



Assay Importer

A formula-based import builder that absorbs any laboratory certificate layout — and replays it on every delivery.

- Group: Sample Management
- Batch import and bundle editor
- 18 formula functions
- Reusable json configurations

Preview

Run Import

Run Multi-Config

Batch Import

Bundle Editor

Mapping Tools

KU26009199.csv

CSV Preview (loaded exactly as provided)

How to build references such as Cell(2, 1), or mark the header and data-start rows for the target table.

	C1	C2	
KU26009199	Finalized		
CLIENT	CLIENT A		
# of SAMPLES	200		
DATE RECEIVED	090126		
DATE COMPLETED	140226		
PROJECT	KDN		
CERTIFICATE COMMENTS			
PO NUMBER	PO_2026_05		
	WEI-21		Au-
SAMPLE	Recvd Wt.		Au
DESCRIPTION	kg		ppr
DETECTION	0.02		1
RCDW00332	2.24		1
RCDW00332	1.6		1
RCDW00332	1.67		1.0

Target Database Table

Table:

Mode: Table rows (one DB record)

Column Mappings (dbl-click DB Field)

DB FIELD

+ Add Row

Executive summary

The Assay Importer turns a laboratory certificate — whatever its format — into database records through reusable formula chains: a configuration written once serves every subsequent delivery from the same laboratory.

01 What it replaces

Cleaning the certificate in Excel before every import, with its hand-moved columns and its invisible errors.

02 How it works

Three panels: the CSV structure, the data preview and the mapping rules. Every rule is a formula; the preview validates it against real rows before anything is written.

03 Who it is for

The database owner, who writes the configuration once — then anyone in the team, who loads it and runs the import.

11 740

assays imported on the project

17

transformation functions

2

multi-element mapping generators

200

preview rows before import

Contents

01 The certificate problem

No two laboratories write their header the same way.

02 Layout interpretation

Header row and data-start row, designated with the mouse.

03 The formula language

Eighteen functions to extract, convert, condition and concatenate.

04 Header generators

Automatic composition of multi-element column names.

05 Multi-table bundles

One file, several target tables, with cross-references.

06 Broken formula detection

A formula that no longer resolves is flagged before the import.

07 Anonymised data

11,740 assays imported, across two distinct analytical suites.

08 Business value

A configuration written once, replayed on every delivery.



GeoMate 2026

SECTION I

The module

The problem addressed, the capabilities delivered, the processing chain and the module's place in the platform.

-
- 01 Context
 - 02 Key capabilities
 - 03 Processing chain
 - 04 Place in the platform
-

Context

A laboratory certificate does not carry column names a database can use: it spreads the information over the lines around the header — the analysis method above, the unit below — and repeats that across forty to seventy element columns.

The Assay Importer does not ask the laboratory to change: it describes the transformation once, saves it as a configuration, and replays it on every subsequent delivery.

The header row is designated

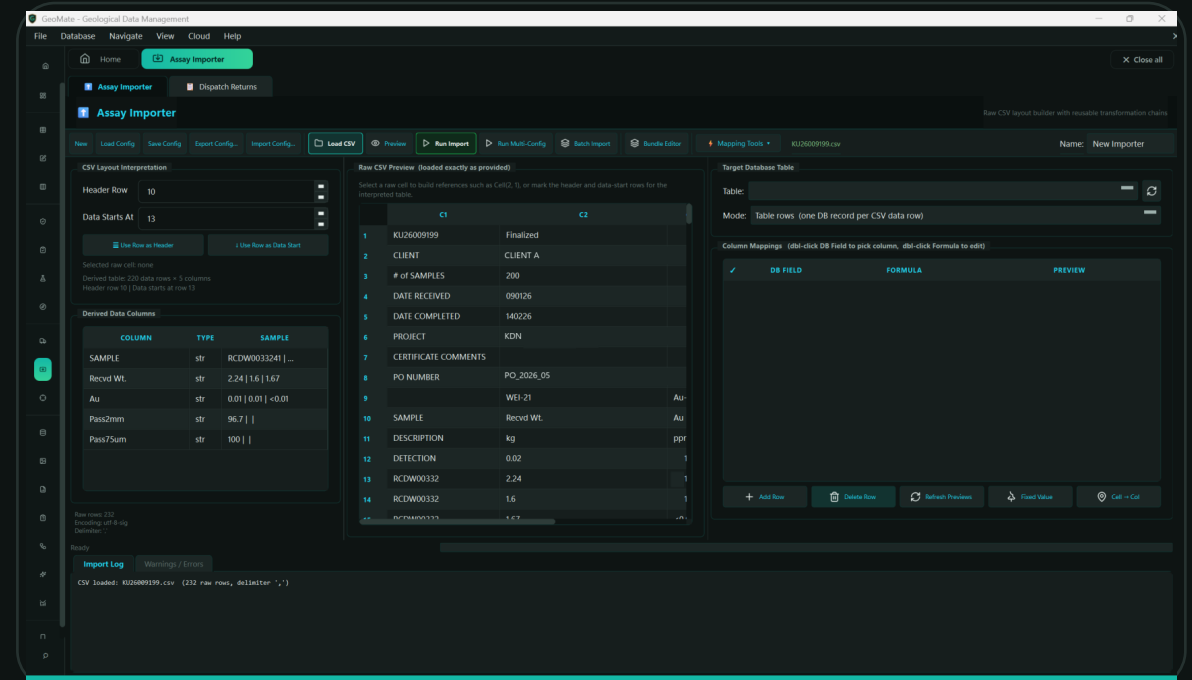
Header row and data-start row are picked with the mouse; the derived table appears immediately on the left.

