

PROCESS SQL

```
PROCESS SQL ddm-name <<statement-string>>
```

Dieses Kapitel behandelt folgende Themen:

- Funktion
- Syntax-Beschreibung
- Entire Access-Optionen
- Beispiele

Funktion

Das Statement `PROCESS SQL` dient dazu, mit SQL-Statements auf eine Datenbank zuzugreifen.

Syntax-Beschreibung

<i>ddm-name</i>	Mit dem DDM-Namen geben Sie an, für welche die Stored Procedure ausführende Datenbank die angegebenen SQL-Statements abgearbeitet werden sollen. Weitere Informationen zu <i>ddm-name</i> .
<i>statement-string</i>	<p>Die Statements, die Sie im <i>statement-string</i> angeben können, sind dieselben, die Sie auch mit dem SQL-Statement EXECUTE (vgl. <i>Flexible SQL</i>) ausführen können.</p> <p> Warnung: Um Transaktionssynchronisationsprobleme zwischen Natural und der zugrundeliegenden Datenbank zu vermeiden, dürfen die Statements COMMIT und ROLLBACK im PROCESS SQL-Statement nicht verwendet werden.</p> <p>Der <i>statement-string</i> kann über mehrere Zeilen gehen, ohne dass am Zeilenende ein Fortsetzungszeichen erforderlich ist. Er kann ganze oder teilweise Kommentarzeilen enthalten.</p> <p>Der <i>statement-string</i> darf auch Parameter enthalten; siehe <i>Parameter</i> weiter unten.</p>

Parameter

```
[ :U ] :host-variable [INDICATOR:host-variable] [LINIDICATOR:host-variable]
```

```
[ :G ]
```

Im Gegensatz zu den an anderer Stelle beschriebenen *Parametern* muss hier den *host-variables* ein Doppelpunkt (:) vorangestellt werden. Außerdem kann ihnen ein weiterer Qualifier (:U bzw. :G) vorangestellt werden.

Weitere Informationen siehe *host-variable*.

Syntax-Element-Beschreibung:

:U:host-variable	Der Präfix :U qualifiziert die <i>host-variable</i> als sogenannte Using-Variable; d.h. ihr Wert wird an die Datenbank übergeben. :U ist der Standardpräfix.
:G:host-variable	Der Präfix :G qualifiziert die <i>host-variable</i> als sogenannte Giving-Variable; d.h. sie erhält einen Wert von der Datenbank.

Entire Access-Optionen

Mit Entire Access können Sie Folgendes auch als *statement-string* angeben:

- SET SQLOPTION *option* = *value*
- SQLCONNECT *option* = *value*
- SQLDISCONNECT

Diese Optionen gelten nur für Entire Access. Sie sind im Abschnitt *Daten in einer SQL-Datenbank aufrufen* (im Leitfaden zur Programmierung) beschrieben.

Beispiele

Beispiel für Adabas D:

```
PROCESS SQL ADABAS_D_DDM << LOCK TABLE EMPLOYEES IN SHARE MODE >>
```

Beispiel für den Aufruf einer in Adabas D gespeicherten Prozedur:

Die aufgerufene Prozedur berechnet die Summe zweier Zahlen.

```
...
COMPUTE #N1 = 1
COMPUTE #N2 = 2
COMPUTE #SUM = 0
...
PROCESS SQL ADABAS_D_DDM << DBPROCEDURE DEMO.SUM (:#N1, :#N2, :G:#SUM) >>
...
WRITE #N1 '+' #N2 ' = ' #SUM
...
```