

Task Factory Functions List A-C

Last Modified on 01 October 2020

🔔 Task Factory users running version 2020.1.4 or older (released prior to May 27, 2020): **There's an important Task Factory update.** Please visit [here](#) for more details.

Abort

Description	Stops the dataflow and issues a specified error message to the dataflow log file.	
Syntax	ABORT(String_Message [, Condition])	
Returns	NULL	
Parameters		
Name	Optional	Description
String_Message	false	The message you want to display in the dataflow log file when the dataflow stops. The string can be any length and you can enter any valid transformation expression.
Condition	false	(Optional) The conditional expression in which the data flow should be aborted.
Examples		
Abort("Processing aborted"), Abort("Zero Length Detected In Name", Length(FirstName) = 0)		

Abs

Description	Returns the absolute value of a numeric value
Syntax	ABS([Numeric_Value])
Returns	Positive Numeric value. ABS returns the same datatype as the numeric value passed in as an argument, can be NULL.

Parameters		
Name	Optional	Description
<h2>Abs</h2>		
Numeric_Value	false	Passes the values for which you want to return the absolute values. You can enter any valid transformation expression.

Acos		
Description	Returns the angle whose cosine is the specified number.	
Syntax	ACOS(Numeric_Value)	
Returns	Double value. NULL if a value passed to the function is NULL.	
Parameters		
Name	Optional	Description
Numeric_Value	false	Numeric datatype. A number representing a cosine, where number_value must be greater than or equal to -1, but less than or equal to 1. You can enter any valid task editor.
Examples		
Acos(0.5), Acos(varValue)		

AddToDate		
Description	Returns a specified date with the specified number interval (integer) added to a specified datepart of that date. If you add or subtract value for specified datepart.	
Syntax	AddToDate(Date, Format, Amount)	
Returns	Returns modified date.	
Parameters		

Name	Optional	Description
AddToDate		
Date	false	valid datetime value or expression
Format	false	Valid string format or expression for datepart you want to add/subtract. Supported format values are as below Year Format => Y, YYY, YY, YYYY Month Format => M, MM, MON, MONTH Day Format => D, DD, DDD, DY, DAY Hour Format => H, HH, HH12, HH24 Minute Format => MI, N Second Format => S, SS Milisecond Format => MS
Amount	false	Any valid integer value or expression specifying number of Year, Month, Day, Hour, Minute, Second or Milisecond. If you supply positive value then it will perform add else it will do subtract.
Examples		
<pre>AddToDate(ORDER_DATE, 'DD', 3) //Adds 3 days to ORDER_DATE and return new datetime AddToDate(ORDER_DATE, 'HH', 14) //Adds 14 hours to ORDER_DATE and return new datetime AddToDate(ORDER_DATE, 'MM', -5) //Subtract 5 months from ORDER_DATE and return new datetime AddToDate(ORDER_DATE, 'YYYY', 1) //Add one year to supplied ORDER_DATE and return new datetime</pre>		

AesDecrypt

Description	Returns decrypted data to string format using Advanced Encryption Standard (AES) algorithm with 128-bit encoding.	
Syntax	AesDecrypt(Value, Key)	
Returns	Returns a decrypted binary value that can be NULL if the input value is a null value.	
Parameters		
Name	Optional	Description
Value	false	The Binary datatype you wish to decrypt.

Key	false AesDecrypt	The String of precision 16 characters or fewer that was originally used to encrypt the value. You can use mapping variables for the key.
Examples		
<pre>AesDecrypt(varEncryptedData, "key-value-1234"), AesDecrypt(EncryptedValueColumn, "key-value-1234"), AesDecrypt(FileReadAllText("C:\data\encrypted.txt"), "key-value-1234")</pre>		

<h2>AesEncrypt</h2>		
Description	Returns data in encrypted format using Advanced Encryption Standard (AES) algorithm with 128-bit encoding.	
Syntax	AesEncrypt(Value, Key)	
Returns	Returns an encrypted binary value that can be NULL if the input is a null value.	
Parameters		
Name	Optional	Description
Value	false	The String value you wish to encrypt.
Key	false	The String of precision 16 characters or fewer that will be used to encrypt the value. You can use mapping variables for the key.
Examples		
<pre>AesEncrypt("Encrypt this value", "key-value-1234"), AesEncrypt>PasswordColumn, "key-value-1234"), AesEncrypt(FileReadAllText("C:\data\unencrypted.txt"), "key-value-1234")</pre>		

