
title: "test" author: "P J" --- = text :home: /home/hos :title: Text :author: P J :email: hos@gmail :icons: font :icon-set: fi :sectnums: :source-highlighter: rouge :rouge-style: github.light :doctype: book :toc: macro :toc: :toclevels: 4 :toc-title: My Content

Intro

Okay, let's do it.

Test

```
Unresolved directive in index.adoc - include::{home}/.cache/wal/colors-wal-st.h[]
```

Lists

Unordered lists

```
* level 1
** level 2
*** level 3
**** level 4
***** level 5
* level 1
```

Example 1. Output

- level 1
 - level 2
 - level 3
 - level 4
 - level 5
- level 1

Ordered lists

```
. level 1
.. level 2
... level 3
.... level 4
```

```
..... level 5
. level 1
```

Example 2. Output

1. level 1
 - a. level 2
 - i. level 3
 - A. level 4
 - I. level 5
2. level 1

TODO lists

- [*] checked
- [x] also checked
- [] not checked
- normal list item

Example 3. Output

- ☒ checked
- ☒ also checked
- ☐ not checked
- normal list item

Separating lists

- . Apples
- . Oranges
- //-
- . Walnuts
- . Almonds

Example 4. Output

1. Apples
2. Oranges
1. Walnuts
2. Almonds

Definitions

Normal definition list

```
first term:: definition of first term
section term:: definition of second term
```

Example 5. Output

first term
definition of first term

section term
definition of second term

Horizontal definition list

```
[horizontal]
first term:: definition of first term
section term:: definition of second term
```

Example 6. Output

first term	definition of first term
section term	definition of second term

Multi-line definition list

```
first term::  
definition of first term  
section term::  
definition of second term
```

Example 7. Output

first term
definition of first term

section term
definition of second term

Q&A

```
[qanda]  
What is Asciidoctor?::  
An implementation of the AsciiDoc processor in Ruby.  
What is the answer to the Ultimate Question?:: 42
```

Example 8. Output

1. *What is Asciidoctor?*
An implementation of the AsciiDoc processor in Ruby.
2. *What is the answer to the Ultimate Question?*
42

Mixed

```
Operating Systems::  
  Linux::  
    . Fedora  
      * Desktop  
    . Ubuntu  
      * Desktop  
      * Server  
  BSD:::
```

```
. FreeBSD
. NetBSD
Cloud Providers::
  PaaS:::
    . OpenShift
    . CloudBees
  IaaS:::
    . Amazon EC2
    . Rackspace
```

Example 9. Output

Operating Systems

Linux

1. Fedora
 - Desktop
2. Ubuntu
 - Desktop
 - Server

BSD

1. FreeBSD
2. NetBSD

Cloud Providers

PaaS

1. OpenShift
2. CloudBees

IaaS

1. Amazon EC2
2. Rackspace

NOTE | Note that the indentation is optional

Complex content in outline lists

* Every list item has at least one paragraph of content,
which may be wrapped, even using a hanging indent.

+

Additional paragraphs or blocks are adjoined by putting
a list continuation on a line adjacent to both blocks.

+

list continuation:: a plus sign (`{plus}`) on a line by itself

* A literal paragraph does not require a list continuation.

```
$ gem install asciidoctor
```

* AsciiDoc lists may contain any complex content.

+

```
[cols="2", options="header"]
```

```
|===
```

```
|Application
```

```
|Language
```

```
|AsciiDoc
```

```
|Python
```

```
|Asciidoctor
```

```
|Ruby
```

```
|===
```

Example 10. Output

- Every list item has at least one paragraph of content, which may be wrapped, even using a hanging indent.

Additional paragraphs or blocks are adjoined by putting a list continuation on a line adjacent to both blocks.

list continuation

a plus sign (+) on a line by itself

- A literal paragraph does not require a list continuation.

```
$ gem install asciidoctor
```

- AsciiDoc lists may contain any complex content.

Application	Language
AsciiDoc	Python
Asciidoctor	Ruby

Links

External

```
http://asciidoctor.org - automatic!  
http://asciidoctor.org[Asciidoctor]  
https://github.com/asciidoctor[Asciidoctor @ *GitHub*]
```

Example 11. Output

<http://asciidoctor.org> - automatic!
[Asciidoctor](#)
[Asciidoctor @ GitHub](#)

Relative

```
link:index.html[Docs]
```

Example 12. Output

[Docs](#)

Email and IRC

```
devel@discuss.arquillian.org  
mailto:devel@discuss.arquillian.org[Discuss Arquillian]  
mailto:devel-join@discuss.arquillian.org[Subscribe, Subscribe me, I want to  
join!]  
irc://irc.freenode.org/#asciidoctor
```

Example 13. Output

devel@discuss.arquillian.org
[Discuss Arquillian](mailto:devel@discuss.arquillian.org)
[Subscribe](mailto:devel-join@discuss.arquillian.org)

Inline anchors

```
[[bookmark-a]]Inline anchors make arbitrary content referenceable.  
anchor:bookmark-b[]Use a cross reference to link to this location.
```

Example 14. Output

Inline anchors make arbitrary content referenceable.
Use a cross reference to link to this location.

Internal cross references

```
See <<paragraphs>> to learn how to write paragraphs.  
Learn how to organize the document into <<section-titles,sections>>.
```

Example 15. Output

See [\[paragraphs\]](#) to learn how to write paragraphs.
Learn how to organize the document into [sections](#).

Images

```
image::rose.png[]  
image::rose.png[Rose]  
[[img-sunset]]  
image::rose.png[caption="Figure 1: ", title="A rose", alt="Rose", width="300",  
height="200", link="http://www.flickr.com/photos/javh/5448336655"]  
image::http://asciidoctor.org/images/octocat.jpg[GitHub mascot]
```


Example 16. Output

[rose] | *rose.png*

[Rose] | *rose.png*

[Rose] | *rose.png*

Figure 1: A rose

[GitHub mascot] | *<http://asciidoctor.org/images/octocat.jpg>*

blocks

blockquotes

After landing the cloaked Klingon bird of prey in Golden Gate park:

Everybody remember where we parked.

— Captain James T. Kirk, *Star Trek IV: The Voyage Home*

WIP