

Report and form

Language: [Deutsch](#) · [English](#)

Report and form

Data sources as the basis of reports and forms – parameters, intid, wildcards and the SQL editor.

REFERENZ APPLICATION DEVELOPER UPDATED: JUL 2026 APPLIES TO NUCLOS 4.2026.X

On this page

- Data source = basis of every report/form
- Structure of the editor
- The special intid parameter
- Text parameters & wildcards
- Internal parameter types
- Conditions & SQL
- Related pages

Data source = basis of every report/form

Data sources are the **basis for reports and forms**. A data source defines which business objects are considered; conditions and joins restrict the data or add attributes. **Parameters** make a saved data source reusable.

Menu: Configuration Data sources Report and form

Structure of the editor

The screen has five areas:

Area	Content
Header	name and description of the data source.
Table selection	system objects (master/transaction data), dynamic objects (BO tables) and saved queries (as sub-queries).
Table view	business objects dragged in via drag&drop.
Parameter definition	name, data type and prompt per parameter. Forms require at least intid.
Column view	visible columns (alias = later field name), sorting and conditions. After saving: <i>SQL</i> and <i>preview</i> .

Modell SQL Vorschau

Statement wieder herstellen

Parameter...	Datentyp	Pflichtpara...	Anzeigename	Valuelist Provider
intid	Long	☐		Kein ValueList Provider defini... ...
logo	Text	☐	Bitte wählen Sie, o...	ReportLogo.ds ...

Parameter definition of the data source.

The special intid parameter

Data sources for forms must define the `intid` parameter for the `intid` column of the BO to print. Nuclos fills it automatically when printing – without a dialog. A [value list provider](#) provides a picklist; the text value via `'$name'`, the ID via `'$nameId'`.

T1.j94r_auftrag

intid

auftragauftragsnum

auftragbemerkung

auftragbezdatum

auftragbezeichnung

1.0 auftragbezsumme

auftragdatum

auftragegausland

1.0 auftraggutsumme

auftragkundenname

1.0 auftragmahngebuehr

1.0 auftragnummer

Tabelle	T1.j94r_auftrag	T1.j94r_auftrag	T1.j94r_auftrag
Spalte	intid	auftragauftragsnummer	auftragbemerkung
Alias	intid	auftragauftragsnummer	auftragbemerkung
Sichtbar	☒	☒	☒
Gruppierung			
Sortierung			
Bedingung	= \$intid		
oder			

intid parameter in the form.

Text parameters & wildcards

Condition	Effect
= '\$param'	no wildcards; „=" comparison; allows searching for % and _.
WIE '\$param'	wildcards allowed; LIKE condition.

=\$param%	always LIKE with appended wildcard; empty value = all records.
-----------	--

