

EXPORT_CSV

EXPORT_CSV action

Function

Export of a row, column or whole [structured local variable](#) (i.e. local variable of *Record* type) into a CSV file.

Declaration

```
EXPORT_CSV toExportStruct, delimiter, fName, timeZone, retCodeIdent_Int[,  
transPalIdx_Int] [TIME] [UTF8] | ENCODING "@APP_DEFAULT@" [APPEND]
```

Parameters

toExportStruct	in	Row identifier , column identifier or identifier of local variable of <i>Record</i> type.
delimiter	in	Identifier of <i>Text</i> type - delimiter to be used in the CSV file.
fName	in	File name with extension.
timeZone	in	Parameter of <i>Text</i> , <i>Int</i> or <i>Bool</i> type - defines interpretation of timestamps.
retCodeIdent_Int	output	Return value of <i>Int</i> type - action success.
transPalIdx	in	Optional parameter of <i>Int</i> type.
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 .
APPEND	in	Keyword.

Description

The action generates a file of CSV format. File name is given by the parameter *fName*. Item delimiter in the CSV file is specified by the parameter *delimiter*. The action success is indicated by the output parameter *retCodeIdent_Int*. The value of 0 means a successful export, values other than 0 are an error.

Depending on the parameter *toExportStruct*, the action exports:

- the entire local variable
- the row of local variable
- the column of local variable

CSV file format

The local variable is saved to the specified CSV file row-by-row (row in a structure presents a row in a CSV file).

Because the CSV file format is not unique defined, the action awaits a delimiter as its second parameter. Delimiter is one character, that is to be used to separate items in the CSV file. Real numbers use the decimal separator '.'.

If the keyword

```
TIME
```

is used, then the CSV file also contains the value time for each item. Time is presented before the item value. CSV file therefore contains the double number of columns.

If the keyword **UTF8** is used, the action generates a text file with **UTF-8** encoding.

If the keyword **APPEND** is used, the action inserts data into CSV file. In case that this file does not exist, it will be created.

Time can be formatted by the mask defined in the [transformation palette](#) - the index is given by parameter *transPalIdx_Int*.

If the parameter is not defined or its value is 0, there is used the mask *dd.mm.rrrr hh:mi:ss*.

If its value is -1, then values of *Absolute time* and *Real* types are to be converted according to the settings specified in the **Regional and Language Options** of the current user on Windows systems. For different operating systems the value of -1 will be automatically replaced with the value of 0.

The parameter *timeZone* defines export interpretation of timestamps. In case of empty text, timestamps are exported as local times. In case of defined time zone name (e.g. "Europe/London"), timestamps are exported as local times for given time zone. In case of integer value, timestamps are exported as times with fixed offset from UTC, where given value defines offset in seconds. For historical reasons boolean values are accepted: @FALSE - timestamps are exported as local times, @TRUE - timestamps are exported 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.

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

Example

Assumes:
The object SD.ArchVal of *Structure definition* is defined with the columns:

Name	Type
Value	Real
Flags	Integer

There is defined the object SV.ArchVal of *Structured variable*. The structure type is SD.ArchVal. The number of rows: 3. Start values are:

Value	Flags
1.2	3
2	2
3	1

The example does not test the action execution success.

```
RECORD (SD.ArchVal) _rec

REDIM _rec[SV.ArchVal\DIM]
SET _rec WITH SV.ArchVal

EXPORT_CSV _rec, ';', "c:\rec.csv", @TRUE, _retCode
EXPORT_CSV _rec[1], ';', "c:\rec1.csv", @TRUE, _retCode
EXPORT_CSV _rec^Value, ';', "c:\recValue.csv", @TRUE, _retCode
```

The actions exports the local variable *_rec*, that contains a copy of *SV.ArchVal*.
Created files:

c:\rec.csv

```
1,2000000000000000;3
2,0000000000000000;2
3,0000000000000000;1
```

c:\rec1.csv

```
1,2000000000000000;3
```

c:\recValue.csv

```
1,2000000000000000
2,0000000000000000
3,0000000000000000
```



Related pages:

[Script actions](#)