**EXCEL EXPORT UTILTIY**

**WRITING EXCEL FILES IN JAVA**

|  |  |
| --- | --- |
| Team: | PRISM |
| Author: | Mahesh Chand |
| Date: | 2016-09-20 |
|  |  |

|  |
| --- |
| **TABLE OF CONTENTS** |

[1. DOCUMENT CONTROL 3](#_Toc462151271)

[1.1 EXCEL EXPORT UTILITY OVERVIEW 4](#_Toc462151272)

[1.1.1 Apache POI API Basics 4](#_Toc462151273)

[1.1.2 Add Apache POI dependency 4](#_Toc462151274)

[1.1.3 Writing data to excel file using database (object-oriented concept) 5](#_Toc462151275)

[1.1.4 Writing data to excel file using UI (object-oriented concept) 6](#_Toc462151276)

[1.1.5 Adding Styles to Cell 6](#_Toc462151277)

# DOCUMENT CONTROL

|  |  |  |
| --- | --- | --- |
| **TEMPLATE CONTROL** | | |
| **1.0 Template Control** | | |
| **Author** | **Reviewer** | **Authorizer** |
| Mahesh Chand .D |  |  |
|  |  |  |
| **1.1 Template History** | | |
| **Version No.** | **Issue Date** | **Reason for Amendment** |
| 1.0 |  |  |
| 1.1 |  |  |

## EXCEL EXPORT UTILITY OVERVIEW

##### This document explains how to create a spread sheet and manipulate it using Java. Spread sheet is a page in an Excel file; it contains rows and columns with specific names. This utility is using apache poi library to export data to excel file.

### Apache POI API Basics

##### The fundamental interfaces include Workbook, Sheet, Row and Cell. For basic formatting, use the Cell Style and Font interfaces. Concrete implementing classes include:

##### Excel 2003: HSSFWorkbook, HSSFSheet, HSSFRow, HSSFCell, etc.

##### Excel 2007: XSSFWorkbook, XSSFSheet, XSSFRow, XSSFCell, etc.

##### But I recommend using the common interfaces for greater flexibility with both Excel formats 2003 (XLS) and 2007(XLSX).

##### Here are the basic steps for writing an Excel file:

##### Create a Workbook.

##### Create a Sheet.

##### Repeat the following steps until all data is processed:

##### Create a Row.

##### Create Cells in a Row. Apply formatting using Cell Style.

##### Write to an OutputStream.

##### Close the output stream.

### Add Apache POI dependency

Make sure to add apache poi jars to your project.

1. If your project uses Maven as dependency management, add following in your Pom.xml file

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi</artifactId>

<version>3.9</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml</artifactId>

<version>3.9</version>

</dependency>

<dependency>

<groupId>org.apache.poi</groupId>

<artifactId>poi-ooxml-schemas</artifactId>

<version>3.9</version>

</dependency>

<dependency>

<groupId>commons-beanutils</groupId>

<artifactId>commons-beanutils</artifactId>

<version>1.8.3</version>

</dependency>

<dependency>

<groupId>dom4j</groupId>

<artifactId>dom4j</artifactId>

<version>1.6</version>

</dependency>

<dependency>

<groupId>org.apache.xmlbeans</groupId>

<artifactId>xmlbeans</artifactId>

<version>2.3.0</version>

</dependency>

1. If you are not using Maven then you can directly add required JAR files in your class path.

### Writing data to excel file using database (object-oriented concept)

##### In this approach, utility is going to get excel file headers list and actual data list from database. Please follow the below steps to achieve this concept

1. Create a new table to store excel header names and header name specific bean property names. Please see the table structure as below.

COLUM Name Type

----------- -------- -------------

EXPORTID NUMBER(5)

HEADERNAME VARCHAR2(200)

COLNAME VARCHAR2(200)

EXPORTORDER NUMBER(5)

PAGEID NUMBER(7)

ACTIVE NUMBER(1)

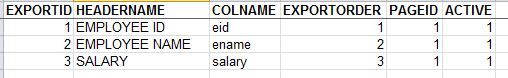
Ex: --

Bean Name : Employee.

Bean properties name : eid, ename, salary.

Excel header names : EMPLOYEE ID, EMPLOYEE NAME, and SALARY.

In this table, we are mapping bean property name and header name as below.



1. Get the headers list from the above mentioned table and get the bean data list from using existing services from the project.
2. Pass these two lists (header list, bean list) to utility code, utility will automatically create excel file.

### Writing data to excel file using UI (object-oriented concept)

##### In this approach, utility is going to get **excel file headers list from UI** and actual data (bean list) list from database. Please follow the below steps to achieve this concept

1. Get the headers list from the UI and get the bean data list from using existing services from the project.
2. Get the headers list from the above mentioned table and get the bean data list from using existing services from the project.
3. Pass these two lists (header list, bean list) to utility code, utility will automatically create excel file.

### Adding Styles to Cell

##### This utility providing header cells format and data cells format functionality also. This utility format methods will format cells with bold, font, background and borders attributes .This utility also providing cell types (*TEXT, DATE, FLOAT, INTEGER, LONG, MONEY, PERCENTAGE*) functionality