Internal parameter types

Text list, integer list, SQL are not offered to the user but used internally in rules. SQL parameters require a VLP as a whitelist of allowed expressions.

```
SELECT
    T1.INTID "<primaryKey>",
    T1.INTARTICLENUMBER "articleNumber",
    T1.STRNAME "name"
FROM
    V594__ARTICLE T1
WHERE
    $pSql
    OR
    T1.INTARTICLENUMBER in $pListOfInteger
    OR
    T1.STRNAME in $pListOfText
```

```
List<Number> listOfInteger = new ArrayList<>();
listOfInteger.add(4711L); // Test with Long
List<String> listOfStrings = new ArrayList<>();
listOfStrings.add("4711");

Map<String, Object> params = new HashMap<>();
params.put("pListOfInteger", listOfInteger);
params.put("pListOfText", listOfStrings);
params.put("pSql", "T1.INTARTICLENUMBER > 4000");

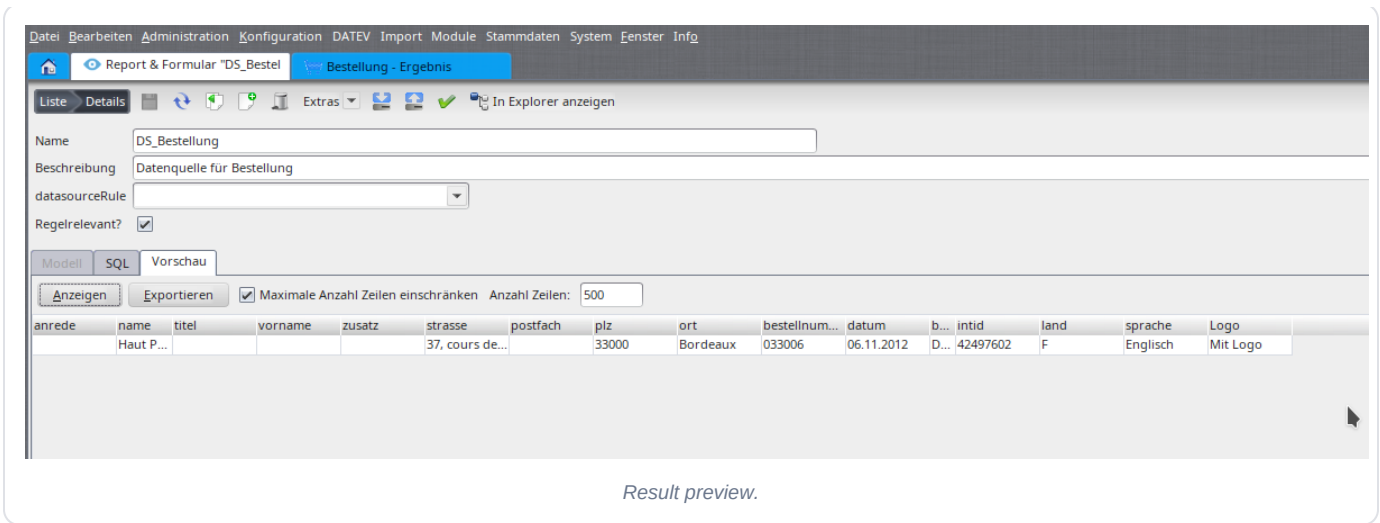
DatasourceProvider.run(TestDS.class, params);
```

Conditions & SQL

Column conditions support comparison operators (= > < >= <= != WIE), ranges (between 'a' and 'b'), brackets and column comparisons. Parameters always start with \$. If graphical modelling is not enough, activate the **SQL** tab and edit the SELECT statement directly; then test via *check syntax* and *preview* (incl. export to Excel/PDF/CSV).

The screenshot shows a software interface for editing a data source. The top menu bar includes 'Datei', 'Bearbeiten', 'Administration', 'Konfiguration', 'DATEV', 'Import', 'Module', 'Stammdaten', 'System', 'Einsteller', and 'Info'. Below the menu, there's a toolbar with icons for 'Liste', 'Details', 'Report & Formular', 'DS_Bestell', 'Extras', and 'In Explorer anzeigen'. The main area is divided into two tabs: 'SQL' (active) and 'Vorschau'. The 'SQL' tab displays a complex SQL query. The query starts with a 'SELECT' statement listing various fields from tables T2 and T3, including 'STRNAME', 'BESTELLUNGSLIEFERANTNAME', 'STRITTEL', 'STROBNAME', 'STRADRESSZUSATZ', 'STRSTRASSE', 'STRPOSTFACH', 'STRPLZ', 'STRORT', 'BESTELLUNGSNUMMER', and 'BESTELLUNGSDATUM'. It then uses a 'CASE' statement to handle different data types. The 'FROM' clause involves multiple joins: 'J94R_BESTELLUNG T2', 'LEFT JOIN t_md_user t20 ON t2.stuid_bestellungbearbeiter = t20.stuid', 'LEFT JOIN J94R_LAND t21 ON t2.intid_k_md_land = t21.intid', 'INNER JOIN J94R_LIEFERANT T1 ON T2.INTID_BESTELLUNGSLIEFERANT1 = T1.INTID', 'LEFT JOIN J94R_SPRACHE T22 ON T1.INTID_STRSPRACHE_REPORTING = T22.INTID', and 'J94R_REPORTLOGO T3'. The 'WHERE' clause filters for 'T2.INTID = \$intid' and 'T3.STRNAME = 'Logo''. The interface also shows a 'Parameters' table on the left with columns 'Parameter', 'Datentyp', and 'Pflichtparameter'. It lists 'intid' as 'Long' and 'logo' as 'Text'.

Editing the SQL statement.



Related pages

Reports

Create reports.

[Open](#)

Forms

Create forms.

[Open](#)

Data sources

Data sources in general.

[Open](#)