About the Tutorial

This tutorial provides a basic understanding of Apache POI library and its features. Here we will learn how to read, write, and manage MS-PowerPoint documents using Java programs.

Audience

This tutorial is designed for all the readers working on Java and especially those who want to create, read, write, and modify PPT files using Java.

Prerequisites

A general awareness of Java programming with JDK1.5 or later versions and IO concepts in Java are the only prerequisites to understand this tutorial.

Copyright & Disclaimer

© Copyright 2014 by Tutorials Point (I) Pvt. Ltd.

All the content and graphics published in this e-book are the property of Tutorials Point (I) Pvt. Ltd. The user of this e-book is prohibited to reuse, retain, copy, distribute or republish any contents or a part of contents of this e-book in any manner without written consent of the publisher.

The API of Apache POI contains a number of methods and classes. In this tutorial, we have used only some of those for demonstration purpose. We encourage the readers to refer the complete API document for a comprehensive understanding.

We strive to update the contents of our website and tutorials as timely and as precisely as possible, however, the contents may contain inaccuracies or errors. Tutorials Point (I) Pvt. Ltd. provides no guarantee regarding the accuracy, timeliness or completeness of our website or its contents including this tutorial. If you discover any errors on our website or in this tutorial, please notify us at contact@tutorialspoint.com
# Table of Contents

About the Tutorial........................................................................................................1

Audience ........................................................................................................................1

Prerequisites ..................................................................................................................1

Copyright & Disclaimer ...............................................................................................1

Table of Contents .........................................................................................................2

1. APACHE POI – OVERVIEW .......................................................................................5

   What is Apache POI? .....................................................................................................5

   Components of Apache POI .......................................................................................5

2. FLAVORS OF JAVA PPT API ..................................................................................7

   Aspose Slides for Java ................................................................................................7

   Apache POI ..................................................................................................................7

3. APACHE POI – INSTALLATION ...............................................................................9

   System Requirements ..............................................................................................9

   Step 1: Verify your Java Installation ........................................................................9

   Step 2: Set your Java Environment ...........................................................................10

   Step 3: Install Apache POI Library ..........................................................................10

4. CLASSES AND METHODS .....................................................................................13

   Presentation ..............................................................................................................13

   Slide ..........................................................................................................................14

   Slide Master ............................................................................................................15

   Slide Layout ............................................................................................................15

   Text Paragraph .......................................................................................................16

   Text Run ..................................................................................................................17
5. PRESENTATION ............................................................................................................. 20
   Creating Empty Presentation ...................................................................................... 20
   Editing an Existing Presentation ................................................................................ 22

6. SLIDE LAYOUTS .................................................................................................................. 25
   Available Slide layouts .............................................................................................. 25
   Title Layout .................................................................................................................. 27
   Title and content Layout ............................................................................................ 31

7. MANAGEMENT OF SLIDES ............................................................................................... 36
   Changing a Slide ............................................................................................................ 36
   Reordering Slides ......................................................................................................... 39
   Deleting Slides ............................................................................................................... 43

8. IMAGES ............................................................................................................................. 47
   Adding Image ................................................................................................................ 47
   Reading Image .............................................................................................................. 50

9. CREATING HYPERLINKS ................................................................................................. 53
   Creating Hyperlinks .................................................................................................... 53

10. READING SHAPES ............................................................................................................ 58
    Reading Shapes from a Presentation ........................................................................... 58

11. FORMATTING TEXT ........................................................................................................ 61
    Formatting Text in a Presentation ............................................................................. 61
12. MERGING ........................................................................................................................................68
   Merging Multiple Presentations........................................................................................................68

13. PPT TO IMAGE ................................................................................................................................73
   Converting Presentation to Image .....................................................................................................73
Many a time, a software application is required to generate reports in Microsoft Office file format. Sometimes, an application is even expected to receive MS-Office files as input data.

Any Java programmer who wants to produce MS Office files as output must use a predefined and read-only API to do so.

What is Apache POI?

Apache POI is a popular API that allows programmers to create, modify, and display MS-Office files using Java programs. It is an open source library developed and distributed by Apache Software Foundation. It contains classes and methods to decode the user input data, or a file into MS Office documents.

