

# RETRY

RETRY

Dieses Kapitel behandelt folgende Themen:

- Funktion
- Einschränkung
- Beispiel

Verwandte Statements: ACCEPT/REJECT | AT BREAK | AT START OF DATA | AT END OF DATA | BACKOUT TRANSACTION | BEFORE BREAK PROCESSING | DELETE | END TRANSACTION | FIND | GET | GET SAME | GET TRANSACTION DATA | HISTOGRAM | LIMIT | PASSW | PERFORM BREAK PROCESSING | READ | STORE | UPDATE

Gehört zur Funktionsgruppe: *Datenbankzugriffe und Datenbankänderungen*

---

## Funktion

Das Statement `RETRY` wird in einem `ON ERROR`-Statement-Block (siehe `ON ERROR`-Statement) verwendet. Es dient dazu, erneut zu versuchen, auf einen Datensatz zuzugreifen, auf den bereits ein anderer Benutzer zugegriffen hat und der sich daher im Hold-Status befindet.

Befindet sich ein Datensatz im Hold für einen anderen Benutzer, gibt Natural die Fehlermeldung 3145 aus. Siehe auch Session-Parameter `WH` (Wait for Record in Hold Status).

Das `RETRY`-Statement muss in dem Objekt stehen, das die Fehlermeldung 3145 verursacht. Weitere Informationen zur Record-Hold-Logik finden Sie im Abschnitt *Datensatz-Kontrolle während einer Transaktion (Hold-Logik)* im *Leitfaden zur Programmierung*.

## Einschränkung

Dieses Statement kann nur für den Zugriff auf Adabas-Datenbanken verwendet werden.

## Beispiel

```
** Example 'RTYEX1': RETRY
**
** CAUTION: Executing this example will modify the database records!
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
*
1 #RETRY (A1) INIT <' '>
END-DEFINE
*
FIND EMPLOY-VIEW WITH NAME = 'ALDEN'
```

```
/*
DELETE
END TRANSACTION
/*
ON ERROR
  IF *ERROR-NR = 3145
    INPUT NO ERASE 10/1
      'RECORD IS IN HOLD' /
      'DO YOU WISH TO RETRY?' /
      #RETRY '(Y)ES OR (N)O?'
    IF #RETRY = 'Y'
      RETRY
    ELSE
      STOP
    END-IF
  END-IF
END-ERROR
/*
AT END OF DATA
  WRITE NOTITLE *NUMBER 'RECORDS DELETED'
END-ENDDATA
END-FIND
*
END
```