

---

# **rst Documentation**

***Release 0.1***

**Kushal Das**

March 19, 2013



# CONTENTS

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Why rst ? . . . . .	3
1.2	rst license . . . . .	3
<b>2</b>	<b>Installation</b>	<b>5</b>
2.1	Distribute & Pip . . . . .	5
2.2	Get the Code . . . . .	5
<b>3</b>	<b>Quickstart</b>	<b>7</b>
<b>4</b>	<b>API</b>	<b>9</b>
<b>5</b>	<b>Indices and tables</b>	<b>13</b>



An easy way to create rst document.

Contents:



# INTRODUCTION

## 1.1 Why rst ?

I was looking for a module to create reStructuredText documents through code for a long time and finally decided to write one for myself.

## 1.2 rst license

Copyright (C) 2012, Kushal Das

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the “Software”), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED “AS IS”, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.





# INSTALLATION

This part of the documentation covers the installation of rst. The first step to using any software package is getting it properly installed.

## 2.1 Distribute & Pip

Installing requests is simple with `pip`:

```
$ pip install rst
```

or, with `easy_install`:

```
$ easy_install rst
```

But, you really *shouldn't* do that.

## 2.2 Get the Code

rst is actively developed on GitHub, where the code is *always available*.

You can either clone the public repository:

```
git clone git://github.com/kushaldas/rst.git
```

Download the *tarball*:

```
$ curl -OL https://github.com/kushaldas/rst/tarball/master
```

Or, download the *zipball*:

```
$ curl -OL https://github.com/kushaldas/rst/tarball/master
```

Once you have a copy of the source, you can embed it in your Python package, or install it into your site-packages easily:

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



# QUICKSTART

Here is a quick example to just start using it.

```
#!/usr/bin/env python
from rst import rst

def main():
    doc = rst.Document('Title of the report')
    para = rst.Paragraph('Just another paragraph. We need few more of these.')
    doc.add_child(para)
    sec = rst.Section('Another', 2)
    doc.add_child(sec)
    para = rst.Paragraph('Can we do this? Yes we can.')
    doc.add_child(para)

    blt = rst.Orderedlist()
    blt.add_item('Red Hat')
    blt.add_item('Fedora')
    blt.add_item('Debian')

    doc.add_child(blt)

    sec2 = rst.Section('Why Python is awesome?', 2)
    doc.add_child(sec2)

    tbl = rst.Table('My friends', ['Name', 'Major Project'])
    tbl.add_item(('Ramki', 'Python'))
    tbl.add_item(('Pradepto', 'KDE'))
    tbl.add_item(('Nicubunu', 'Fedora'))
    doc.add_child(tbl)

    print doc.get_rst()

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



# API

**class** `rst.Document (title)`  
Returns a Document object.

```
>>> import rst
>>> doc = rst.Document('Title of the report')
>>> print doc.get_rst()
=====
Title of the report
=====
```

**add\_child (node)**  
Adds a Node object to the Document. Returns True in case of success.

**get\_rst ()**  
Returns the rst representation of the document in unicode format.

**save (path)**  
Saves the document in the given path.

**Parameters path** – Path to save the document.

**class** `rst.Paragraph (text='')`  
Represents a paragraph

**Parameters text** – Text to be present in the paragraph.

. doctest:

```
>>> import rst
>>> doc = rst.Document('Title of the report')
>>> para = rst.Paragraph('This is a paragraph. A long one.')
>>> doc.add_child(para)
True
>>> print doc.get_rst()
=====
Title of the report
=====

This is a paragraph. A long one.
```

**class** `rst.Section (title, depth=1)`  
Represents a Section object.

**Parameters**

- **depth** – Depth of the section, default is 1

- **text** – Title of the section

**class** `rst.Orderedlist`

Represents a Ordered List.

```
>>> import rst
>>> doc = rst.Document('Title of the report')
>>> blt = rst.Orderedlist()
>>> blt.add_item('Fedora')
>>> blt.add_item('Debian')
>>> doc.add_child(blt)
True
>>> print doc.get_rst()
=====
Title of the report
=====

1. Fedora
2. Debian
```

**add\_item**(*text*)

Adds a new text block in the Bulletlist.

**Parameters** **text** – text to be added in the list, remember it is ordered list.

**class** `rst.Bulletlist`

Represents a Bullet List.

```
>>> import rst
>>> doc = rst.Document('Title of the report')
>>> blt = rst.Bulletlist()
>>> blt.add_item('Fedora')
>>> blt.add_item('Debian')
>>> doc.add_child(blt)
True
>>> print doc.get_rst()
=====
Title of the report
=====

* Fedora
* Debian
```

**add\_item**(*text*)

Adds a new text block in the Bulletlist.

**Parameters** **text** – text to be added in the list.

**class** `rst.Table`(*title='', header=None, width=None*)

Represents a Table, (will be written in csv-table style)

```
>>> import rst
>>> doc = rst.Document('Title of the report')
>>> tbl = rst.Table('My friends', ['Name', 'Major Project'])
>>> tbl.add_item(('Ramki', 'Python'))
>>> tbl.add_item(('Pradeepto', 'Kde'))
>>> tbl.add_item(('Nicubunu', 'Fedora'))
>>> doc.add_child(tbl)
True
>>> print doc.get_rst()
=====
```

Title of the report  
=====

```
.. list-table:: My friends
   :header-rows: 1
```

```
   * - Name
     - Major Project
   * - Ramki
     - Python
   * - Pradeepto
     - Kde
   * - Nicubunu
     - Fedora
```

**add\_item**(*row*)

Adds a new row to the table.

**Parameters** *row* – list of items in the table.





# INDICES AND TABLES

- *genindex*
- *modindex*
- *search*