

# Asciidoctor Demo

Dan Allen

# Table of Contents

1. First Steps with AsciiDoc .....	1
1.1. Lists Upon Lists .....	2
2. ...and we're back! .....	3
2.1. Block Quotes and "Smart" Ones .....	4
3. Getting Literal .....	5
4. Wrap-up .....	5

This is a demonstration of Asciidoctor 1.5.4. And this is the preamble of this document.

### Purpose

This document exercises many of the features of AsciiDoc to test the Asciidoctor implementation.

#### TIP

If you want the output to look familiar, copy (or link) the AsciiDoc stylesheet, `asciidoc.css`, to the output directory.

#### NOTE

Items marked with TODO are either not yet supported or a work in progress.

## 1. First Steps with AsciiDoc

### Inline markup

- single quotes around a phrase place *emphasis*
- astericks around a phrase make the text **bold**
- double astericks around one or more letters in a word make those letters bold
- double underscore around a *substring* in a word emphasizes that substring
- use carrots around characters to make them <sup>super</sup>script
- use tildes around characters to make them <sub>sub</sub>script
- characters can be escaped using a \
  - for instance, you can escape a quote inside emphasized text like *Here's Johnny!*
- you can safely use reserved XML characters like `<`, `>` and `&`, which are escaped when rendering
- force a space between inline elements using the `{sp}` attribute
- hold text together with an intrinsic non-breaking space attribute, `{nbsp}`
- handle words with unicode characters like in the name Gregory Romé
- claim your copyright ©, registered trademark ® or trademark ™

You can write text [with inline links](#), optionally using an explicit [link prefix](#). In either case, the link can have a [query string](#).

If you want to break a line

just end it in a + sign  
and continue typing on the next line.

## 1.1. Lists Upon Lists

### *Adjacent lists*

- this list
- should join
- to have
- four items

### *Ordered lists*

1. These items
2. will be auto-numbered
  - a. and can be nested
3. A numbered list can nest
  - unordered
  - list
  - items

### *Statement*

I swear I left it in *Guy's* car. Let's go look for it.

### *term*

definition line two

### *another term*

another definition, which can be literal (indented) or regular paragraph

This should be a standalone paragraph, not grabbed by the definition list.

- first level written on two lines
- first level

with this literal text

- first level

with more literal text

- second level
  - third level
    - fourth level
- back to first level

Let's make a horizontal rule...

---

then take a break.

## 2. ...and we're back!

Want to see a [Tiger]?

Do you feel safer with the tiger in a box?

[ tiger ] | *tiger.png*

*Figure 1. Tiger in a box*

Unresolved directive in demo.adoc - include::includes/include.adoc[]

*Asciidoctor usage example. The listing should contain 5 lines.*

```
require 'asciidoctor'

doc = Asciidoctor.load '*This* is http://asciidoc.org[AsciiDoc]!', header_footer: true

puts doc.render
```

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<meta name="generator" content="AsciiDoctor 0.1.4">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Untitled</title>
<link rel="stylesheet" href="./asciidoctor.css">
</head>
<body class="article">
<div id="header">
</div>
<div id="content">
<div class="paragraph">
<p><strong>This</strong> is <a href="http://asciidoc.org">AsciiDoc</a>!</p>
</div>
</div>
<div id="footer">
<div id="footer-text">
Last updated 2014-01-28 20:11:37 MST
</div>
</div>
</body>
</html>
```

## 2.1. Block Quotes and “Smart” Ones

AsciiDoc is *so* **powerful**!

This verse comes to mind.

La la la

Here’s another quote:

When you have eliminated all which is impossible, then whatever remains, however improbable, must be the truth.

— Sir Arthur Conan Doyle, *The Adventures of Sherlock Holmes*

“Get moving!” he shouted.

## 3. Getting Literal

Want to get literal? Just prefix a line with a space (just 1 space will do).

I'll join that party, too.

We forgot to mention in [Ordered lists](#) that you can change the numbering style.

- a. first item (yeah!)
- b. second item, looking *so mono*
- c. third item, *mono* it is!

## 4. Wrap-up

**NOTE** | AsciiDoc is quite cool, you should try it!

*Info*

Go to this URL to learn more about it:

**TIP**

- <http://asciidoc.org>

Or you could return to the [\[first-steps-with-asciidoc\]](#) or [Purpose](#).

Here's a reference to the definition of [\[another\\_term\]](#), in case you forgot it.

**NOTE** | One more thing. Happy documenting!

When all else fails, head over to <http://google.com>.