

Example “Batch processing”

This example demonstrates the use of iterations to process a list of files in EasyMorph. To understand better how iterations work please read this article: [Tutorial – Loops and iterations](#).

In this example there are 3 text files in CSV format. Each file contains a history of prices for a stock exchange ticket. The example uses iterations to arrange a loop in which each file is filtered and only records with dates later than a particular date are kept. The result is exported into a new CSV file.

The loop (iteration) is arranged by two projects – **Main.morph** and **trim csv.morph**. Here is what **Iterations.morph** does:

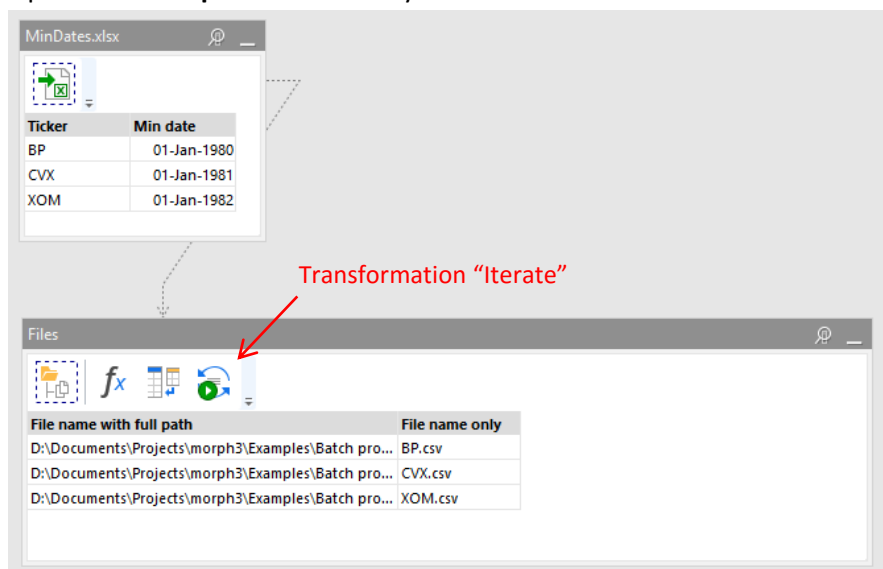
1. Generate a list of files in specified folder.
2. Load a list of min dates for each ticker from a spreadsheet.
3. Merge both lists to create a table with file names and a date for each file.
4. Iterate (run) **trim csv.morph** passing file names and dates through its parameters {File name} and {Min.Date}.

Here is what **trim csv.morph** does:

1. Load a CSV file specified by a parameter {File name}.
2. Convert text dates into number dates.
3. Filter rows, keeping only dates greater than parameter {Min.date}
4. Convert number dates back into text.
5. Export result into a file, defined by a calculated parameter {Output file name}. This parameter is calculated as {File name} with modified path, where the folder name is changed from “Source files” to “Output”.

How to use the example

- Open **Main.morph**. Here is what you should see:



- Note that “Iterate” transformation is not calculated, because it iterates a project with a *side-effect* transformations, i.e. a transformation that changes external files.
- Notice that Output folder in the example is empty
- Run iteration, by pressing Run Project button, or simply F5.
- Notice that Output folder now contains 3 files, just generated by the project.
- Click “Iterate” transformation and explore its properties:

Iterate
Iterate through table values passing them to another EasyMorph project as parameters. [More](#)

For each file run another project that will perform filtering and exporting to Output folder.

[Edit](#)

Project to run: .. trim csv.morph

Iteration mode

☒ Iterate

☐ Iterate and append results

Assign parameters of the project using these columns:

Parameter in trim csv		Column
File name	=	File name with full p
Min.Date	=	Min date

Click to open the iterated project, if needed

Iteration mode

Parameters in the iterated project

Columns in the iterating (i.e. current) project

Notice that iteration mode has two options: “Iterate” and “Iterate and append results”:

- In the first mode EasyMorph runs iterated project (here its **trim csv.morph**), but doesn’t return any table from it, because it’s not necessary – the iterated project performs export itself. Therefore output of “Iterate” transformation remains the same as its input, i.e. stays unchanged.
- If the second option was selected (and a result table name specified), EasyMorph would have returned the result table from **trim csv.morph** to **Main.morph** in each iteration, and appended them into one table that became the output of “Iterate” transformation. This mode is used, for instance, for loading multiple uniform files and concatenating them into one table.