IMPORT_CSV

IMPORT_CSV action

Function

The action allows importing of a specified column, or whole structure from the given CSV file.

Declaration

```
 IMPORT_CSV destStruct, delimiter, fName, timeZone, retCodeIdent_Int[,
timeMaskIdent [, lineFrom, lineTo]][TIME] [UTF8 | ENCODING
"@APP_DEFAULT@"] [COLMAP _colMap]
```

or

```
 IMPORT_CSV destStruct, delimiter, handle, timeZone, retCodeIdent_Int,
[timeMaskIdent], numLines [TIME] [COLMAP _colMap]
```

Parameters

destStruct	in	Column identifier or identifier of a local variable of the *Record* type.
delimiter	in	Identifier of the *Text* type - delimiter used in a CSV file.
fName	in	File name with extension.
timeZone	in	Parameter of the *Text, Int or Bool* type - defines interpretation of timestamps.
timeMaskIdent	in	Identifier of the *Text* type - mask for reading a value of the *Absolute time* type.
retCodeIdent	out	Return code of the *Int* type - action success.
lineFrom	in	Parameter of the *Int* type - begin line.
lineTo	in	Parameter of the *Int* type - end line.
handle	in	Parameter of the *Int* type - value returned by the function FIO_OpenRead.
numLines	in	Parameter of *Int* type - maximum number of rows to read.
TIME	in	Keyword.
UTF8	in	Keyword.
ENCODING	in	Keyword which is followed by a name of file encoding in the text format. The list of supported encodings in ESL.
COLMAP	in	Keyword which is followed by an identifier of the TEXT type - **_colMap.**
_colMap	in	Identifier of the TEXT type. It includes the column names of structure, into which data from csv file will be inserted.

Return code

The value of the parameter *transHandle_Int*. See the table of error codes.

Description

The action reads the contents of a CSV file. File name is given by a value of the parameter *fName*. Item delimiter in a CSV file is given by a value of the parameter *delimiter*. The file can use the line-spacing by **LF** (Unix format), **CR** (Mac format) or **CR LF** sign (Dos format).
The action success is indicated by the output parameter *retCodeIdent_Int*. The value of 0 means the successful import, value other than 0 means an error. Depending on the parameter *destStruct*, the action imports following:

- a whole structure - the parameter is a local structure identifier
- a structure column - the parameter is the reference to a structure column

The CSV file format is described in the action EXPORT_CSV.
If the identifier *timeMaskIdent* is not defined, the function expects the *"dd.mm.yyyy hh:mi:ss"* format for items containing an absolute time. If the parameter is defined, the format of the time is to be imported according to that mask. In case that the *hh*, *mi* or *ss* time components are missing, they will be automatically replaced with 0.
When real numbers are imported the separator of decimal places may be both "." (dot) and "," (comma) symbols.
When integer and real numbers are imported from CSV file the thousand separator is a blank space.
If the value of the parameter *timeMaskIdent* is "" (empty mask), values of the *Absolute time* and *Real* type are to be converted according to the settings in the **Regional and Language Options** of the current user on Windows systems. For different operating systems the empty mask is automatically replaced by the value of *"dd.mm.rrrr hh:mi:ss"*.
When the item is converted to Absolute time and is an empty string, the action interprets such item as invalid value.

If the keyword **TIME** is used, there is the assumption, that the occurrence times of values are saved in the CSV file (the double number of columns in this CSV file).
If the keyword **UTF8** is used the action read the text file and supposes that it is encoded in UTF-8. From that reason the action converts the file from UTF-8 -> WIN1250 at its reading.

In a case of need, the local variable `destStruct` will be resized. Its size is given by the number of values in the particular CSV file.

The parameter *timeZone* defines interpretation of timestamps. In case of empty text, timestamps are interpreted as local times. In case of defined time zone name (e.g. "Europe/London"), timestamps are interpreted as local times for given time zone. In case of integer value, timestamps are interpreted as times with fixed offset from UTC, where given value defines offset in seconds. For historical reasons boolean values are accepted: @FALSE - timestamps are interpreted as local times, @TRUE - timestamps are interpreted as times with fixed offset of 3600 seconds (1 hour) from UTC. Usage of boolean value @TRUE is not recommended and generates warning when ESL script is being saved. It is recommended to replace it with integer value.

If the parameters *lineFrom* and *lineTo* are defined, there will be imported all the lines from the line defined by *LineFrom* up to the line defined by the parameter *lineTo*. If the value of both the parameters is -1, there will be imported the whole CSV file.

The second variant of the action reads the next rows (number is defined by the parameter *numLines*) from the given CSV file. The file must be opened by the function FIO_OpenRead.
It is useful to use this variant for large CSV files - we recommend you not to read them at the same time.

The CSV delimiter (the parameter *delimiter*) may be obtained by calling the function %GetCSVDelimiter.

The parameter *_colMap* enables to define the columns of destination structure and the order of inserting data from the imported file. The names of columns must be separated by a delimiter. If some columns form .csv file are ignored, they must be defined in the parameter *_colMap* with the help of delimiter.

If the column names in *_colMap* differ from the structure definition, script declares an error "Invalid name of Cols in ColMap: bad column names".
If parameter *_colMap* contains a zero-length string, script declares the error "ColMap is empty".
If .csv file contains less columns than is required, the missing columns are imported as invalid values!

Example of use:

- **without ignoring columns**
  SD.Data consists of these columns - **DATE**, **VALUE**, **NR**

  *.CSV contains this header - **NR**, **DATE**, **VALUE**, **DESCRIPT**, **ID**

  _colMap may contain:
  - **NR;DATE;VALUE**
  - **NR;DATE**
  - **NR**
- **with ignored columns**
  SD.Data consists of these columns - **ID**, **VALUE**, **NR**

  *.CSV contains this header - **NR**, **DATE**, **VALUE**, **DESCRIPT**, **ID**

  _colMap may contain:
  - **NR;;VALUE;;ID**
  - **NR;;;;ID**
  - **;;VALUE;;ID**

Example

To omit the parameter *timeMaskIdent*, declare as follows:

```
IMPORT_CSV destStruct, delimiter, fName, bMonoTimes, retCodeIdent_Int, ,
lineFrom, lineTo
```

Example 2

Reading of the entire CSV file in parts of 1000 rows:

```
RECORD  NOALIAS (SD.AZZD_Imp_Day) _Import_H
 INT _retCode
 INT _handle

 _handle := %FIO_OpenRead("c:\Application\Import\LP_0407_0408180810.csv")

 DO_LOOP
 IMPORT_CSV _Import_H, ";", _handle, @FALSE, _retCode, "dd.mm.rr", 1000
 ; processing read data
 EXIT_LOOP _Import_H\DIM < 1000
 END_LOOP

 _bOk := %FIO_Close(_handle)
```

Related pages:

Script actions