

Natural SQL Statements - **Syntactical** Items

The following **common** syntactical items are either DB2-specific and do not conform to the standard SQL syntax definitions (that is, to the Common Set of Natural SQL syntax) or impose restrictions when used with DB2 (see also SQL Statements in the Natural Statements documentation).

This section covers the following topics:

- atom
 - comparison
 - factor
 - scalar-function
 - **column-function**
 -
-

atom

An atom can be either a parameter (that is, a Natural program variable or host variable) or a constant. When running dynamically, however, the use of host variables is restricted by DB2. For further details, refer to the relevant literature on DB2 **by IBM**.

comparison

The following three comparison operators are specific to DB2 and belong to the Natural Extended Set.

¬ =
¬ >
¬ <

factor

The following three factors are specific to DB2 and belong to the Natural Extended Set:

special-register
scalar-function (*scalar-expression*, ...)
scalar-expression unit
case-expression

scalar-function(???)

A **scalar function** (*ohne Bindestrich, lt. IBM???*) is a built-in function that can be used in the construction of scalar computational expressions. Scalar functions are specific to DB2 and belong to the Natural Extended Set.

The scalar functions NDB supports are listed below in alphabetical order:

A - H	I - R	S - Z
ABS	IDENTITY_VAL_LOCAL	SECOND
ABSVAL	IFNULL	SIGN
ACOS	INSERT	SIN
ADD_MONTHS	INTEGER	SINH
ASIN	JULIAN_DAY	SMALLINT
ATAN	LAST_DAY	SPACE
ATAN2	LCASE	SQRT
ATANH	LEFT	STRIP
BLOB	LENGTH	SUBSTR
CCSID_ENCODING	LN	TAN
CEIL	LOCATE	TANH
CEILING	LOG	TIME
CHAR	LOG10	TIMESTAMP
CLOB	LOWER	TIMESTAMP_FORMAT
COALESCE	LTRIM	TO_CHAR
CONCAT	MAX	TO_DATE
COS	MICROSECOND	TRANSLATE
COSH	MIDNIGHT_SECONDS	TRUNC
DATE	MIN	TRUNC_TIMESTAMP
DAY	MINUTE	TRUNCATE
DAYOFMONTH	MOD	UCASE
DAYOFWEEK	MONTH	UPPER
DAYOFWEEK_ISO	MULTIPLY_ALT	VALUE
DAYOFYEAR	NEXT_DAY	VARCHAR
DAYS	NULLIF	VARCHAR_FORMAT
DBCLOB	POSSTR	VARGRAPHIC
DEC	POWER	WEEK
DECIMAL	QUARTER	WEEK_ISO
DEGREES	RADIANS	YEAR
DIGITS	RAISE_ERROR	
DOUBLE	RAND	
DOUBLE-PRECISION	REAL	
(Bindestrich???)	REPEAT	
EXP	REPLACE	
FLOAT	RIGHT	
FLOOR	ROUND	
GRAPHIC	ROUND_TIMESTAMP	
HEX	ROWID	
HOUR	RTRIM	

Each scalar function is followed by one or more scalar expressions in parentheses. The number of scalar expressions depends upon the scalar function. Multiple scalar expressions must be separated from one another by commas.

```

Example:

SELECT NAME
  INTO NAME
  FROM SQL-PERSONNEL
 WHERE SUBSTR ( NAME, 1, 3 ) = 'Fri'
  ...
    
```

column-function (klein???mit Bindestrich erforderlich???)

The following column functions do not conform to standard SQL. They are specific to DB2 and belong to the Natural Extended Set. Column functions operate on a set of values that derive from an expression (???) and return the defined value (???) or the NULL value.

AVG
COUNT
COUNT_BIG
MAX
MIN
STDDEV
STDDEV_POP
STDDEV_SAMP
SUM
VAR
VAR_POP
VAR_SAMP
VARIANCE
VARIANCE_SAMP

scalar-operator

The concatenation operator (CONCAT or "//") does not conform to standard SQL. It is specific to DB2 and belongs to the Natural Extended Set.

special-register

The following special registers do not conform to standard SQL. They are specific to DB2 and belong to the Natural Extended Set:

CURRENT APPLICATION ENCODING SCHEME
CURRENT DATE
CURRENT_DATE (???)
CURRENT DEGREE
CURRENT FUNCTION PATH
CURRENT_LC_CTYPE (???)
CURRENT LC_CTYPE
CURRENT LOCALE LC_CTYPE
CURRENT OPTIMIZATION HINT
CURRENT PACKAGESET
CURRENT_PATH
CURRENT PRECISION
CURRENT RULES
CURRENT SQLID
CURRENT SERVER
CURRENT TIME
CURRENT_TIME (???)
CURRENT TIMESTAMP
CURRENT TIMEZONE
CURRENT_TIMEZONE (???)
USER

A reference to a special register returns a scalar value.

Using the command SET CURRENT SQLID, the creator name of a table can be substituted by the current SQLID. This enables you to access identical tables with the same table name but with different creator names.

units

Units, also called durations, are specific to DB2 and belong to the Natural Extended Set.

The following units are supported:

DAY
 DAYS
 HOUR
 HOURS
 MICROSECOND
 MICROSECONDS
 MINUTE
 MINUTES
 MONTH
 MONTHS
 SECOND
 SECONDS
 YEAR
 YEARS

case-expression

<p>CASE { <i>searched-when-clause ...</i> } { <i>simple-when-clause</i> } { ELSE { NULL } { <i>scalar expression</i> } } { END</p>
--

Case-expressions do not conform to standard SQL and are therefore supported by the Natural SQL Extended Set only.

Example:

```

DEFINE DATA LOCAL
01 #EMP
02 #EMPNO (A10)
02 #FIRSTNME (A15)
02 #MIDINIT (A5)
02 #LASTNAME (A15)
02 #EDLEVEL (A13)
02 #INCOME (P7)
END-DEFINE
SELECT EMPNO, FIRSTNME, MIDINIT, LASTNAME,
      (CASE WHEN EDLEVEL < 15 THEN 'SECONDARY'
            WHEN EDLEVEL < 19 THEN 'COLLEGE'
            ELSE 'POST GRADUATE'
            END ) AS EDUCATION, SALARY + COMM AS INCOME
INTO
      FROM DSN8510-EMP
      WHERE (CASE WHEN SALARY = 0 THEN NULL
                ELSE SALARY / COMM

```

```
END ) > 0.25  
DISPLAY #EMP  
END-SELECT  
END
```