

## SQL Functions used in GuardDraw Expressions

Database: **SQL Server 7.0**

When building expressions with GuardDraw V5 using an ODBC connection to an SQLServer 7.0 database, please refer to the present document or your SQLServer 7.0 documentation.

### Operators

- + Concatenation
- ' Text delimiter
- + Addition
- Subtraction

### String Functions

- ASCII(char)** Returns the ASCII (0-255) value of the first character in the string "str". Example: ASC('ascii') returns 97 (ASCII value for 'a').
- CHAR(number)** Returns the character for which the ASCII code is "number". Example: CHR(10) returns a line feed.
- CHARINDEX(ch1, ch2)** Returns the starting position of "ch1" in "ch2". Example: CHARINDEX('123', '9876523123987567') will return 8.
- DATALength (char)** Returns the length of string "char". Example: DATALength('0987654') will return 7.
- LEFT(str,n)** Returns the first "n" characters of "str". Example: LEFT ('Mary',1) will return 'M'.
- LOWER(str)** Returns the string "str" with all characters converted to lower case. Example: LOWER('Mary') will return 'mary'.
- LTRIM(str)** Removes leading blanks in "str". Example: LTRIM(' Mary') will return 'Mary'.
- RIGHT(char,n)** Returns a sub-string of "char" starting at the "n"<sup>th</sup> characters from the right. Example: RIGHT('1234500',3) will return '500'.
- RTRIM(str)** Removes trailing spaces in "str". Example: RTRIM('Mary ') will return 'Mary'.
- SUBSTRING(str, m, n)** Returns a sub-string of "str", beginning at character "m", "n" characters long. Example: SUBSTR('00012345',4,3) will return '123'.
- UPPER(str)** Converts the characters of "str" to upper case. Example: UPPER('Mary') will return 'MARY'.

### Numeric Functions

- CEILING (num)** Returns the smallest integer greater than or equal to "n". Example: CEILING(99.8) returns 100.
- FLOOR(n)** Returns the largest integer equal to or less than "n". Example: FLOOR(99.8) returns 99.

**ROUND(n, m) or ROUND(n)** Returns "n" rounded to "m" decimal places; "m" defaults to 0.  
Example: ROUND(99.8) returns 100.

### Date and Time Functions

- DATEADD (datepart, n, d)** Returns a date that is "d" plus "n" times "datepart". Example: DATEADD('mm', 6, Creation\_Date) will return a date that is 6 months later than the value stored in the field Creation\_Date.
- DATEDIFF (datepart, d, e)** Returns a signed integer value equal to the number of "datepart" boundaries crossed between "e" and "d". Example: DATEDIFF('yy', Date\_Hired, GETDATE()) will return the number of years past since the employee was hired.
- DATENAME (datepart, d)** Returns a character string representing the specified "datepart" of "d".
- DATEPART (datepart, d)** Returns an integer representing the specified "datepart" of "d".
- GETDATE()** Returns the current system date and time.

### Date Parts

Date part	Abbreviation	Values
year	yy	1753-9999
quarter	qq	1-4
month	mm	1-12 (Jan-Dec)
day of year	dy	1-366
day	dd	1-31
week	wk	1-53
weekday	dw	1-7 (Sun-Sat)

### Conversion Functions

**CONVERT (dt, sd) or CONVERT(dt(n),sd) or CONVERT(dt, sd,style)** Converts the source data "sd" into an expression of the specified data type (dt or dt(n)). An optional style could be specified when converting dates.

### Date Styles

Style without century (yy)	Style with century (yyyy)	Standard	Output
-	0 or 100 (*)	Default	mon dd yyyy hh:miAM (or PM)
1	101	USA	mm/dd/yy
2	102	ANSI	yy.mm.dd
3	103	British/French	dd/mm/yy
4	104	German	dd.mm.yy
5	105	Italian	dd-mm-yy
6	106	-	dd mon yy
7	107	-	mon dd, yy
8	108	-	hh:mm:ss
-	9 or 109(*)	Default + ms	mon dd yyyy hh:mi:ss:mmmAM (or PM)
10	110	USA	mm-dd-yy

11	111	JAPAN	yy/mm/dd
12	112	ISO	yymmdd
13	13 or 113 (*)	Eur. def.t + ms	dd mon yyyy hh:mm:ss:mmm(24h)
14	114		hh:mi:ss:mmm(24h)

\* The default values (style 0 or 100, 9 or 109, and 13 or 113) always return the century (yyyy).