A clicked cell becomes a reference

Clicking a cell in the preview inserts its Column() reference into the formula being edited.

A broken formula is flagged

A formula naming a missing column writes NULL without failing. The editor lists those formulas after every generation and on every new file.



Read the list of unresolved formulas before running the import: it is the only warning you will get before whole columns are written as NULL.

Key capabilities



01

Layout interpretation

Configurable header row and data-start row; the derived table and detected types appear at once.



02

Eighteen functions

Column, Cell, CellByLabel, FindRow, RowNumber, If, IfNull, Upper, Lower, Trim, Left, Right, Mid, ReplaceText, Concat, Value, ToDate, Round, TabColumn.



03

Header composition

Assembles the field name out of several raw header rows: Ag + ME-ICP41 + ppm becomes Ag_ME-ICP41_ppm, with an editable preview table.



04

Element generation

Picks out the element columns, lines them up with those the target table already has, and adds the missing ones on request — without exceeding the column ceiling.



05

Bundle editor

One file feeds several target tables; TabColumn lets a tab reuse a sibling tab's computed value.



06

Batch import

Several CSVs in one run, or several configurations applied to the same file, with a detailed log.

Processing chain

From the raw certificate to written records, with a checkpoint before writing.

A Inputs

Certificate and configuration

- Laboratory CSV file, headers spread over several rows
- A .json mapping configuration, where one already exists
- Target table and its existing columns
- Assay method and unit carried by the certificate
- Several files, for a batch import

B Mapping & check

Formulas, preview, warnings

- Compose Headers: assembles Ag + ME-ICP41 + ppm into one field name
- Generate Elements: lines the element columns up with the target table
- Formulas: Column, Concat, If, ToDate, Round, TabColumn...
- Preview over the first 200 rows, error per mapping row
- List of formulas that no longer resolve — the only warning

C Outputs

Populated database and reusable configuration

- Records written into the target table or tables
- Import log: progress, status, warnings
- A .json configuration saved for the next delivery
- A multi-table bundle for a certificate feeding several tables
- Assays immediately available to QA/QC and to analytics

THE TRAP TO KNOW ABOUT

A formula naming a column the CSV does not have does not fail: it writes NULL and the import reports success. After every generation, and every time a saved configuration meets a new file, the editor lists the formulas that no longer resolve. That is the only warning, and it must be read before running.

Clicking a cell in the preview automatically inserts its `Column()` reference into the formula being edited.

Place in the platform

FED BY

09 Dispatch Manager

Supplies the dispatch identifier that reconciles the certificate with the submission.

— Laboratory certificate

The source file, with its multi-row headers and its forty to seventy element columns.

02 Table Designer

Declares the target columns; the element generator can add the missing ones.



Assay Importer

The gateway for laboratory results: everything analysed afterwards comes through a rule written here.

FEEDS

06 QAQC Manager

Receives the certificate's standards, blanks and duplicates to score them.

07 Geochemical Analytics

Receives the multi-element columns, correctly named and united.

11 Assay Intercept

Receives the per-interval grades to composite.



GeoMate 2026

SECTION II

In practice

The real interface, the operating workflow, the reference settings, good practice and the questions that come up.