<h2>Asin</h2>	
Description	Returns the angle whose sine is the specified number..
Syntax	Asin(Numeric_Value)

Returns	Double value. NULL if a value passed to the function is NULL.	
Asin		
Parameters		
Name	Optional	Description
Numeric_Value	false	Numeric datatype. A number representing a sine, where number_value must be greater than or equal to -1, but less than or equal to 1. You can enter any valid task editor.
Examples		
Acos(varValue)		

Atan		
Description	Returns the angle whose tangent is the specified number.	
Syntax	Atan(Numeric_Value)	
Returns	Numeric	
Parameters		
Name	Optional	Description
Numeric_Value	false	Passes the values for which you want to the Atan of
Examples		
Atan(55)		

Avg		
Description	Aggregation that calculates the average of all values. Can only be used as part of the Advanced Aggregation component	
Syntax	Avg(Value [, Condition])	
Returns	Double.	

Parameters		
Name	Optional	Description
Value	false	The value to add to the aggregation
Condition	false	The condition allows you to conditionally add values to the aggregation based on the expression. For instance, you could define the aggregate as Avg(SalesTotal, CustomerState="FL") and this would only add the value to the aggregation if the customers state was florida.
Examples		
Avg(SalesTotal), Avg(SalesTotal, CustomerState="FL")		

BinToDecimal		
Description	Converts a binary value to an decimal value.	
Syntax	BinToDecimal(Value)	
Returns	Decimal	
Parameters		
Name	Optional	Description
Value	false	The binary value to convert.
Examples		
BinToDecimal(10)		

BinToHex	
Description	Converts a binary value to an hexadecimal value.

Syntax	BinToHex(Value)	
Returns	Decimal	
Parameters		
Name	Optional	Description
Value	false	The binary value to convert.
Examples		
BinToHex(10)		

<h2>BinToOct</h2>		
Description	Converts a binary value to an octal value.	
Syntax	BinToOct(Value)	
Returns	Decimal	
Parameters		
Name	Optional	Description
Value	false	The binary value to convert.
Examples		
BinToOct(10)		

<h2>Ceiling</h2>	
Description	Returns the smallest integer that is greater than or equal to the numeric value passed to this function.
Syntax	Ceiling(Numeric_Value)
Returns	Returns an integer if you pass a numeric value with declared precision between 0 and 28, otherwise it returns a Double. NULL values will return a NULL.

Parameters		
Name	Optional	Description
<h2>Ceiling</h2>		
Numeric_Value	false	Must be a numeric datatype. You can enter any valid transformation expression.

<h2>Choose</h2>		
Description	Chooses a string from a list of strings based on a given position. You specify the position and the value. If the value matches the position, the Integration Service returns the value. You can either pass in comma delimited static values or pass in a variable or column with delimited values	
Syntax	Choose(Index, String1 [, String2, ..., StringN]), Choose(Index, Delimeted_String [, Delimeter_Value=","])	
Returns	The string that matches the position of the index value. NULL if no string matches the position of the index value.	
Parameters		
Name	Optional	Description
Index	false	Numeric datatype. Enter a number based on the position of the value you want to match.
String	false	Any character value.
Examples		
Choose(1, "SentryOne, 4001 Yancey Road, Charlotte, NC") - Will return "SentryOne" Choose(1, varPipeDelimetedAddress, " ")		

<h2>Chr</h2>	
Description	CHR returns the ASCII character corresponding to the numeric value you pass to this function. ASCII values fall in the range 0 to 255. You can pass any integer to CHR, but only ASCII codes 32 to 126 are printable characters.
	CHR(Numeric_Value)

