

# RESET

`RESET [INITIAL] operand1 ...`

Dieses Kapitel behandelt folgende Themen:

- Funktion
- Syntax-Beschreibung
- Beispiel

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

Verwandte Statements: ADD | COMPRESS | COMPUTE | DIVIDE | EXAMINE | MOVE | MOVE ALL | MULTIPLY | SEPARATE | SUBTRACT

Gehört zur Funktionsgruppe: *Arithmetische Funktionen und Datenzuweisungen*

---

## Funktion

Das RESET-Statement wird benutzt, um den Wert eines Feldes zurückzusetzen.

- Mit dem Statement RESET (ohne INITIAL) können Sie in Abhängigkeit von seinem Format den Inhalt jedes angegebenen Feldes auf seinen Standard-Ausgangswert zurücksetzen.
- Mit RESET INITIAL können Sie jedes Feld auf einen im DEFINE DATA-Statement definierten Ausgangswert zurücksetzen.

Bei einem ohne INIT-Klausel im DEFINE DATA-Statement deklarierten Feld hat RESET INITIAL die gleiche Auswirkung wie RESET (ohne INITIAL).

### Anmerkungen:

1. Ein mit einer CONSTANT-Klausel im DEFINE DATA-Statement deklariertes Feld kann in einem RESET-Statement nicht referenziert werden, da sein Inhalt nicht geändert werden kann.
2. Im Reporting Mode kann das RESET-Statement auch verwendet werden, um eine Variable zu definieren, vorausgesetzt dass das Programm kein DEFINE DATA LOCAL-Statement enthält.

## Syntax-Beschreibung

Operanden-Definitionstabelle:

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

Syntax-Element-Beschreibung:

<p><b>RESET <i>operand1</i></b></p>	<p><b>Zurücksetzen auf Nullwert:</b></p> <p>RESET (ohne INITIAL) setzt jedes angegebene Feld (<i>operand1</i>) auf seinen Standard-Ausgangswert.</p> <p>Wenn <i>operand1</i> eine dynamische Variable ist, wird sie auf einen Nullwert zurückgesetzt, und zwar mit der Länge, die die Variable zum Zeitpunkt der Ausführung des RESET-Statements hat. Die aktuelle Länge einer dynamischen Variable kann mittels der Systemvariable *LENGTH ermittelt werden.</p> <p>Allgemeine Informationen über dynamische Variablen finden Sie in dem Abschnitt <i>Dynamische und große Variablen benutzen</i>.</p>
<p><b>RESET INITIAL <i>operand1</i></b></p>	<p><b>Zurücksetzen auf Ausgangswert:</b></p> <p>Mit RESET INITIAL werden die angegebenen Felder (<i>operand1</i>) auf die für sie im DEFINE DATA-Statement definierten Ausgangswerte zurückgesetzt.</p> <ul style="list-style-type: none"> <li>● Wenn für ein Feld kein Ausgangswert definiert ist, wird es abhängig von seinem Format auf einen Standard-Ausgangswert (siehe unten) zurückgesetzt.</li> <li>● Wenn eine dynamische Variable benutzt wird, wird die Systemvariable *LENGTH auf Null gesetzt, wenn kein Ausgangswert definiert ist.</li> <li>● Wenn Sie RESET INITIAL auf ein Array anwenden, müssen Sie es auf das gesamte Array (wie im DEFINE DATA-Statement definiert) anwenden; RESET INITIAL für einzelne Array-Ausprägungen ist nicht möglich.</li> <li>● Wird ein X-array verwendet, wird die Systemvariable *OCCURRENCE auf Null gesetzt.</li> <li>● Ein RESET INITIAL von aus einer Redefinition hervorgehenden Feldern ist ebenfalls nicht möglich.</li> <li>● RESET INITIAL wird für eine dynamische Variable benutzt.</li> <li>● Auf Datenbankfelder ist RESET INITIAL nicht anwendbar.</li> </ul>

## Beispiel

```

** Example 'RSTEX1': RESET (with/without INITIAL)
*****
DEFINE DATA LOCAL
1 EMPLOY-VIEW VIEW OF EMPLOYEES
  2 NAME
1 #BINARY (B4) INIT <1>
1 #INTEGER (I4) INIT <5>
1 #NUMERIC (N2) INIT <25>
END-DEFINE
*
LIMIT 1
READ EMPLOY-VIEW
/*
WRITE NOTITLE 'VALUES BEFORE RESET STATEMENT:'
WRITE / '=' NAME '=' #BINARY '=' #INTEGER '=' #NUMERIC
/*
RESET NAME #BINARY #INTEGER #NUMERIC
/*
WRITE /// 'VALUES AFTER RESET STATEMENT:'
WRITE / '=' NAME '=' #BINARY '=' #INTEGER '=' #NUMERIC
/*
RESET INITIAL #BINARY #INTEGER #NUMERIC
/*
WRITE /// 'VALUES AFTER RESET INITIAL STATEMENT:'
WRITE / '=' NAME '=' #BINARY '=' #INTEGER '=' #NUMERIC
/*
END-READ
END

```

### Ausgabe des Programms RSTEX1:

VALUES BEFORE RESET STATEMENT:

```

NAME: ADAM                #BINARY: 00000001 #INTEGER:          5 #NUMERIC:
25

```

VALUES AFTER RESET STATEMENT:

```

NAME:                    #BINARY: 00000000 #INTEGER:          0 #NUMERIC:
0

```

VALUES AFTER RESET INITIAL STATEMENT:

```

NAME:                    #BINARY: 00000001 #INTEGER:          5 #NUMERIC:
25

```