

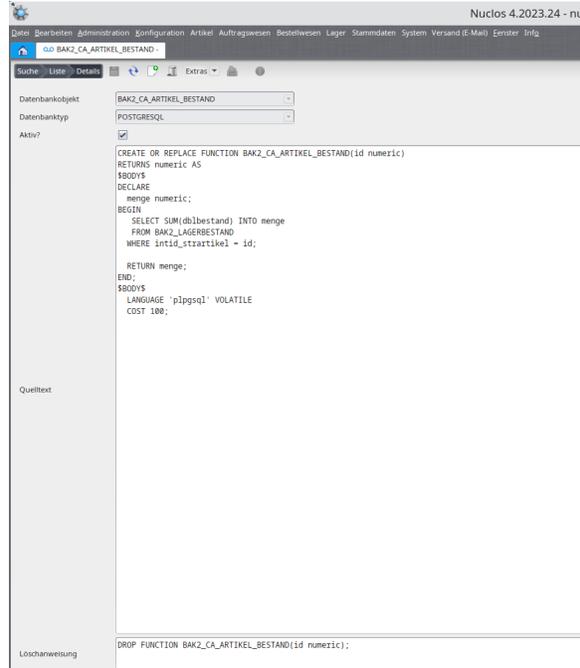
# Datenbankobjekt-Quelltexte

- Datenbankobjekt-Quelltexte
  - Rümpfe für Functions
  - Beispiel für Löschanweisung:
  - Beispiel (PostgreSQL):

## Datenbankobjekt-Quelltexte

(Konfiguration - Datenbank - Datenbankobjekt-Quelltexte)

In dieser Maske wird der eigentliche Quelltext des Datenbankobjektes eingetragen. Wenn **Aktiv** gesetzt, wird das Objekt beim Speichern in der Datenbank generiert oder aktualisiert. Der **Quelltext** stellt den DB-spezifischen Code dar. Dabei ist die entsprechende Syntax zu beachten. Beim Speichern wird der Schema- oder Datenbankname entsprechend der aktuellen Datenbankverbindung automatisch vor den Namen der Function gesetzt, z.B. `nuclos.GET_NEXT_NUMBER(...)`. Sie dürfen diesen daher an dieser Stelle nicht setzen. In der **Löschanweisung** kann das entsprechende Statement eingetragen werden. Es wird ausgeführt, wenn der Datensatz aus Nuclös gelöscht wird.



## Rümpfe für Functions

### PostgreSQL

```
CREATE OR REPLACE FUNCTION <name>(imoduleid numeric)
RETURNS numeric AS
$BODY$
$BODY$
LANGUAGE 'plpgsql' VOLATILE
COST 100;v
```

### Beispiel

```
CREATE OR REPLACE FUNCTION GET_NEXT_NUMBER(imoduleid numeric)
RETURNS numeric AS
$BODY$
DECLARE
iretult NUMERIC (20, 0);
BEGIN
SELECT intnaechstenummer INTO iretult
FROM t_eo_nummern
WHERE strbezeichnung = 'Kundenummer';

RETURN iretult;
END;
$BODY$
LANGUAGE 'plpgsql' VOLATILE
COST 100;
```

## Beispiel für Löschanweisung:

### PostgreSQL

```
DROP FUNCTION IF EXISTS GET_NEXT_NUMBER(imoduleid numeric) CASCADE
```

### Oracle

```
CREATE OR REPLACE  
FUNCTION <name>(imoduleid numeric) RETURN number IS  
  <variable>  
  
end <name>;
```

### Beispiel

```
CREATE OR REPLACE  
FUNCTION "CA_ANZAHL_POSITIONEN"(id IN number) RETURN number IS  
  nummer number(15);  
begin  
  SELECT count(*) INTO nummer  
  FROM t_eo_auftrag_positionen t1  
  WHERE intid_strreferenz=id;  
  
RETURN nummer;  
end CA_ANZAHL_POSITIONEN;
```

## Beispiel (PostgreSQL):

Berechnet die Anzahl aller Bestellwerte, die noch keine Rechnungen oder Lieferscheine enthalten. Diese Berechnung wird anhand eines dynamischen Businessobjekt durchgeführt und es wird somit auf die View zugegriffen. Der Übergabeparameter **id** stellt in die INTID von dem aktuellen Objekt dar. Dieses wird automatisch an die Funktion mit übergeben.

```
CREATE OR REPLACE FUNCTION CA_BESTELLWERTGESAMT(id numeric)  
  RETURNS numeric AS  
$BODY$  
DECLARE  
  summe numeric (15, 2);  
BEGIN  
  
  SELECT SUM(bestellwert) INTO summe  
  FROM v_de_bestellungen  
  WHERE intid_t_ud_genericobject = id;  
  
  RETURN summe;  
END;  
$BODY$  
LANGUAGE 'plpgsql' VOLATILE  
COST 100;
```



**Achtung:** Bei PostgreSQL muss immer eine Löschanweisung mit angegeben werden, die für den Fall der Löschung des Berechnetem Attribut greifen kann.