# asciidoctor-maven-plugin-example (0.0.1)

Maksim Kostromin

# Table of Contents

Travis CI status: [Build Status]

# Chapter 1. Introduction

This project contains example of `asciidoctor-maven-plugin` usage

Read project reference documentation

Project generated by generator-jvm yeoman generator (java)

# Chapter 2. Implementation

*plugion configuration*

```xml
<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xmlns="http://maven.apache.org/POM/4.0.0"
         xsi:schemaLocation="
         http://maven.apache.org/POM/4.0.0
         http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <build>
    <plugins>
      <plugin>
        <groupId>org.asciidoctor</groupId>
        <artifactId>asciidoctor-maven-plugin</artifactId>
        <version>1.5.6</version>
        <dependencies>
          <dependency>
            <groupId>org.asciidoctor</groupId>
            <artifactId>asciidoctorj-pdf</artifactId>
            <version>1.5.0-alpha.16</version>
          </dependency>
        </dependencies>
        <executions>
          <execution>
            <id>output-pdf-doc</id>
            <phase>generate-resources</phase>
            <goals>
              <goal>process-asciidoc</goal>
            </goals>
            <configuration>
              <backend>pdf</backend>
              <!--
                Since 1.5.0-alpha.9 PDF back-end can use 'rouge' as well as 'coderay'
                for source highlighting
              -->
              <!--
                Due to a known issue on windows, it is recommended to use 'coderay'
                until an new version of 'rouge' is released.

                see discussions: https://github.com/asciidoctor/asciidoctor-maven-examples/pull/58
                                 https://github.com/asciidoctor/asciidoctorj-pdf/issues/3
                                 https://github.com/jneen/rouge/issues/661
              -->
              <sourceHighlighter>coderay</sourceHighlighter>
            </configuration>
          </execution>
          <execution>
            <id>output-html</id>
```

```xml
          <phase>generate-resources</phase>
          <goals>
            <goal>process-asciidoc</goal>
          </goals>
          <configuration>
            <backend>html</backend>
          </configuration>
        </execution>
        <execution>
          <id>output-docbook</id>
          <phase>generate-resources</phase>
          <goals>
            <goal>process-asciidoc</goal>
          </goals>
          <configuration>
            <backend>docbook</backend>
          </configuration>
        </execution>
      </executions>
      <configuration>
        <sourceDirectory>docs</sourceDirectory>
        <preserveDirectories>true</preserveDirectories>
        <outputDirectory>target/generated-docs</outputDirectory>
        <relativeBaseDir>true</relativeBaseDir>
        <headerFooter>true</headerFooter>
        <imagesDir>images</imagesDir>
        <sourceHighlighter>coderay</sourceHighlighter>
        <attributes>
          <toc>left</toc>
          <linkcss>false</linkcss>
          <doctype>book</doctype>
          <ruby>erubis</ruby>
          <encoding>${project.build.sourceEncoding}</encoding>
          <linkattrs>true</linkattrs>
          <numbered>true</numbered>
          <sectanchors>true</sectanchors>
          <sectlink>true</sectlink>
          <idprefix/>
          <docinfo1/>
          <icons>font</icons>
          <idseparator>-</idseparator>
          <setanchors>true</setanchors>
          <source-highlighter>coderay</source-highlighter>
          <toc-title>Table of Contents</toc-title>
          <project-name>${project.artifactId}</project-name>
          <root-project-name>${project.artifactId}</root-project-name>
          <project-version>${project.version}</project-version>
          <github-url>https://github.com/daggerok/${project.artifactId}</github-url>
        </attributes>
      </configuration>
    </plugin>
```

```
      </plugins>

      <defaultGoal>clean process-resources</defaultGoal>
   </build>
</project>
```

ℹ️ this plugin configured for base directory: `./docs`

# Chapter 3. Test usage

*build fat jar*

```
./mvnw clean process-resources
```

*run app*

```
npm i -g serve
serve ./target/generated-docs
open http://127.0.0.1:3000
```

# Chapter 4. Build documentation

*build documentation*

```
./mvnw process-resources
```

# Chapter 5. Links

- [GitHub repo](#)
- [GitHub pages](#)

# Chapter 6. Other links

- Asciidoctor maven examples
- asciidoctor-maven-plugin page
- asciidoctor-maven-plugin example