

# Natural System Error Messages 0201-0250

## **NAT0207: REDEFINE group not allowed in DEFINE DATA PARAMETER.**

Text	REDEFINE group not allowed in DEFINE DATA PARAMETER.
Expl.	Only elementary fields may be defined in a DEFINE DATA PARAMETER statement.
Actn.	Check program and correct error.

## **NAT0214: Specify "FIRST" or "EVERY" in DECIDE statement.**

Text	Specify "FIRST" or "EVERY" in DECIDE statement.
Expl.	See the Natural documentation for detailed information on the DECIDE statement.
Actn.	Check program and correct error.

## **NAT0215: Keyword "VALUE" missing in DECIDE ON statement.**

Text	Keyword "VALUE" missing in DECIDE ON statement.
Expl.	The keyword "VALUE" (or "VALUES") is a required part of the DECIDE ON syntax.
Actn.	Check program and correct error.

## **NAT0216: Invalid field specification in DECIDE ON statement.**

Text	Invalid field specification in DECIDE ON statement.
Expl.	See the Natural documentation for information on how to specify fields correctly in a DECIDE ON statement.
Actn.	Check program and correct error.

## **NAT0218: Error in construction of DECIDE ON/DECIDE FOR statement.**

Text	Error in construction of DECIDE ON/DECIDE FOR statement.
Expl.	See the Natural documentation for information on the correct syntax for the statements DECIDE FOR and DECIDE ON.
Actn.	Check program and correct error.

## **NAT0219: ALL clause is only allowed with DECIDE FOR/ON EVERY.**

Text	ALL clause is only allowed with DECIDE FOR/ON EVERY.
Expl.	The ALL clause can only be used with DECIDE FOR EVERY and DECIDE ON EVERY, not for DECIDE FOR FIRST or DECIDE ON FIRST.
Actn.	Check program and correct error.

**NAT0220: Invalid statement :1:.**

Text	Invalid statement ....
Expl.	<p>Possible reasons:</p> <ol style="list-style-type: none"> <li>1. Depending on the programming mode (structured/reporting mode), keywords are missing (e.g., "COMPUTE" and "ASSIGN" must not be omitted when using ROUNDED in structured mode).</li> <li>2. The statement is not suitable for this object type (e.g., DEFINE SUBROUTINE must not be the first statement in an object of type function).</li> <li>3. A compatibility switch is set (e.g., the profile parameter V41MFCOMP), which causes some statements to be disallowed.</li> </ol>
Actn.	<ol style="list-style-type: none"> <li>1. Add the missing keywords.</li> <li>2. Change the object type or the statement.</li> <li>3. Switch off the compatibility switch.</li> </ol>

**NAT0222: Syntactical errors prevent execution of program.**

Text	Syntactical errors prevent execution of program.
Expl.	<p>In batch mode, Natural always continues the syntax checking of a program even after an error has been detected.</p> <p>A program containing a syntactical error cannot be executed because the compiled code does not agree with the source input.</p>
Actn.	Check program and correct error.

**NAT0223: Keyword "MAP" or "FORM" is required after "USING".**

Text	Keyword "MAP" or "FORM" is required after "USING".
Actn.	Check program and correct error.

**NAT0224: Input map in WRITE or write map in INPUT statement.**

Text	Input map in WRITE or write map in INPUT statement.
Expl.	Either you have specified in a WRITE USING MAP statement the name of a map which is an input map, not an output map; or you have specified in an INPUT USING MAP statement the name of a map which is an output map, not an input map. Whether a map is an input or output (write) map is indicated in its Map Settings.
Actn.	Check program and correct error.

**NAT0225: GIVING LENGTH is only valid for non-indexable variables.**

Text	GIVING LENGTH is only valid for non-indexable variables.
Actn.	Check program and correct error.

**NAT0226: An empty statement block is not allowed.**

Text	An empty statement block is not allowed.
Expl.	The only way of having a statement block which does nothing is by filling it with an IGNORE statement.
Actn.	Correct error in program.

**NAT0227: Invalid variable/constant in DELIMITER clause.**

Text	Invalid variable/constant in DELIMITER clause.
Expl.	If the receiving field of the COMPRESS statement is of format - alphanumeric or binary, the delimiter format has to be A1, B1 or U1. - Unicode, the delimiter format has to be A1, B2 or U1.
Actn.	Check program and correct error.

