



SOFTWARESILOS

DATEV Export Pro Extension

PDF Manual

MODULE

datev

LAST UPDATED

2026-03-12

Mapping Language

DATEV Export Pro Extension · /datev/features/mapping-language

The DATEV exporter evaluates mapping expressions with an enhanced processor. This page documents the supported syntax for **CSV**, **XML**, and **master data** mappings.

Context Objects

- **Invoice exports:** `invoice.*` is available (root object)
- **Credit memo exports:** `creditmemo.*` is available (root object)
- **XML exports:** `invoice.*` / `creditmemo.*` are available (root object)
- **Master data exports:** `customer.*` and `address.*` are available

Variable Paths

Use dot-notation paths to access data:

- `invoice.increment_id`
- `invoice.created_at`
- `invoice.order.increment_id`
- `invoice.order.customer_firstname`
- `invoice.order.billing_address.country_id`

The processor also supports `order.customer` by loading the customer entity via `customer_id` when needed.

String Literals

Use single quotes for constants:

- `'EUR'`
- `'S'`
- `''` (empty string)

Templates

You can wrap a path in `{{...}}` or embed multiple variables in one string:

- `{{invoice.increment_id}}`
- `Invoice {{invoice.increment_id}} (Order {{invoice.order.increment_id}})`

Concatenation

Use `+` for string concatenation:

```
invoice.order.customer_firstname + ' ' + invoice.order.customer_lastname
```

Conditionals (Ternary)

Use `? :` for conditional values:

```
invoice.grand_total > 0 ? 'S' : 'H'
```

Supported comparison operators:

- `==, !=, >, <, >=, <=`

Math Expressions

For numeric-only expressions, `+ - * / ()` are supported:

```
invoice.grand_total - invoice.tax_amount
```

Built-in Functions

These functions can be used like `functionName(arg1, arg2, ...)`:

- `upper(value) / uppercase(value)`
- `lower(value) / lowercase(value)`
- `trim(value)`
- `length(value) / strlen(value)`
- `substr(value, start, length?) / substring(...)`
- `replace(subject, search, replace)`
- `round(value, precision?), floor(value), ceil(value), abs(value)`
- `default(value, fallback) / ifempty(value, fallback)`
- `concat(a, b, c, ...)`
- `number_format(number, decimals?, decPoint?, thousandsSep?)`

Examples:

```
upper(invoice.order.customer_lastname)
default(invoice.order.customer_email, 'no-email@example.com')
number_format(invoice.grand_total, 2, ',', '')
```

Function Providers

Function providers are called with the `function:` prefix:

- `function:getRevenueAccount`
- `function:getTaxRate`

You can also pass parameters: `function:myFunction('param1', invoice.increment_id)`.

See: [Function Providers](#)

Config References

You can reference Magento config values using:

```
config:general/store_information/name
```

Defaults For Missing Values

When a variable path resolves to `null`, the processor can return a default value configured in:

- **Variable Default Values → Default Values for NULL Variables (CSV)**
- **Variable Default Values → Default Values for NULL Variables (XML)**

Defaults are keyed by the variable path (e.g. `invoice.order.customer_firstname`).