

Tutorial

Cesar Hernandez

Version 1.0-SNAPSHOT, 2019-04-08 23:20:17 UTC

Table of Contents

1. Introduction	1
2. Prerequisites	1
3. Creating Base Project Structure	1
3.1. Create a Maven project	1
3.2. Add Asciidoctor to the project	2
3.3. Create your first AsciiDoc file	3
3.4. Generate your first HTML and PDF files	4

1. Introduction

This tutorial shows how to start a Java Maven project from scratch and built all the necessary structure to have AsciiDoctor capabilities along with spell checking for different languages as part of the project tests.

2. Prerequisites

- Java Development Kit >= 7
- Maven >= 3.3.9

3. Creating Base Project Structure

3.1. Create a Maven project

On a terminal execute the following command:

```
mvn archetype:generate -DgroupId=asciidoctor.doc.demo -DartifactId=book-demo
-DarchetypeArtifactId=maven-archetype-quickstart -DarchetypeVersion=1.4
-DinteractiveMode=false
```

On a terminal `cd` into the generated folder:

```
cd book-demo/
```

The generated folder structure should look like this:

```
|— pom.xml
|— src
|   |— main
|   |   |— java
|   |   |   |— asciidoctor
|   |   |   |   |— doc
|   |   |   |   |   |— demo
|   |   |   |   |       App.java
|   |— test
|   |   |— java
|   |   |   |— asciidoctor
|   |   |   |   |— doc
|   |   |   |   |   |— demo
|   |   |   |       AppTest.java
```

The `pom.xml` file contains at this point the basic project structure.

3.2. Add Asciidoctor to the project

On a text editor, open the `pom.xml` file and add the following sections:

- Update the `<properties>` section with the following content:

```
<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <maven.compiler.source>1.8</maven.compiler.source>
  <maven.compiler.target>1.8</maven.compiler.target>
  <asciidoctor.maven.plugin.version>1.5.7.1</asciidoctor.maven.plugin.version>
  <asciidoctorj.pdf.version>1.5.0-alpha.16</asciidoctorj.pdf.version>
  <asciidoctorj.version>1.5.8.1</asciidoctorj.version>
  <maven.build.timestamp.format>yyyy-MM-dd HH:mm:ss
z</maven.build.timestamp.format>
</properties>
```

- In the `<build>` section, add the following content after the `</pluginManagement>` tag:

```
<plugins>
  <plugin>
    <groupId>org.asciidoctor</groupId> ①
    <artifactId>asciidoctor-maven-plugin</artifactId>
    <version>${asciidoctor.maven.plugin.version}</version>

    <dependencies> ②
      <dependency>
        <groupId>org.asciidoctor</groupId>
        <artifactId>asciidoctorj-pdf</artifactId> ③
        <version>${asciidoctorj.pdf.version}</version>
      </dependency>
      <dependency>
        <groupId>org.asciidoctor</groupId>
        <artifactId>asciidoctorj</artifactId>
        <version>${asciidoctorj.version}</version>
      </dependency>
    </dependencies>

    <executions>
      <execution>
        <id>output-html</id> ④
        <phase>generate-resources</phase>
        <goals>
          <goal>process-asciidoc</goal>
        </goals>
        <configuration>
          <backend>html5</backend>
          <sourceHighlighter>highlight.js</sourceHighlighter>
          <preserveDirectories>true</preserveDirectories>
```

```

        <attributes>
          <toc>left</toc>
          <sectanchors>>true</sectanchors>
        </attributes>
      </configuration>
    </execution>
    <execution>
      <id>generate-pdf-doc</id> ⑤
      <phase>generate-resources</phase>
      <goals>
        <goal>process-asciidoc</goal>
      </goals>
      <configuration>
        <backend>pdf</backend>
        <sourceHighlighter>highlight.js</sourceHighlighter>
        <preserveDirectories>>true</preserveDirectories>
        <attributes>
          <pagenums/>
          <toc/>
          <idprefix/>
          <idseparator>-</idseparator>
        </attributes>
      </configuration>
    </execution>
  </executions>

  <configuration> ⑥
    <attributes>
      <project-version>${project.version}</project-version>
      <revdate>${maven.build.timestamp}</revdate>
      <sectnums>>true</sectnums>
      <icons>font</icons>
    </attributes>
  </configuration>
</plugin>
</plugins>

```

- ① Reference to `asciidoctor-maven-plugin`.
- ② Dependencies used by the `asciidoctor-maven-plugin`.
- ③ Provides PDF export capabilities to the project.
- ④ Configuration for HTML export.
- ⑤ Configuration for PDF export.
- ⑥ Global configuration for both executions.

3.3. Create your first AsciiDoc file

- Create the default AsciiDoc folder:

```
mkdir src/main/asciidoc
```

- Inside `src/main/asciidoc` folder create the file `index_en.adoc` with the following content:

```
= Tutorial ①
:nofooter: ②
Author ③
{project-version}, {revdate} ④

== Chapter 1 ⑤
This is my first chapter.
```

- ① Set the document main Header.
- ② Removes from the document the auto generated header.
- ③ Author name.
- ④ Maps the project version and revision date from the `pom.xml` file.
- ⑤ Set the first document sub header.

3.4. Generate your first HTML and PDF files

- On a terminal execute the following command:

```
mvn clean generate-resources
```

- The following files will be generated under the folder `target/generated-docs/`
 - `index_en.html`
 - `index_en.pdf`

The generated html version should look similar to:

