

---

# abc Documentation

*Release 1.0.0*

**WZZ**

**Aug 19, 2018**



---

## Contents

---

<b>1</b>		<b>1</b>
1.1	H1 – Top of Page Header . . . . .	1
<b>2</b>	<b>H2 – Page Sections</b>	<b>3</b>
2.1	H3 – Subsection . . . . .	3
2.1.1	Installlation . . . . .	4
2.1.1.1	. . . . .	4
2.1.1.2	H3 – Subsection . . . . .	5
2.1.2	QuickStart . . . . .	6
2.1.2.1	. . . . .	6
2.1.2.2	H3 – Subsection . . . . .	6
	<b>Bibliography</b>	<b>9</b>



- api
- datetime

**bold** and *italics*

- Some.
- Thing.
- Different.

## 1.1 H1 – Top of Page Header

There should only be one of these per page and this will also – when converting to pdf – be used for the chapters.



### 2.1 H3 – Subsection

COMPLEX TABLE:

Header 1	Header 2	Header 3
body row 1	column 2	column 3
body row 2	Cells may span columns.	
body row 3	Cells may span rows.	<ul style="list-style-type: none"> <li>• Cells</li> <li>• contain</li> <li>• blocks.</li> </ul>
body row 4		

SIMPLE TABLE:

Inputs		Output
A	B	A or B
False	False	False
True	False	True
False	True	True
True	True	True

[Docs for this project](#)

Here is something I want to talk about:

```
def my_fn(foo, bar=True):
    """A really useful function.

    Returns None
    """
```

This is inline if `__name__ == '__main__':`

I really like the `threading` module which has the `threading.Thread` class.

Here is a link `time.time()`.

Typical result

A paragraph containing only two colons indicates that the following indented or quoted text is a literal block.

```
Whitespace, newlines, blank lines, and
all kinds of markup (like *this* or
\this) is preserved by literal blocks.
```

```
The paragraph containing only '::'
will be omitted from the result.
```

The `::` may be tacked onto the very end of any paragraph. The `::` will be omitted if it is preceded by whitespace. The `::` will be converted to a single colon if preceded by text, like this:

```
It's very convenient to use this form.
```

Citation references, like `[CIT2002]`. Note that citations may get rearranged, e.g., to the bottom of the “page”.

Citation labels contain alphanumerics, underlines, hyphens and fullstops. Case is not significant.

Given a citation like , one

External hyperlinks, like `Python`.

Internal crossreferences, like `example`.

This is an example crossreference target.

**Danger:** Beware killer rabbits!

## 2.1.1 Installlation

### 2.1.1.1

- api
- datetime

**bold** and *italics*

- Some.
- Thing.
- Different.

## H1 – Top of Page Header

There should only be one of these per page and this will also – when converting to pdf – be used for the chapters.



### 2.1.1.2 H3 – Subsection

COMPLEX TABLE:

Header 1	Header 2	Header 3
body row 1	column 2	column 3
body row 2	Cells may span columns.	
body row 3	Cells may span rows.	<ul style="list-style-type: none"> <li>• Cells</li> <li>• contain</li> <li>• blocks.</li> </ul>
body row 4		

SIMPLE TABLE:

Inputs		Output
A	B	A or B
False	False	False
True	False	True
False	True	True
True	True	True

Docs for this project

Here is something I want to talk about:

```
def my_fn(foo, bar=True):
    """A really useful function.

    Returns None
    """
```

This is inline if `__name__ == '__main__':`

I really like the `threading` module which has the `threading.Thread` class.

Here is a link `time.time()`.

Typical result

A paragraph containing only two colons indicates that the following indented or quoted text is a literal block.

```
Whitespace, newlines, blank lines, and
all kinds of markup (like *this* or
\this) is preserved by literal blocks.
```

```
The paragraph containing only '::'
will be omitted from the result.
```

The `::` may be tacked onto the very end of any paragraph. The `:` will be omitted if it is preceded by whitespace. The `::` will be converted to a single colon if preceded by text, like this:

```
It's very convenient to use this form.
```

External hyperlinks, like `Python`.

Internal crossreferences, like `example`.

This is an example crossreference target.

**Danger:** Beware killer rabbits!

## 2.1.2 QuickStart

### 2.1.2.1

- api
- datetime

**bold** and *italics*

- Some.
- Thing.
- Different.

### H1 – Top of Page Header

There should only be one of these per page and this will also – when converting to pdf – be used for the chapters.

### 2.1.2.2 H3 – Subsection

COMPLEX TABLE:

Header 1	Header 2	Header 3
body row 1	column 2	column 3
body row 2	Cells may span columns.	
body row 3	Cells may span rows.	<ul style="list-style-type: none"><li>• Cells</li><li>• contain</li><li>• blocks.</li></ul>
body row 4		

SIMPLE TABLE:

Inputs		Output
A	B	A or B
False	False	False
True	False	True
False	True	True
True	True	True

[Docs for this project](#)

Here is something I want to talk about:

```
def my_fn(foo, bar=True):  
    """A really useful function.  
  
    Returns None  
    """
```

This is inline if `__name__ == '__main__':`

I really like the `threading` module which has the `threading.Thread` class.

Here is a link `time.time()`.

Typical result

A paragraph containing only two colons indicates that the following indented or quoted text is a literal block.

```
Whitespace, newlines, blank lines, and
all kinds of markup (like *this* or
\this) is preserved by literal blocks.
```

The paragraph containing only `:::` will be omitted from the result.

The `:::` may be tacked onto the very end of any paragraph. The `:::` will be omitted if it is preceded by whitespace. The `:::` will be converted to a single colon if preceded by text, like this:

```
It's very convenient to use this form.
```

External hyperlinks, like [Python](#).

Internal crossreferences, like [example](#).

This is an example crossreference target.

**Danger:** Beware killer rabbits!





---

## Bibliography

---

[CIT2002] A citation (as often used in journals).