-
- 01 The interface
 - 02 Operating workflow
 - 03 In depth
 - 04 Good practice
 - 05 Frequently asked
-

The interface

FileDatabaseNavigateViewCloudHelp

HomeAssay ImporterClose all

Assay importerDispatch Returns

Assay ImporterRaw CSV layout builder with reusable transformation chains

NewLoad ConfigSave ConfigExport Config...Import Config...Load CSVPreviewRun ImportRun Multi-ConfigBatch ImportBundle EditorMapping ToolsKU26009199.csvName: New Importer

CSV Layout Interpretation

Header Row10Data Starts At13Use Row as HeaderUse Row as Data StartSelected raw cell: noneDerived table: 220 data rows × 5 columnsHeader row 10 | Data starts at row 13

Derived Data Columns

COLUMN	TYPE	SAMPLE
SAMPLE	str	RCDW0033241 ...
Recvd Wt.	str	2.24 1.6 1.67
Au	str	0.01 0.01 <0.01
Pass2mm	str	96.7
Pass75um	str	100

Raw rows: 232Encoding: utf-8-sigDelimiter: ,

Raw CSV Preview (loaded exactly as provided)

Select a raw cell to build references such as Cell(2, 1), or mark the header and data-start rows for the interpreted table.

	C1	C2
1	KU26009199	Finalized
2	CLIENT	CLIENT A
3	# OF SAMPLES	200
4	DATE RECEIVED	090126
5	DATE COMPLETED	140226
6	PROJECT	KDN
7	CERTIFICATE COMMENTS	
8	PO NUMBER	PO_2026_05
9		WEI-21
10	SAMPLE	Recvd Wt.
11	DESCRIPTION	kg
12	DETECTION	0.02
13	RCDW00332	2.24
14	RCDW00332	1.6
15	RCDW00332	1.67

Target Database TableTableMode: Table rows (one DB record per CSV data row)

Column Mappings (dbcli-click DB Field to pick column, dbcli-click Formula to edit)

DB FIELD	FORMULA	PREVIEW
----------	---------	---------

+ Add RowDelete RowRefresh PreviewsFixed ValueCell → Col

ReadyImport LogWarnings / Errors

CSV loaded: KU26009199.csv (232 raw rows, delimiter ',')

- 1 Layout interpretation**

Header row 10, data from row 13 — 220 derived data rows.
- 2 Derived columns**

Name, detected type and sample values for each reconstructed column.
- 3 Raw preview**

The CSV exactly as delivered; clicking a cell inserts its reference.
- 4 Target table and mode**

Destination table and write mode: one database row per CSV row.
- 5 Column mappings**

Database field, formula and preview of the produced value, row by row.

Operating workflow

Five steps the first time; two clicks on every subsequent delivery.

1

Load the CSV

The structure appears on the left, the raw preview in the centre, exactly as delivered.

2

Interpret the layout

Designate the header row and the data-start row; the derived table is built.

3

Build the mapping

One row per target field; the header and element generators write the multi-element mappings.

4

Preview

Validate the formulas against real rows and read the list of unresolved formulas.

5

Save and import

The configuration is saved as .json and replayed; the import writes to the database with a log and warnings.

The formula language

FUNCTION	DESCRIPTION	EXAMPLE
Column("name")	Value from a CSV column by name	Column("SampleID")
RowNumber()	Current import row number, 1-based	RowNumber()
Cell(row, col)	Value of a cell by position	Cell(2, 1)
CellByLabel("label", col)	Value below a cell carrying that label	CellByLabel("Project", 1)
FindRow("text", col)	Row index of the first matching cell	FindRow("HOLE", 1)
If(cond, true, false)	Conditional expression	If(Column("Type")="STD", "Standard", "Sample")
IfNull(val, default)	Fallback when null or empty	IfNull(Column("RL"), 0)
Concat(a, b, ...)	String concatenation	Concat(Column("Area"), "-", Column("Num"))
Value(str) · Round(v, n)	Numeric conversion and rounding	Round(Value(Column("Grade")), 3)
ToDate(str, "DD/MM/YYYY")	Parse a date against a format	ToDate(Column("Date"), "DD/MM/YYYY")
TabColumn("tab", "field")	Reuses a sibling tab's computed value for the same source row; circular references are detected	Concat(TabColumn("Collar", "HoleID"), "-", Column("From"))

Good practice

The procedure for importing a certificate, from file to log.

✓ Read the resolution warning

The list of formulas that no longer resolve is the only signal that a certificate has changed format. A "successful" import full of NULLs is worse than a failed one.