Syntax		
Returns	ASCII or Unicode character. A string containing one character. NULL if a value passed to the function is NULL.	
Parameters		
Name	Optional	Description
Numeric_Value	false	Numeric datatype. The value you want to return as an ASCII or Unicode character. You can enter any valid transformation expression.
Examples		
Chr(65), Chr(varCharacterValue)		

ChrCode

Description	ChrCode returns the numeric ASCII value of the first character of the string passed to the function. ASCII values fall in the range 0 to 255.	
Syntax	ChrCode (String)	
Returns	ASCII or Unicode character. A string containing one character. NULL if a value passed to the function is NULL.	
Parameters		
Name	Optional	Description
String	false	Character string. Passes the values you want to return as ASCII or Unicode values. You can enter any valid task editor.
Examples		
ChrCode("A"), ChrCode("?")		

Concat

--	--

Description	Concatenates two strings.	
Syntax	CONCAT(First_String, Second_String)	
Returns	String. NULL if both string values are NULL. Nulls If one of the strings is NULL, CONCAT ignores it and returns the other string. If both strings are NULL, CONCAT returns NULL.	
Parameters		
Name	Optional	Description
First_String	false	Any datatype except Binary. The first part of the string you want to concatenate. You can enter any valid task editor.
Second_String	false	Any datatype except Binary. The second part of the string you want to concatenate. You can enter any valid task editor.
Examples		
Concat("Sentry", " ", "One"), Concat(FirstNameColumn, " ", LastNameColumn)		

ConvertBase

Description	Converts a integer base from one to another.	
Syntax	ConvertBase(Value, From_Base, To_Base)	
Returns	Integer.	
Parameters		
Name	Optional	Description
Value	false	The value to change the base of.
From_Base	false	The existing base of the integer. Valid values are 2, 8, 10, 16
To_Base	false	The new base of the integer. Valid values are 2, 8, 10, 16

Examples

ConvertBase("1234", 10, 2)

ConvertBase

Cos

Description

Returns the cosine of a numeric value (expressed in radians).

Syntax

Cos(Numeric_Value)

Returns

Double value. NULL if a value passed to the function is NULL.

Parameters

Name

Optional

Description

Numeric_Value

false

Numeric datatype. Numeric data expressed in radians (degrees multiplied by pi divided by 180). Passes the values for which you want to calculate a cosine. You can enter any valid task editor.

Examples

Cos(10), Cos(varValue)

Cosh

Description

Returns the hyperbolic cosine of a numeric value (expressed in radians).

Syntax

Cosh(Numeric_Value)

Returns

Double value. NULL if a value passed to the function is NULL.

Parameters

Name

Optional

Description

Numeric datatype. Numeric data expressed in radians (degrees multiplied by pi

Numeric_Value	false	divided by 180). Passes the values for which you want to calculate the hyperbolic cosine. You can enter any valid task editor.
<h2>Cosh</h2>		
Examples		
Cosh(10), Cosh(varValue)		

<h2>Count (*)</h2>		
Description	Aggregation that calculates the total number based on the grouping. Can only be used as part of the Advanced Aggregation component	
Syntax	Count(Value [, Condition])	
Returns	Integer.	
Parameters		
Name	Optional	Description
Value	false	The value to add to the aggregation
Condition	false	The condition allows you to conditionally add values to the aggregation based on the expression. For instance, you could define the aggregate as Count(*, CustomerState="FL") and this would only add the value to the aggregation if the customers state was florida.
Examples		
Count(*), Count(*, CustomerState="FL")		

<h2>Count (column name)</h2>	
Description	Aggregation that calculates the total number based on the grouping. Can only be used as part of the Advanced Aggregation component

Syntax	Count(Column_Name [, Condition])	
Returns	<h2>Count (column name)</h2> <p>Integer.</p>	
Parameters		
Name	Optional	Description
Column_Name	false	The column to retrieve the value of to add to the aggregation
Condition	false	The condition allows you to conditionally add values to the aggregation based on the expression. For instance, you could define the aggregate as Count(CustomerID, CustomerState="FL") and this would only add the value to the aggregation if the customers state was florida.
Examples		
Count(CustomerID), Count(CustomerID, CustomerState="FL")		