

# SQL Exporter Pipe

# Table of Contents

INTRODUCTION ..... 3

CHAPTER 1. GENERAL INFORMATION ..... 4

CHAPTER 2. PIPE PARAMETERS ..... 5

CHAPTER 3. CSV FILE FORMAT ..... 7

CHAPTER 4. EXAMPLE OF EXPORT IN XLSX FORMAT ..... 8

# Introduction

This document provides information on working with the pipe `com.openwaygroup.pipe.sql_exporter.jar`.

Warnings and information messages are marked as follows:



Warnings about potentially hazardous situations or actions.



Messages with information about important features, additional options, or the best use of certain system functions.

# Chapter 1. General information

The pipe `com.openwaygroup.pipe.sql_exporter.jar` is used to export arbitrary data from the database (DB) to a text file in CSV format (see RFC 4180) or to an XLSX format file.


An example of running the pipe is implemented for the `sy_handbook` table, menu item "Full → DB Administrator Utilities → System Utilities → Custom Handbook Export".

## Chapter 2. Pipe parameters

The pipe `com.openwaygroup.pipe.sql_exporter.jar` has the following parameters, which are specified in the tables.

### *Generic pipe parameters*

Parameter	Parameter description
OUTPUT_FORMAT	Exported file format: <ul style="list-style-type: none"><li>• PLAIN – CSV format</li><li>• EXCEL – XLSX format</li></ul>
QUERY	Text of the SQL query to select data from the DB (SELECT statement).
QUERY_FILE	Path to the file with the SQL query. It is recommended to use this parameter for long and complex queries.
OUTPUT_FILE	Path to the file to which the query's results will be written (in CSV format).
OUTPUT_DIR	Path to the directory to which exported files will be written. When this parameter is used, the query must contain the FILE_NAME column for determining the name of the exported file. The FILE_NAME column is not written to the exported file. For XLSX files, queries may contain the SHEET_NAME column to separate by EXCEL file sheets.
FETCH_SIZE	Number of rows fetched per DB round trip. The parameter makes it possible to reduce the load on the DB and improve performance.

 Only one parameter pair should be set: QUERY or QUERY\_FILE, OUTPUT\_FILE or OUTPUT\_DIR.

### *Specific pipe parameters for exporting in CSV format*

 Obsolete names of parameters are indicated in brackets; these names can be used up to WAY4 Cards version 03.51.30.

PLAIN_OUTPUT_ENCODING (OUTPUT_ENCODING)	Exported file encoding. By default, UTF-8.
--	--

PLAIN_COLUMN_SEPARATOR (COLUMN_SEPARATOR)	<p>Sequence of characters that is used to separate column values in an exported file. By default, a comma is used (",").</p> <p>There are several predefined values for this parameter:</p> <ul style="list-style-type: none"><li>• "\ t" and "&lt;&lt; tab &gt;&gt;" make it possible to create an exported file that uses tabs as separators.</li><li>• "&lt;&lt;space&gt;&gt;" makes it possible to use spaces as separators.</li></ul>
PLAIN_FILE_HEADER (FILE_HEADER)	<p>Adds ("Y") or excludes ("N") a file header. By default, a file header is added.</p>
PLAIN_QUOTE_STRINGS (QUOTE_STRINGS)	<p>Enables ("Y") or disables ("N") the mode for quoting strings in an exported file (addition of quotes at the beginning and end of a string). By default, quoting is enabled.</p>
PLAIN_LINE_SEPARATOR (LINE_SEPARATOR)	<p>Sequence of characters that is used to separate lines. Possible values: "CRLF", "LF" and "native". "native" is the default value (determined by the platform that is used).</p>

## Chapter 3. CSV file format

By default, output file format is CSV with headers (see RFC 4180).

For example:

```
COLUMN_1,COLUMN_2,COLUMN_3  
1,, "text value"
```

File format can be changed using the pipe parameters PLAIN\_COLUMN\_SEPARATOR, PLAIN\_FILE\_HEADER, PLAIN\_QUOTE\_STRINGS.

## Chapter 4. Example of export in XLSX format

For example, the following SQL query is used to export data in XLSX format:

```
select 'due_contract_' || item_due.due_contract || '.xlsx' AS file_name,  
       'account_' || item_due.due_account AS sheet_name,  
       item.id, item.date_from, item.date_to, item.item_total,  
       item.fee_total, item.item_total + item.fee_total item_amount,  
       item_due.due_amount  
from   item, item_due  
where  item.id = item_due.id and item_due.due_status = 'C'  
and item_due.due_date > sysdate-90  
and item_due.due_amount != (item.item_total + item.fee_total)
```

As a result of the pipe's operation, files will be created with the names `due_contract_<item_due.due_contract>`, each of which will contain the tabs `account_<item_due.due_account>`.