**NAT0228: Block structure of GDA must be specified in WITH clause.**

Text	Block structure of GDA must be specified in WITH clause.
Expl.	In this GDA, a block structure is defined. The block structure must be specified in the WITH clause of DEFINE DATA GLOBAL USING.
Actn.	Check program and correct error.

**NAT0229: Invalid use of WITH clause in UPDATE or STORE statement.**

Text	Invalid use of WITH clause in UPDATE or STORE statement.
Expl.	If an UPDATE or STORE statement is based on a view defined with DEFINE DATA or within a data area, the use of the WITH clause is not permitted. An UPDATE or STORE statement always includes all fields when it applies to a view.
Actn.	Remove the WITH clause from the UPDATE/STORE statement.

**NAT0230: Operand type not supported.**

Text	Operand type not supported.
Expl.	Operands defined as - HANDLE or UNICODE, - alphanumeric/binary DYNAMIC, - alphanumeric with a length of more than 253 bytes, or - binary with a length of more than 126 bytes are not supported at this position.
Actn.	Check program and correct error.

**NAT0231: Only NSD in linkage attributes allowed here.**

Text	Only NSD in linkage attributes allowed here.
Expl.	An NSD attribute is an element associated with a view. In a FIND statement, one expects an attribute name after the "VIA" keyword in a REFERENCED, REFERENCING or RECURSIVELY clause. In a SHOW statement, one expects attributes after SHOW, or SHOW RECURSIVELY, in brackets as follows: SHOW ATTR1(ATTR2(ATTR3))
Actn.	Check that the attribute is found in the view specified by "L F <view-name>" and correct the source accordingly.

**NAT0232: Operand of type DYNAMIC not allowed at this position.**

Text	Operand of type DYNAMIC not allowed at this position.
Expl.	Alphanumeric or binary operands defined as DYNAMIC cannot be used at this position.
Actn.	Replace DYNAMIC field with FIXED length field.

**NAT0233: Level number must be numeric integer.**

Text	Level number must be numeric integer.
Expl.	This refers to the level of recursion a FIND statement execution is to go to. The level number must be numeric. For further information, please refer to the FIND statement, UNTIL clause, in the Entire DB Programmer's Guide.
Actn.	Correct error in program.

**NAT0234: "ERE SET" must start with a category name.**

Text	"ERE SET" must start with a category name.
Expl.	In a FIND statement, after the FIND header for an Entire view, the syntax logic searches for components called "ERE SET"s. An "ERE SET" begins with a view name and ends with a search criterion. The search criterion finishes when a Natural keyword or the view name of the next "ERE SET" is encountered. When neither of these conditions is met, this error appears; and the syntax analysis cannot proceed.
Actn.	Check source code and correct error.

**NAT0235: ISN must be an integer value.**

Text	ISN must be an integer value.
Expl.	In a FIND statement, ISN values in the RELATIONSHIP or ISN clauses of the basic search criterion must be integer values. The ISNs found are numeric, but not integer values. For further information, please refer to the FIND statement, RELATIONSHIP and ISN clauses of the basic search criterion, in the Entire DB Programmer's Guide.
Actn.	Correct error in program.

**NAT0236: Equal sign "=" missing after Entire keyword.**

Text	Equal sign "=" missing after Entire keyword.
Expl.	This error can occur in two Entire statements: 1) FIND view-name WITH ISN = 12335 The equal sign is necessary in the ISN clause of the basic search criterion of the FIND statement. 2) DLOGON USER = "XXX" PASSWORD = "YYY" LIBRARY = "LLL" APPLICATION = "AER" AUTHORIZATION = A In this syntax form of the DLOGON statement, the equal sign must follow each keyword. For further information, see FIND and DLOGON statements in the Entire DB Programmer's Guide.
Actn.	Correct error in program.

**NAT0237: Unmatched closing parenthesis.**

Text	Unmatched closing parenthesis.
Expl.	Entire DB checks whether each opening parenthesis is paired with a corresponding closing parenthesis.
Actn.	Correct error in program.

