
json2xls Documentation

Release 0.1.3c

axiaoxin

Aug 10, 2017

Contents

1	3
2	5
3 API	9

jsonexceljsonExceljson

jsonjsonurljsonjson

CHAPTER 1

json2xls [GitHub](#) [PyPI](#) *pip* :

```
$ pip install json2xls
```

:

```
$ python setup.py install
```

json2xls [MIT](#)

API:

```
from json2xls import Json2Xls

# jsonexcel
json_data = u'''[
    {"": "John", "": 30, "": ""},
    {"": "Alice", "": 18, "": ""}
]'''
obj = Json2Xls('tests/json_strlist_test.xls', json_data)
obj.make()

# getjsonexcel
params = {
    'location': u'',
    'output': 'json',
    'ak': '5slgyqGDENN7Sy7pw29IUvrZ'
}
Json2Xls('tests/url_get_test.xls', "http://httpbin.org/get", params=params).make()

# postjsonexcel
post_data = {
    'location': u'',
    'output': 'json',
    'ak': '5slgyqGDENN7Sy7pw29IUvrZ'
}
Json2Xls('tests/url_post_test1.xls', "http://httpbin.org/post", method='post', post_
↪data=post_data, form_encoded=True).make()
# post_data
post_data = 'tests/post_data.json'
Json2Xls('tests/url_post_test2.xls', "http://httpbin.org/post", method='post', post_
↪data=post_data, form_encoded=True).make()

# jsonexcel
```

```

obj = Json2Xls('tests/json_line_test.xls', json_data='tests/line_data.json')
obj.make()
# jsonexcel
Json2Xls('tests/json_list_test.xls', json_data='tests/list_data.json').make()

# excel
def title_callback(self, data):
    '''use one of data record to generate excel title'''
    self.sheet.write_merge(0, 0, 0, 3, 'title', self.title_style)
    self.sheet.write_merge(1, 2, 0, 0, 'tag', self.title_style)
    self.sheet.write_merge(1, 2, 1, 1, 'ner', self.title_style)
    self.sheet.write_merge(1, 1, 2, 3, 'comment', self.title_style)
    self.sheet.row(2).write(2, 'x', self.title_style)
    self.sheet.row(2).write(3, 'y', self.title_style)

    self.sheet.write_merge(0, 0, 4, 7, 'body', self.title_style)
    self.sheet.write_merge(1, 2, 4, 4, 'tag', self.title_style)
    self.sheet.write_merge(1, 2, 5, 5, 'ner', self.title_style)
    self.sheet.write_merge(1, 1, 6, 7, 'comment', self.title_style)
    self.sheet.row(2).write(6, 'x', self.title_style)
    self.sheet.row(2).write(7, 'y', self.title_style)

    self.start_row += 3

def body_callback(self, data):

    key1 = ['title', 'body']
    key2 = ['tag', 'ner', 'comment']

    col = 0
    for ii, i in enumerate(key1):
        for ij, j in enumerate(key2):
            if j != 'comment':
                value = ', '.join(data[ii][i][j])
                self.sheet.row(self.start_row).write(col, value)
                col += 1
            else:
                for x in data[ii][i][j].values():
                    self.sheet.row(self.start_row).write(col, x)
                    col += 1
    self.start_row += 1

data = 'tests/callback_data.json'
j = Json2Xls('tests/callback.xls', data)
j.make(title_callback=title_callback, body_callback=body_callback)

```

:

```

# from json string
json2xls tests/cmd_str_test.xls '{"a":"a", "b":"b"}'
json2xls tests/cmd_str_test1.xls '[{"a":"a", "b":"b"}, {"a":1, "b":2}]'

# from file: whole file is a complete json data
json2xls tests/cmd_list_test.xls "`cat tests/list_data.json`"

# from file: each line is a json data
json2xls tests/cmd_line_test.xls tests/line_data.json

```

```
# from url
json2xls tests/cmd_get_test.xls http://httpbin.org/get
json2xls tests/cmd_post_test.xls http://httpbin.org/post -m post -d "hello json2xls"
↪' -h '{"X-Token': 'bolobolomi'}"
```

```
class json2xls.Json2Xls(xls_filename, json_data, method='get', params=None, post_data=None,
                       headers=None, form_encoded=False, dumps=False, sheet_name='sheet0',
                       title_style=None)
```

Json2Xls API

Parameters

- **xls_filename** (*string*) – excel
- **json_data** (*string*) – json jsonurl json json
- **method** (*string*) – urlurl get
- **params** (*dict*) – get None
- **post_data** (*dict*) – post None
- **headers** (*dict*) – urlHTTP (json)
- **form_encoded** (*bool*) – post False
- **sheet_name** (*string*) – Excelsheet sheet0
- **title_style** (*string*) – Excel None
- **json_dumps** (*function*) – ensure_asciijson.dumps() False
- **json_loads** (*function*) – object_pairs_hookjson.loads()
- **dumps** (*bool*) – exceljson_dumps False

auto_width (*row, col, value*)

- **row** (*int*) –
- **col** (*int*) –
- **value** (*int*) –

Parameters

flatten (*data_dict*, *parent_key*='', *sep*='.')
dictflattendict

Parameters

- **data_dict** (*dict*) – dict
- **parent_key** (*str*) – key
- **sep** (*str*) – key flatten “.”

make (*title_callback*=None, *body_callback*=None)
Excel

Parameters

- **title_callback** (*func*) – Excel None
- **body_callback** (*func*) – Excel None

A

`auto_width()` (`json2xls.Json2Xls` method), 9

F

`flatten()` (`json2xls.Json2Xls` method), 9

J

`Json2Xls` (class in `json2xls`), 9

`json2xls` (module), 1

M

`make()` (`json2xls.Json2Xls` method), 10