✓ Always preview

The preview validates each formula on real rows and gives the error per row. It is the checkpoint before writing.

✓ Respect column-name casing

Column("...") is case-sensitive and must match the CSV header exactly. Column, not column.

✓ Compose the names rather than typing them

Compose Headers assembles the field name from several header rows: safer and reproducible from one certificate to the next.

✓ Save the configuration

The .json makes the next import immediate, and makes the mapping legible to someone other than its author.

✓ Use a bundle for a multi-table certificate

A file that feeds collar, sample and assay belongs in the bundle editor: TabColumn there reuses another tab's computed value.

Frequently asked

The most common import errors and questions.

Q The formula preview returns an error

Check the spelling and casing of the function names — Column, not column — and that the quoted column name exists exactly in the CSV header. The preview details the error per mapping row.

Q The import succeeds but the columns are empty

A formula points at a missing column: it writes NULL without failing. That is precisely what the unresolved-formula list flags.

Q How do I handle forty element columns?

Generate Elements picks them, lines them up with the table's existing columns and can create the missing ones — within the database's own column ceiling.

Q Can several files be imported at once?

Yes: Batch Import processes several CSVs in one operation, and Run Multi-Config applies several configurations to the same file.

Q What is TabColumn for?

To reuse the value computed by another bundle tab for the same source row — the HoleID composed on the collar side, for instance. Circular references are detected.

Q Is the configuration portable?

Yes: it is a .json file that loads on another project or another workstation, as long as the certificate format is the same.



GeoMate 2026

SECTION III

The data

What the module produces on a real dataset, and the vocabulary needed to read those results.

01 Anonymised data

02 Key concepts

Anonymised data

The real result of certificate import on the KOUKOUDAN database.

11 740

ASSAYS IMPORTED

from the laboratory certificates

2

ANALYTICAL SUITES

Au_AA26_ppm (11 165) ·
Au_FAOG_AAS_ppm (575)

54

ELEMENT COLUMNS

recognised out of 66 by the element
generator

18

FORMULA FUNCTIONS

including TabColumn for cross-
references

The two suites — Au_AA26_ppm and Au_FAOG_AAS_ppm — come from different certificate templates and are nonetheless written into the same target table: that is precisely the work the import configuration does, once written and then replayed.

Source: KOUKOUDAN project database (Senegal) — figures computed from the real data exported from GeoMate.

Key concepts

The vocabulary of import mapping.

Transformation chain

The sequence of formulas converting a CSV row into a database record. It is saved and replayed.

Analysis suite

The package of methods applied to the sample — ME-ICP41, for instance. It governs the detection limits.

Upsert

Write a record if absent, update it if present. The mode to choose for a corrective delivery.

Certificate

The laboratory's analytical document: identifiers, methods, units and results, often on multi-row headers.

Bundle

A multi-table import: one file feeds several target tables, each in its own tab, with shared values.

Unresolved formula

A formula whose source column has vanished from the file. It writes NULL silently — hence the dedicated warning.



GeoMate 2026

SECTION IV

Value & specifications

What the module changes for the project, and its technical integration profile.

01 Business value

02 Technical profile

Business value

The laboratory format stops being a problem

Every laboratory keeps its template; it is the import configuration that adapts, once and for all.

.json configuration

Seventy columns map in one click

The header-composition and element generators write the multi-element mappings nobody wants to type by hand.

2 generators

One file feeds several tables

The bundle editor writes collars, samples and assays from the same certificate, with cross-references.

Multi-table bundle

Silent columns are called out

A formula that would silently write NULL is listed before the import — the hardest defect to spot afterwards.

Explicit warning

The weekly delivery becomes routine

Load, preview, import: the operation drops from half a day to a quarter of an hour.

Batch import

Technical profile

Inputs

Formats

CSV, delimiter and encoding detected

Preview

First 200 rows, raw

Layout

Configurable header and data-start rows

Volume

Batch import of several files

Transformation

Functions

18, including TabColumn for cross-references

Generators

Header composition · element generation

Bundle

Several target tables from one file

Checking

List of unresolved formulas before import

Outputs

Write

One database row per CSV row

Log

Progress, warnings and errors

Configuration

A .json file, loadable and exportable

Automation

Triggerable from the Automation Builder



One platform, from field to report

The “Assay Importer” module is one of the 19 modules of GeoMate 2026 — a single, fully offline platform covering the whole chain: database design, data entry, validation, QA/QC, dispatch, analysis and reporting.