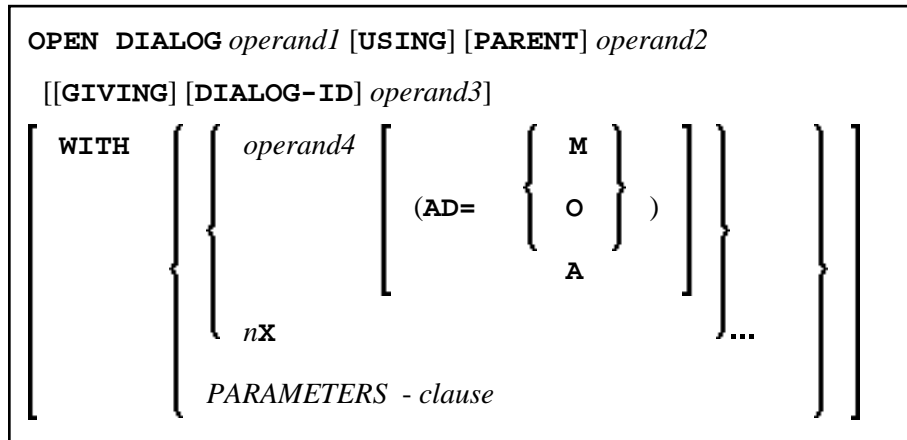


OPEN DIALOG



This chapter covers the following topics:

- Function
- Syntax Description
- Further Information and Examples

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

Related Statement: CLOSE DIALOG | PROCESS GUI | SEND EVENT

Belongs to Function Group: *Event-Driven Programming*

Function

This statement is used to open a dialog dynamically.

Syntax Description

Operand Definition Table:

Operand	Possible Structure	Possible Formats	Referencing Permitted	Dynamic Definition
operand1	C S	A	yes	no
operand2	C S		no	no
operand3	S	I	yes	no
operand4	C S A	A U N P I F B D T L C G O	yes	no

Syntax Element Description:

Syntax Element	Description
<i>operand1</i>	<p>Dialog Name:</p> <p><i>operand1</i> is the name of the dialog to be opened. <i>operand1</i> must be a constant if the <i>PARAMETERS-clause</i> is used.</p>
<i>operand2</i>	<p>Handle Name:</p> <p><i>operand2</i> is the handle name of the parent.</p>
<i>operand3</i>	<p>Dialog ID:</p> <p><i>operand3</i> is a unique identifier returned from the creation of the dialog. It must be defined with format/length I4.</p>
<i>operand4</i>	<p>Passing Parameters to the Dialog:</p> <p>When a dialog is opened, parameters may be passed to this dialog.</p> <p>As <i>operand4</i> you specify the parameters which are passed to the dialog.</p>
<i>PARAMETERS-clause</i>	<p>Passing Parameters Selectively:</p> <p>With the <i>PARAMETERS-clause</i>, parameters may be passed selectively. For full details, see <i>PARAMETERS Clause</i> below.</p> <p>Note: You can only use the <i>PARAMETERS-clause</i> if <i>operand1</i> is a constant and the dialog is cataloged.</p>
<i>nX</i>	<p>Specifying Parameters to be Skipped:</p> <p>With the notation <i>nX</i> you can specify that the next <i>n</i> parameters are to be skipped (for example, 1X to skip the next parameter, or 3X to skip the next three parameters); this means that for the next <i>n</i> parameters no values are passed to the dialog. A parameter that is to be skipped must be defined with the keyword OPTIONAL in the dialog's DEFINE DATA PARAMETER statement. OPTIONAL means that a value can - but need not - be passed from the invoking object to such a parameter.</p>

Syntax Element	Description
AD=	<p>Attribute Assignment: If <i>operand4</i> is a variable, you can mark it in one of the following ways:</p> <p>AD=O Non-modifiable, see session parameter AD=O.</p> <p>AD=M Modifiable, see session parameter AD=M.</p> <p>AD=A Input only, see session parameter AD=A.</p> <p><i>operand4</i> cannot be explicitly specified if <i>operand4</i> is a constant. AD=O always applies to constants.</p>

PARAMETERS Clause

```
PARAMETERS {parameter-name = operand4}...
END-PARAMETERS
```

Syntax Element Description:

Syntax Element	Description
<i>parameter-name</i>	<p>Parameter Name:</p> <p>The name of the parameter as defined in the parameter data area section of the dialog.</p> <p>Note: If the value of a parameter marked with AD=O and passed "by reference" is changed in a dialog, this will lead to a runtime error.</p>
<i>operand4</i>	<p>Parameters to be Passed:</p> <p>As <i>operand4</i> you specify the parameters which are passed to the dialog.</p>
END-PARAMETERS	<p>End of PARAMETERS Clause:</p> <p>The Natural reserved word END-PARAMETERS must be used to end the PARAMETERS clause.</p>

Further Information and Examples

See the section *Event-Driven Programming Techniques* in the *Programming Guide*.