Components of Apache POI

Apache POI contains classes and methods to work on all OLE2 Compound documents of MS-Office. The list of components of this API is given below:

- **POIFS** (Poor Obfuscation Implementation File System): This component is the basic factor of all other POI elements. It is used to read different files explicitly.
- **HSSF** (Horrible SpreadSheet Format): It is used to read and write .xls format of MS-Excel files.
- **XSSF** (XML SpreadSheet Format): It is used for .xlsx file format of MS-Excel.
- **HPSF** (Horrible Property Set Format): It is used to extract property sets of the MS-Office files.
- **HWPF** (Horrible Word Processor Format): It is used to read and write .doc extension files of MS-Word.
- **XWPF** (XML Word Processor Format): It is used to read and write .docx extension files of MS-Word.
- **HSLF** (Horrible Slide Layout Format): It is used to read, create, and edit PowerPoint presentations.
- **HDGF** (Horrible DiaGram Format): It contains classes and methods for MS-Visio binary files.
- **HPBF** (Horrible PuBlisher Format): It is used to read and write MS-Publisher files.
This tutorial guides you through the process of working on Microsoft PowerPoint presentation using Java. Therefore the discussion is confined to **XSLF component**.

**Note:** Older versions of POI support binary file formats such as doc, xls, ppt, etc. Version 3.5 onwards, POI supports OOXML file formats of MS-Office such as docx, xlsx, pptx, etc.
This chapter takes you through some of the flavors of Java PowerPoint API and their features. There are many vendors who provide Java PPT related APIs; some of them are considered in this chapter.

### Aspose Slides for Java

Aspose slides for Java is a purely licensed Java PPT API, developed and distributed by the vendor Aspose. The latest version of this API is 8.1.2, released in July 2014. It is a rich and heavy API (combination of plain Java classes and AWT classes) for designing the PPT component that can read, write, and manage slides.

The common uses of this API are as follows:

- Build dynamic presentations
- Render and print high-fidelity presentations
- Generate, edit, convert, and print presentations

### Apache POI

Apache POI is a 100% open source library provided by Apache Software Foundation. Most of the small and medium scale application developers depend heavily on Apache POI (HSLF + XSLF). It supports all the basic features of PPT libraries; however, rendering and text extraction are its main features. Given below is the architecture of Apache POI for PPT.
Architecture – Apache POI for PPT
This chapter takes you through the process of setting up Apache POI on Windows and Linux based systems. Apache POI can easily be installed and integrated with your current Java environment, following a few simple steps without any complex setup procedures. User administration is required for installation.

**System Requirements**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>JDK</strong></td>
<td>Java SE 2 JDK 1.5 or above</td>
</tr>
<tr>
<td><strong>Memory</strong></td>
<td>1 GB RAM (recommended)</td>
</tr>
<tr>
<td><strong>Disk Space</strong></td>
<td>No minimum requirement</td>
</tr>
<tr>
<td><strong>Operating System Version</strong></td>
<td>Windows XP or above, Linux</td>
</tr>
</tbody>
</table>

Let us now proceed with the steps to install Apache POI.

**Step 1: Verify your Java Installation**

First of all, you need to have Java Software Development Kit (SDK) installed on your system. To verify this, execute any of the following two commands depending on the platform you are working on.

If the Java installation has been done properly, then it will display the current version and specification of your Java installation. A sample output is given in the following table.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Command</th>
<th>Sample Output</th>
</tr>
</thead>
</table>
| Windows  | Open command console and type: \>java –version | Java version "1.7.0_60"
Java (TM) SE Run Time Environment (build 1.7.0_60-b19)
Java Hotspot (TM) 64-bit Server VM (build 24.60-b09,mixed mode) |
| Linux    | Open command terminal and type: $java –version | java version "1.7.0_25"
Open JDK Runtime Environment (rhel-2.3.10.4.el6_4-x86_64)
Open JDK 64-Bit Server VM (build 23.7-b01, mixed mode) |
We assume that the readers of this tutorial have Java SDK version 1.7.0_60 installed on their system.

In case you do not have Java SDK, download its current version from http://www.oracle.com/technetwork/java/javase/downloads/index.html and install it.

**Step 2: Set your Java Environment**

Set the environment variable JAVA_HOME to point to the base directory location where Java is installed on your machine. For example,

<table>
<thead>
<tr>
<th>Platform</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>Set JAVA_HOME to C:\ProgramFiles\java\jdk1.7.0_60</td>
</tr>
<tr>
<td>Linux</td>
<td>Export JAVA_HOME=/usr/local/java-current</td>
</tr>
</tbody>
</table>

Append the full path of Java compiler location to the System Path.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>Append the String &quot;C:\Program Files\Java\jdk1.7.0_60\bin&quot; to the end of the system variable PATH.</td>
</tr>
<tr>
<td>Linux</td>
<td>Export PATH=$PATH:$JAVA_HOME/bin/</td>
</tr>
</tbody>
</table>

Execute the command **java -version** from the command prompt as explained above.

