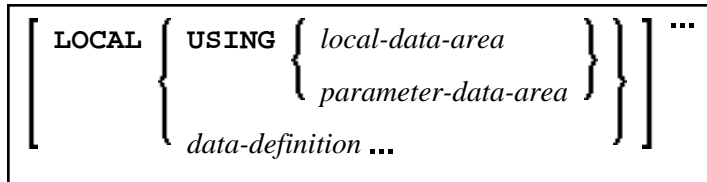


Definition von Local Data

Allgemeine Syntax von DEFINE DATA LOCAL:



Dieses Kapitel behandelt folgende Themen:

- Funktion
- Einschränkung
- Syntax-Beschreibung

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

Funktion

Das DEFINE DATA LOCAL-Statement dient zur Definition der Datenelemente, die ausschließlich von einem einzelnen Natural-Modul in einer Anwendung benutzt werden. Diese Elemente oder Felder können innerhalb des Statements selbst definiert werden (siehe *data-definition*); oder sie können außerhalb des Programms in einer separaten Local Data Area (LDA) oder einer Parameter Data Area (PDA) definiert werden, wobei das Statement diese Data Area referenziert.

Einschränkung

Die LDA und die sie referenzierenden Objekte müssen in derselben Library (oder in einer Steplib) enthalten sein.

Syntax-Beschreibung

<i>local-data-area</i>	<p>Eine Local Data Area enthält vordefinierte Datenelemente, die in das <code>DEFINE DATA LOCAL</code>-Statement übernommen werden können. Sie können mehr als eine Data Area referenzieren; in diesem Fall müssen Sie die reservierten Wörter <code>LOCAL</code> und <code>USING</code> wiederholen, zum Beispiel:</p> <pre>DEFINE DATA LOCAL USING DATX_L LOCAL USING DATX_P . . . END-DEFINE ;</pre> <p>Weitere Informationen siehe auch <i>Felder in einer separaten Data Area und Local Data Area</i> und <i>Local Data Area, Beispiel 2</i> im Leitfaden zur Programmierung.</p>
<i>parameter-data-area</i>	<p>Eine mit <code>DEFINE DATA LOCAL</code> referenzierte Data Area kann auch eine Parameter Data Area (PDA) sein. Durch Benutzung einer PDA als LDA können Sie sich die zusätzliche Mühe sparen, eine LDA zu erstellen, die dieselbe Struktur wie die PDA hat.</p>
<i>direct-data-definition</i>	<p>Siehe <i>Direkte Daten-Definition</i> weiter unten.</p>
END-DEFINE	<p>Das für Natural reservierte Wort <code>END-DEFINE</code> muss zum Beenden des <code>DEFINE DATA</code>-Statements benutzt werden.</p>

Direkte Daten-Definition

Local Data können direkt in einem Programm oder Subprogramm definiert werden. Für eine direkte Daten-Definition gilt die folgende Syntax:

$\left\{ \begin{array}{l} \textit{level} \left\{ \begin{array}{l} \textit{group-name} [(\textit{array-definition})] \\ \textit{variable-definition} \\ \textit{view-definition} \\ \textit{redefinition} \\ \textit{handle-definition} \end{array} \right\} \end{array} \right\}$

Weitere Informationen siehe

- *Beispiel 1 - DEFINE DATA LOCAL* (Direkte Daten-Definition)
- *Definition von Feldern in einem DEFINE DATA-Statement* im Leitfaden zur Programmierung
- *Local Data Area, Beispiel 1* im Leitfaden zur Programmierung

Syntax-Elementbeschreibung für die direkte Daten-Definition:

<i>level</i>	<p>Dies ist eine ein- oder zweistellige Zahl im Bereich von 01 bis 99 (die vorangestellte 0 ist nicht erforderlich), die in Verbindung mit der Gruppierung von Feldern verwendet wird. Felder mit einer Level-Nummer von 02 an aufwärts werden als Teil einer unmittelbar vorangehenden Gruppe mit einer jeweils nächstniedrigeren Level-Nummer betrachtet.</p> <p>Durch die Definition einer Gruppe (die auch nur aus einem Feld bestehen kann) ist es möglich, durch Angabe lediglich des Gruppennamens eine ganze Reihe von aufeinanderfolgenden Feldern gleichzeitig zu referenzieren. Bei manchen Statements (CALL, CALLNAT, RESET, WRITE usw.) können Sie den Gruppennamen als Aufrufnamen angeben, um die in der Gruppe enthaltenen Felder zu referenzieren.</p> <p>Eine Gruppe kann aus anderen Gruppen bestehen. Bei der Vergabe von Level-Nummern für eine Gruppe darf kein Level ausgelassen werden.</p> <p>Eine <i>view-definition</i> muss immer auf Level 1 definiert werden.</p>
<i>group-name</i>	<p>Der Name einer Gruppe. Der Name muss den Regeln zur Definition eines Natural-Variablennamens entsprechen. Siehe auch die folgenden Abschnitte:</p> <ul style="list-style-type: none"> • <i>Namen von Benutzervariablen</i> in der Dokumentation <i>Natural benutzen</i>. • <i>Datenstrukturen qualifizieren</i> im Leitfaden zur Programmierung.
<i>array-definition</i>	<p>Mit <i>array-definition</i> definieren Sie die untere und obere Grenze einer Dimension in einer Array-Definition. Siehe <i>Definition von Array-Dimensionen</i>.</p>
<i>variable-definition</i>	<p>Die <i>variable-definition</i> dient zur Definition einer einzelnen Variablen (oder Feldes), die aus einem Wert (Skalar) oder mehreren Werten (Array) bestehen kann. Siehe <i>Definition von Variablen</i>.</p>
<i>view-definition</i>	<p>Die <i>view-definition</i> wird benutzt, um eine Datensicht (View) mit Bestandteilen aus einem Datendefinitionsmodul (DDM) zu definieren. Siehe <i>View Definition</i>.</p>
<i>redefinition</i>	<p>Eine <i>redefinition</i> kann zur Redefinition einer Gruppe, eines Views, eines DDM-Felds oder eines einzelnen Feldes oder einer einzelnen Variablen benutzt werden (d.h. Skalar oder Array). Siehe <i>Redefinition</i>.</p>
<i>handle-definition</i>	<p>Eine Handle identifiziert ein Dialog-Element im Code und wird in Handle-Variablen gespeichert. Siehe <i>Handle-Definition</i>.</p>