**NAT0238: Invalid operator in descriptor search criterion.**

Text	Invalid operator in descriptor search criterion.
Expl.	<p>In a FIND statement, the following relational operators are valid in a descriptor search criterion:</p> <p>= (equal to)  &lt;&gt; (not equal)  &lt; (less than)  &lt;= (less than or equal to)  &gt; (greater than)  &gt;= (greater than or equal to)</p> <p>In descriptor search criteria (following WITH), only the above operators are allowed after the attribute (NAME or LINE-NUMBER).  See the description of the FIND statement the in Entire DB Programmer"s Guide for further information.</p>
Actn.	Correct descriptor search criterion.

**NAT0239: Move values for Entire must be integer values.**

Text	Move values for Entire must be integer values.
Expl.	<p>A move value is a constant or variable moved into the search buffer of an Entire call at runtime.</p> <p>When this message appears, a numeric value is expected: integer, numeric unpacked or packed without decimal digits, but not binary.</p>
Actn.	Correct error in program.

**NAT0240: Operand format invalid. Format :1: expected.**

Text	Operand format invalid. Format ... expected.
Expl.	<p>The format of this operand is invalid.</p> <p>For further information on valid formats, see the operand definition table of this statement.</p>
Actn.	Supply a valid operand format.

**NAT0242: Open bracket "(" expected.**

Text	Open bracket "(" expected.
Expl.	Entire DB checks for open parentheses in the NATIVE and RELATIONSHIP clauses of the FIND statement. Clause elements must be enclosed in parentheses. See FIND statement in the Entire DB Programmer's Guide for further information.
Actn.	Correct error in program.

**NAT0243: Syntax error in DEFINE DATA statement/structure.**

Text	Syntax error in DEFINE DATA statement/structure.
Expl.	See the Natural documentation for information on the correct syntax of the DEFINE DATA statement.
Actn.	Check program and correct error.

**NAT0244: "GLOBAL" must be followed by "USING" and GDA name.**

Text	"GLOBAL" must be followed by "USING" and GDA name.
Expl.	A global data area (GDA) is always a separate object, never part of the data definition within the DEFINE DATA statement of a program. The GDA is referenced in the GLOBAL clause of the DEFINE DATA statement with the following syntax: DEFINE DATA GLOBAL USING name-of-gda ... END-DEFINE
Actn.	Check program and correct error.

**NAT0245: Invalid use/order of clauses in DEFINE DATA statement.**

Text	Invalid use/order of clauses in DEFINE DATA statement.
Expl.	The sequence of clauses in the DEFINE DATA statement is: DEFINE DATA GLOBAL USING PARAMETER OBJECT LOCAL In a subsequent validation rule, only DEFINE DATA LOCAL is allowed.
Actn.	Make sure that the clauses are in the correct order.

**NAT0246: Formats "C", "L" and HANDLE OF OBJECT not allowed here.**

Text	Formats "C", "L" and HANDLE OF OBJECT not allowed here.
Expl.	Data types "C" (control) and "L" (logical) cannot be used in: - GET TRANSACTION DATA statement, - END TRANSACTION statement, - sort criteria of a SORT statement. Data type HANDLE OF OBJECT cannot be used in: - GET TRANSACTION DATA statement, - END TRANSACTION statement, - SORT statement.
Actn.	Check program and correct error.

**NAT0247: Error during processing of parameter :1:.**

Text	Error during processing of parameter ....
Expl.	When an INPUT/WRITE USING MAP or a PROCESS PAGE USING statement is used with a constant map or adapter name and without explicit parameters, the names of the fields to be processed are taken dynamically from the map or adapter source at compilation. This requires that the same field names with the same array structure be defined in both the program and the map or adapter. The indicated variable name is either not defined or the number of array dimensions does not match.
Actn.	Check the indicated data field. The variable definition in the program should be same as in the map or adapter.

**NAT0248: Invalid USING variable in PROCESS statement.**

Text	Invalid USING variable in PROCESS statement.
Expl.	The left part of the ASSIGN variable in the USING clause of the PROCESS statement must be part of the view defined to Natural PROCESS.
Actn.	Check program and correct error.

**NAT0249: Error in "ASSIGN" of Natural PROCESS "USING" clause.**

Text	Error in "ASSIGN" of Natural PROCESS "USING" clause.
Actn.	Check program and correct error.

**NAT0250: Error in GIVING clause of Natural "PROCESS" statement.**

Text	Error in GIVING clause of Natural "PROCESS" statement.
Actn.	Check program and correct error.