

Anpassung der Datenquellen

 Sprache: **Deutsch** · [English](#)

Anpassung der Datenquellen

Datenquellen des Basisformularsatzes sicher anpassen – Alias-Namen und Formularbezug beachten.

HOW-TO

ANWENDUNGSENTWICKLER

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Datenquellen des Basisformularsatzes anpassen](#)
- [Alias-Namen nicht verändern](#)
- [Zusammenhang Datenquelle Formular](#)
- [Verwandte Seiten](#)

Datenquellen des Basisformularsatzes anpassen



Wichtig

Bevor du an einer Datenquelle arbeitest: **lege eine Sicherung an!**

Mit dem Import des Basisformularsatzes wird zu jedem Formular eine Datenquelle mitgeliefert. Für Anpassungen sind SQL-Kenntnisse (je nach DBMS) hilfreich.

Alias-Namen nicht verändern

Auf der rechten Seite einer Datenquelle stehen die **Alias-Namen** der Attribute. Diese sollten **nicht** geändert werden – andernfalls muss die Änderung auch in der Formularvorlage nachgezogen werden.

Formular Mahnkonto

Formular Mahnkonto

SQL **Vorschau**

ent wieder herstellen

eter... Datentyp Pflichtpara...
\$intid

Der Parameter muss hier
nachgetragen werden
\$intid
Datentyp= Integer

```
SELECT
/*Tl.strkundennamen*/'Kundennamen'
/*adr.stradressenzeile*/'Name Zusatz'
/*adr.strpostfach*/'Postfach'
/*adr.strstrasse*/'Strasse'
/*lr.strname*/'Land'
/*adr.strort*/'Ort'
/*adr.strplz*/'PLZ'
/*(substring(ba.strvorname from 1 for 1)||substring(ba.strnachname from 1 for 1))*'TT'"stunserzeichen',
'1111/1111' null
/*ba.strvorname*/'Vorname'
/*ba.strnachname*/'Nachname'

/*FROM
t_eo_kunde Tl
left outer join t_eo_mitarbeiter ba on Tl.intkundevon::text = ba.strmitarbeiternr
left outer join t_eo_adresse adr on adr.intid_strkunde = Tl.intid
left outer join t_eo_adressart aty on aty.intid = adr.intid_stradressart
left outer join t_eo_land lr on lr.intid = adr.intid_strland

WHERE
(
Tl."intid" = $intid and
aty.strname = 'Rechnungsadresse'
)*/
```

Wert

Alias
Hier darf nichts geändert werden!

"strname",
"strkundennummer",
"strnamezusatz",
"strrpostfach",
"strstrasse",
"strstrassezusatz",
"strrland",
"strrort",
"strrplz",
"strtelefon",
"strtelefax",
"strbearbeitervn",
"strbearbeiterna"

Der Alias-name hat sein Äquivalent im Formular.
Sollte hier geändert werden muß auch im Formular
geändert werden.

Wird ein Attribut nicht benötigt,
dann bekommt es den Wert null.

Zusammenhang Datenquelle Formular

Das Datenquellen-Attribut wird über seinen Namen ins **Formular** übertragen. Ist ein **Alias** angegeben, muss genau dieser auch im Formular verwendet werden. Im Basisformularsatz sollten immer Alias-Namen vergeben sein.

The screenshot shows the NetBeans IDE with the 'Mahnkonto.jrxml' file open. The 'select' tab displays the following SQL query:

```
select
  l.strname "strrland",
  k.strname "strkunde",
  k.strnummer "strkundennummer",
  k.strnamezusatz "strnamezusatz",
  k.strstrassezusatz "strstrassezusatz",
  k.strpostfach "strpostfach",
  k.strstrasse "strstrasse",
  plo.strort "strort",
  plo.strplz "strplz",
  benutzer.stunserzeichen,
  benutzer.strtelefax,
  benutzer.strbearbeiternn,
  benutzer.strbearbeitervn,
  benutzer.strtelefon
from
  t_eo_land l
  inner join t_eo_kunde k on k.intid_strl
  left outer join t_eo_postleitzahlen plo
    (select (left(ba.strvorname,1)||left(ba.
      ba.strtelefax,
      ba.strtelefon,
      ba.strvorname "strbearbeitervn",
      ba.strnachname "strbearbeiternn"
    from t_eo_bearbeiter ba
      left outer join t_md_user u on t

where
  k.intid = $intid
```

The 'Designer' tab shows the XML form structure. Red arrows connect the SQL field names to the XML field IDs:

- `strrland` (SQL) → `$R{Absender}` (XML)
- `strkunde` (SQL) → `$F{strkunde}` (XML)
- `strkundennummer` (SQL) → `$F{strkundennummer}` (XML)
- `strnamezusatz` (SQL) → `$F{strnamezusatz}` (XML)
- `strstrassezusatz` (SQL) → `$F{strstrassezusatz}` (XML)
- `strpostfach` (SQL) → `$F{strpostfach}` (XML)
- `strstrasse` (SQL) → `$F{strstrasse}` (XML)
- `strort` (SQL) → `$F{strort}` (XML)
- `strplz` (SQL) → `$F{strplz}` (XML)
- `stunserzeichen` (SQL) → `$F{stunserzeichen}` (XML)
- `strtelefax` (SQL) → `$F{strtelefax}` (XML)
- `strbearbeiternn` (SQL) → `$F{strbearbeiternn}` (XML)
- `strbearbeitervn` (SQL) → `$F{strbearbeitervn}` (XML)
- `strtelefon` (SQL) → `$F{strtelefon}` (XML)

The XML form structure also includes a title bar with 'Firmenlogo' and a main content area with the following XML code:

```
<form>
  <title>Mahnkonto</title>
  <field id="$R{Absender}" type="Text" value="$R{Absender}" />
  <field id="$F{strkunde}" type="Text" value="$F{strkunde}" />
  <field id="$F{strkundennummer}" type="Text" value="$F{strkundennummer}" />
  <field id="$F{strnamezusatz}" type="Text" value="$F{strnamezusatz}" />
  <field id="$F{strstrassezusatz}" type="Text" value="$F{strstrassezusatz}" />
  <field id="$F{strpostfach}" type="Text" value="$F{strpostfach}" />
  <field id="$F{strstrasse}" type="Text" value="$F{strstrasse}" />
  <field id="$F{strort}" type="Text" value="$F{strort}" />
  <field id="$F{strplz}" type="Text" value="$F{strplz}" />
  <field id="$F{stunserzeichen}" type="Text" value="$F{stunserzeichen}" />
  <field id="$F{strtelefax}" type="Text" value="$F{strtelefax}" />
  <field id="$F{strbearbeiternn}" type="Text" value="$F{strbearbeiternn}" />
  <field id="$F{strbearbeitervn}" type="Text" value="$F{strbearbeitervn}" />
  <field id="$F{strtelefon}" type="Text" value="$F{strtelefon}" />
  <button id="$R{mahnkonto}" type="Submit" value="$R{mahnkonto}" />
  <button id="$F{kunde.kundennr}" type="Submit" value="$F{kunde.kundennr}" />
</form>
```

Datenquelle und Formular im Zusammenspiel.

Verwandte Seiten

Reporting Formulare

Formulare.

[Öffnen](#)

Report und Formular

Datenquelle.

[Öffnen](#)