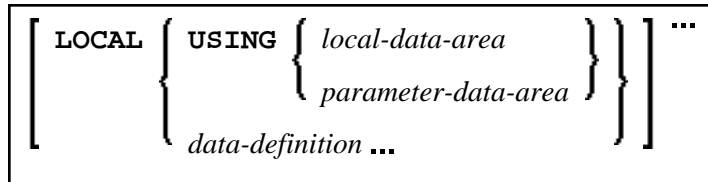


# Defining Local Data

General syntax of `DEFINE DATA LOCAL`:



This chapter covers the following topics:

- Function
- Restriction
- Syntax Description

For an explanation of the symbols used in the syntax diagram, see *Syntax Symbols*.

---

## Function

The `DEFINE DATA LOCAL` statement is used to define the data elements that are to be used exclusively by a single Natural module in an application. These elements or fields can be defined within the statement itself (see *data-definition*); or they can be defined outside the program in a separate local data area (LDA) or a parameter data area (PDA), with the statement referencing that data area.

## Restriction

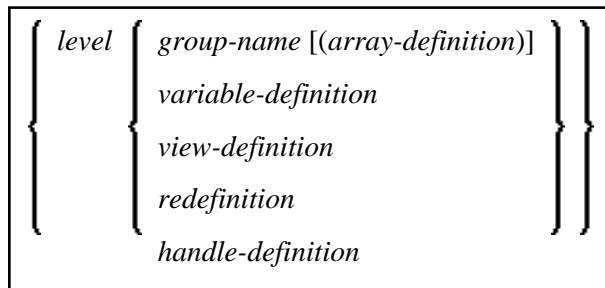
The LDA and the objects which reference it must be contained in the same library (or in a steplib).

## Syntax Description

<b><i>local-data-area</i></b>	<p>A local data area contains predefined data elements which can be included in the <code>DEFINE DATA LOCAL</code> statement. You may reference more than one data area; in that case you have to repeat the reserved words <code>LOCAL</code> and <code>USING</code>, for example:</p> <pre>DEFINE DATA LOCAL   LOCAL USING DATX_L   LOCAL USING DATX_P   . . . END-DEFINE ;</pre> <p>For further information, see also <i>Defining Fields in a Separate Data Area</i> and <i>Local Data Area, Example 2</i> in the <i>Programming Guide</i>.</p>
<b><i>parameter-data-area</i></b>	<p>A data area referenced with <code>DEFINE DATA LOCAL</code> may also be a parameter data area (PDA). By using a PDA as an LDA you can avoid the extra effort of creating an LDA that has the same structure as the PDA.</p>
<b><i>direct-data-definition</i></b>	<p>See <i>Direct Data Definition</i> below.</p>
<b>END-DEFINE</b>	<p>The Natural reserved word <code>END-DEFINE</code> must be used to end the <code>DEFINE DATA</code> statement.</p>

## Direct Data Definition

Local data can be defined directly within a program or routine. For direct data definition, the following syntax applies:



For further information, see

- *Example 1 - DEFINE DATA LOCAL* (Direct Data Definition)
- *Defining Fields within a DEFINE DATA Statement* in the *Programming Guide*
- *Local Data Area, Example 1* in the *Programming Guide*

Syntax Element Description for Direct Data Definition:

<b><i>level</i></b>	<p>Level number is a 1- or 2-digit number in the range from 01 to 99 (the leading zero is optional) used in conjunction with field grouping. Fields assigned a level number of 02 or greater are considered to be a part of the immediately preceding group which has been assigned a lower level number.</p> <p>The definition of a group enables reference to a series of fields (may also be only 1 field) by using the group name. With certain statements (<i>CALL</i>, <i>CALLNAT</i>, <i>RESET</i>, <i>WRITE</i>, etc.), you may specify the group name as a shortcut to reference the fields contained in the group.</p> <p>A group may consist of other groups. When assigning the level numbers for a group, no level numbers may be skipped.</p> <p>A view-definition must always be defined at Level 1.</p>
<b><i>group-name</i></b>	<p>The name of a group. The name must adhere to the rules for defining a Natural variable name. See also the following sections:</p> <ul style="list-style-type: none"> <li>● <i>Naming Conventions for User-Defined Variables</i> in the <i>Using Natural</i> documentation.</li> <li>● <i>Qualifying Data Structures</i> in the <i>Programming Guide</i>.</li> </ul>
<b><i>array-definition</i></b>	<p>With an <i>array-definition</i>, you define the lower and upper bounds of dimensions in an array-definition. See <i>Array Dimension Definition</i>.</p>
<b><i>variable-definition</i></b>	<p>A <i>variable-definition</i> is used to define a single field/variable that may be single-valued (scalar) or multi-valued (array). See <i>Variable Definition</i>.</p>
<b><i>view-definition</i></b>	<p>A <i>view-definition</i> is used to define a view as derived from a data definition module (DDM). See <i>View Definition</i>.</p>
<b><i>redefinition</i></b>	<p>A <i>redefinition</i> may be used to redefine a group, a view, a DDM field or a single field/variable (that is a scalar or an array). See <i>Redefinition</i>.</p>
<b><i>handle-definition</i></b>	<p>A handle identifies a dialog element in code and is stored in handle variables. See <i>Handle Definition</i>.</p>