

# GET TRANSACTION DATA

`GET TRANSACTION [DATA] operand1 ...`

Dieses Kapitel behandelt folgende Themen:

- Funktion
- Einschränkung
- Syntax-Beschreibung
- Beispiel

Eine Erläuterung der in dem Syntax-Diagramm verwendeten Symbole entnehmen Sie dem Abschnitt *Syntax-Symbole*.

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 | HISTOGRAM | LIMIT | PASSW | PERFORM BREAK PROCESSING | READ | RETRY | STORE | UPDATE

Gehört zur Funktionsgruppe: *Datenbankzugriffe und Datenbankänderungen*

---

## Funktion

Das Statement `GET TRANSACTION DATA` dient dazu, die Transaktionsdaten, die mit einem vorherigen `END TRANSACTION`-Statement gespeichert wurden, zu lesen.

`GET TRANSACTION DATA` erzeugt keine Verarbeitungsschleife.

### Anmerkung:

Bei DL/I: Das `GET TRANSACTION DATA`-Statement liest Checkpoint-Daten, die mit einem `END TRANSACTION`-Statement gespeichert wurden.

## Systemvariable \*ETID

Um die von der Datenbank zu lesenden Transaktionsdaten zu identifizieren, kann die Natural-Systemvariable `*ETID` eingesetzt werden.

## Keine Transaktionsdaten gespeichert

Werden bei der Ausführung des `GET TRANSACTION DATA`-Statements keine Transaktionsdaten gefunden, werden alle mit dem Statement angegebenen Felder mit Leerzeichen gefüllt, gleichgültig welches Format die Felder haben.

**Warnung:**

Achten Sie darauf, dass keine arithmetischen Operationen auf der Grundlage "leerer" Transaktionsdaten ausgeführt werden, da dies einen Programmabbruch zur Folge hätte.

## Einschränkung

Das GET TRANSACTION DATA-Statement gilt nur für Transaktionen auf Adabas-Datenbanken, sowie auf DL/I-Datenbanken in einer batch-orientierten BMP-Region (nur in IMS/TM-Umgebungen).

## Syntax-Beschreibung

Operanden-Definitionstabelle:

Operand	Mögliche Struktur	Mögliche Formate	Referenzierung erlaubt	Dynam. Definition
<i>operand1</i>	S	A U N P I F B D T	ja	ja

Syntax-Element-Beschreibung:

<i>operand1</i>	<p><b>Angabe der Felder:</b></p> <p>Reihenfolge, Länge und Format der mit GET TRANSACTION DATA zu lesenden Felder muss mit Reihenfolge, Länge und Format der mit dem jeweiligen END TRANSACTION-Statement angegebenen Felder übereinstimmen.</p> <p><b>Anmerkung:</b> Bei DL/I: Der erste <i>operand1</i> muss eine 8 Byte lange Checkpoint-ID sein.</p>
-----------------	--

## Beispiel

```

** Example 'GTREX1': GET TRANSACTION
**
** CAUTION: Executing this example will modify the database records!
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 PERSONNEL-ID
  2 NAME
  2 FIRST-NAME
  2 MIDDLE-I
  2 CITY
*
1 #PERS-NR (A8) INIT <' '>
END-DEFINE
*
GET TRANSACTION DATA #PERS-NR
IF #PERS-NR NE ' '
  WRITE 'LAST TRANSACTION PROCESSED FROM PREVIOUS SESSION' #PERS-NR
END-IF
*

```

```
REPEAT
/*
INPUT 10X 'ENTER PERSONNEL NUMBER TO BE UPDATED:' #PERS-NR
IF #PERS-NR = ' '
    STOP
END-IF
/*
FIND EMPLOY-VIEW WITH PERSONNEL-ID = #PERS-NR
    IF NO RECORDS FOUND
        REINPUT 'NO RECORD FOUND'
    END-NOREC
    INPUT (AD=M) PERSONNEL-ID (AD=O)
        / NAME
        / FIRST-NAME
        / CITY

    UPDATE
    END TRANSACTION #PERS-NR
END-FIND
/*
END-REPEAT
END
```