**Step 3: Install Apache POI Library**

Download the latest version of Apache POI from http://poi.apache.org/download.html and unzip its contents to a folder from where the required libraries can be linked to your Java program. Let us assume the files are collected in a folder on C drive.

The following images show the directories and the file structures inside the downloaded folder:
Add the complete path of the five **jars** as highlighted in the above image to the CLASSPATH.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Description</th>
</tr>
</thead>
</table>
| Windows  | Append the following strings to the end of the user variable CLASSPATH:  
  “C:\poi-3.9\poi-3.9-20121203.jar;”  
  “C:\poi-3.9\poi-ooxml-3.9-20121203.jar;”  
  “C:\poi-3.9\poi-ooxml-schemas-3.9-20121203.jar;”  
  “C:\poi-3.9\ooxml-lib\dom4j-1.6.1.jar;”  
  “C:\poi-3.9\ooxml-lib\xmlbeans-2.3.0.jar; ;”  |
| Linux    | Export CLASSPATH=CLASSPATH:  
  /usr/share/poi-3.9/poi-3.9-20121203.tar:  
  /usr/share/poi-3.9/poi-ooxml-schemas-3.9-20121203.tar:  
  /usr/share/poi-3.9/poi-ooxml-3.9-20121203.tar:  
  /usr/share/poi-3.9/ooxml-lib/dom4j-1.6.1.tar:  
  /usr/share/poi-3.9/ooxml-lib/xmlbeans-2.3.0.tar |
4. CLASSES AND METHODS

In this chapter, we will learn about a few classes and methods under Apache POI API that are crucial to work on PPT files using Java programs.

Presentation

To create and manage a presentation, you have a class called XMLSlideShow in the package org.apache.poi.xslf.usermodel. Given below are some important methods and a constructor of this class.

**Class:** XMLSlideShow

**Package:** org.apache.poi.xslf.usermodel

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Constructor and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>XMLSlideShow(java.io.InputStream inputStream)</td>
</tr>
<tr>
<td></td>
<td>You can instantiate this class by passing an inputstream class object to it.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Methods and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>int addPicture (byte[] pictureData, int format)</td>
</tr>
<tr>
<td></td>
<td>Using this method, you can add a picture to a presentation.</td>
</tr>
<tr>
<td>2</td>
<td>XSLFSlide createSlide()</td>
</tr>
<tr>
<td></td>
<td>Creates a blank slide in a presentation.</td>
</tr>
<tr>
<td>3</td>
<td>XSLFSlide createSlide(XSLFSlideLayout layout)</td>
</tr>
<tr>
<td></td>
<td>Creates a slide with a given slide layout.</td>
</tr>
<tr>
<td>4</td>
<td>java.util.List&lt;XSLFPictureData&gt; getAllPictures()</td>
</tr>
<tr>
<td></td>
<td>Returns an array of all the pictures in a presentation.</td>
</tr>
<tr>
<td>5</td>
<td>java.awt.Dimension getPageSize()</td>
</tr>
<tr>
<td></td>
<td>Using this method, you can get to know the current page size.</td>
</tr>
</tbody>
</table>
To create and manage a slide in a presentation, the methods of the **XSLFSlide** class are used. Some important methods of this class are mentioned below.

**Class**: XSLFSlide

**Package**: org.apache.poi.xslf.usermodel

<table>
<thead>
<tr>
<th>S. No.</th>
<th>Methods and Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>XSLFBackground getBackground()</td>
</tr>
<tr>
<td></td>
<td>Returns the <strong>XSLFBackground</strong> object which can be used to retrieve details like color and anchor of the background of the slide. You can also draw shapes in the slide using this object.</td>
</tr>
<tr>
<td>2</td>
<td>XSLFSlideLayout getSlideLayout()</td>
</tr>
<tr>
<td></td>
<td>Provides access to the <strong>XSLFSlideLayout</strong> object of the current slide.</td>
</tr>
<tr>
<td>3</td>
<td>XSLFSlideMaster getSlideMaster()</td>
</tr>
<tr>
<td></td>
<td>Provides access to the slide master of the current slide.</td>
</tr>
</tbody>
</table>
End of ebook preview
If you liked what you saw...
Buy it from our store @ https://store.tutorialspoint.com