.log')[1]  
  
set_log_file(log_abs_filename)  
  
logger.info('test fish base log')  
logger.warn('test fish base log')  
logger.error('test fish base log')  
  
print('log ok')
```


f

- `fish_common`, 7
- `fish_csv`, 14
- `fish_file`, 13
- `fish_logger`, 14
- `fish_system`, 12

B

`big_file()` (`fish_common.GetMD5` static method), 8

C

`check_sub_path_create()` (in module `fish_file`), 13

`conf_as_dict()` (in module `fish_common`), 8

`csv_file_to_list()` (in module `fish_csv`), 14

F

`file()` (`fish_common.GetMD5` static method), 8

`fish_common` (module), 7

`fish_csv` (module), 14

`fish_file` (module), 13

`fish_logger` (module), 14

`fish_system` (module), 12

G

`get_abs_filename_with_sub_path()` (in module `fish_file`),
13

`get_platform()` (in module `fish_system`), 12

`get_time_uuid()` (in module `fish_common`), 9

`get_uuid()` (in module `fish_common`), 10

`GetMD5` (class in `fish_common`), 7

I

`if_json_contain()` (in module `fish_common`), 11

S

`set_log_file()` (in module `fish_logger`), 15

`Singleton` (class in `fish_common`), 8

`sorted_list_from_dict()` (in module `fish_common`), 11

`splice_url_params()` (in module `fish_common`), 12

`string()` (`fish_common.GetMD5` static method), 8