

Nachträgliches Durchnummerieren bestehender Datensätze per DB-Statement

🌐 Sprache: **Deutsch** · [English](#)

📖 Nachträgliches Durchnummerieren bestehender Datensätze per DB-Statement

Für Nummernkreise mit einfachem (nicht-datumsspezifischem) Prefix.

REFERENZ

INTEGRATOREN

STAND: JUL 2026

GILT FUER NUCLOS 4.2026.X

Auf dieser Seite

- [Verwandte Seiten](#)

Für Nummernkreise mit einfachem (nicht-datumsspezifischem) Prefix.

Beispiel für Anpassung mit postgresSQL über den Query-Editor von pgAdmin.

```
set search_path to <schemaname>;

-- Sequenz erstellen
CREATE SEQUENCE meinenummer
INCREMENT 1
MINVALUE 1
MAXVALUE 9999999999 -- muss groß genug sein um alle Datensätze durchnummerieren zu können
START <Nächste zu vergebende Nummer>;

-- Datensätze durchnummerieren
WITH L AS (
  SELECT intid, datcreated
  FROM <Tabelle>
  WHERE datcreated > make_date(<Jahr>, <Monat>, <Tag>) -- Durchnummerierung ab diesem Erstellungsdatum
  ORDER BY datcreated
  FOR UPDATE
)
UPDATE <Tabelle> SET <Nummernfeld> = '<MeinPrefix>' || NEXTVAL('meinenummer')::varchar(255)
FROM L
WHERE <Tabelle>.intid = L.intid;

-- Nummernkreis aktualisieren
UPDATE zz47_nummernkreise SET nextvalue = NEXTVAL('meinenummer')
WHERE name='<Nummernkreisname>';

-- Sequenz löschen
DROP SEQUENCE meinenummer;

-- Veränderte Datensätze anzeigen
SELECT intid, datcreated, <Nummernfeld>
FROM <Tabelle>
WHERE datcreated > make_date(<Jahr>, <Monat>, <Tag>)
ORDER BY datcreated ASC;
```

Verwandte Seiten

📖 **Module**

📖 **Nuclet Wiki**

Kategorie

[Öffnen](#)

Alle Kategorien

[Öffnen](#)