# Open Orders Export Manual

# 

## Short process description

This automated process analyzes all current open customer orders from the 'CustomerOrderLinesExport1.csv' file. It enriches this data with corresponding IDs from the 'CustomerOrdersExport1.csv' file and then separates the orders for each individual client. Subsequently, the system uses the 'Email.xlsx' configuration to send personalized email notifications to clients who have pending open orders.

## How to run the automation

1. Prepare both input files and place them into the input folder.
2. Maintain the Emails.xlsx file with the list of clients which need to receive a notification
3. Double-click Cleaner script to prepare the environment for the main run.
4. Double-click OpenOrdersSendNotif script
5. Wait for the pop-up with the information that process ended
6. Check Outlook email if all emails were sent

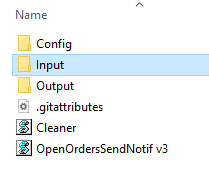
Process successful run message:

## 

## Requirements

### 

### Input files



#### CustomerOrderLinesExport1.csv

The primary source of data for this process is a CSV text file containing all active open order information. This file serves as the foundation from which all client-specific order files are generated.

File needs to be imported into Input folder as a .CSV with TAB separated data without double quoting (“data”)

#### CustomerOrdersExport1.csv

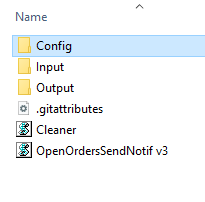
A secondary data source provides Order IDs and corresponding Customer Purchase Order (Cust PO) numbers. The automation reads this file and uses it to map these values to the data in the main open orders file.

As the main file, this too needs to be imported as .CSV with TAB separator.

### 

### 

### Config files



**Config** folder contains configuration files Emails.xlsx and SubjectText.txt where the process picks up client report emails and body of the email that will be sent to a client. Name of the file is case specific to “Emails.xlsx” and “SubjectText.txt”.

**Emails.xlsx**

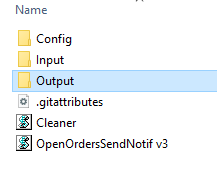
Is a list of all the clients where automation will look for emails of the clients that have open orders. File contains 2 columns, Name and Value, where Name contains names of the clients written exactly the same as in ‘'CustomerOrderLinesExport1’ file. If a client with an open order is found in the Emails.xlsx file, a notification will be sent to an email from the “Value” column.

If a client has an open order but email is not listed in the Emails.xlsx file, notification will be skipped and the process will continue to the next client until finish. Report file will contain information about all sent or skipped clients.

**SubjectText.txt**

Is a configurable text file allowing for modification of the email body content that clients will receive in their notifications. The system reads this file and dynamically constructs a personalized email body for each client's open order notification.

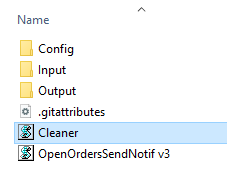
### Output



Automation is using Output to place temporary .csv files with the list of open orders splitted by the client's name. Also, once all clients are splitted automation uses Output folder to construct .xlsx files that will be attached to the notification email.

Before each run, automation cleans this folder to avoid duplicates.

### Cleaner script

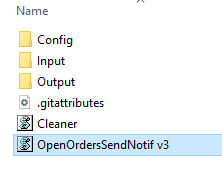


In case a main process fails while running, double click this script first before the next main run. This process will close all open excel or script instances (and files if you have any excels open) and clear the settings for the main process.

Good practice case:

Run this script before each run to clear all excel and script instances.

### OpenOrdersSendNotifscript



A main process script. In short, this script combines data from input files to create a list of open orders split by the client's name, and send the files in email as a notification to the clients. Script is dependent on the Input and Config files with case specific names to read the data and construct the report files.

In case of a failure, either files are missing or naming is not correct, or data in the file is not as expected. When an event like this happens, there will be a pop up with error or information.