

文章

[姚鑫](#) · 三月 3 阅读大约需分钟

第七章 SQL函数 \$JUSTIFY

第七章 SQL函数 \$JUSTIFY

在指定宽度内右对齐值的函数,可以选择舍入到指定的小数位数。

大纲

`$JUSTIFY(expression,width[,decimal])`

参数

- `expression` - 要右对齐的值。它可以是数字值、字符串文字或解析为数字或字符串的表达式。
- `width` - 表达式要在其中右对齐的字符数。正整数或计算结果为正整数的表达式。
- `decimal` - 可选-小数位数。正整数或计算结果为正整数的表达式。将表达式中的小数位数四舍五入或填充到此值。如果指定DECIMAL,会将表达式视为数字。

描述

`$JUSTIFY`返回在指定宽度内右对齐的表达式指定的值。可以包括小数参数以在宽度内小数对齐数字。

- `$JUSTUST(expression,width)`: 双参数语法在宽度内右对齐表达式。它不执行任何表达式转换。该表达式可以是数字字符串,也可以是非数字字符串。
- `$JUSTUST(expression,width,decimal)`: 3参数语法将表达式转换为规范数字,将小数位四舍五入或零填充为小数,然后在宽度内右对齐生成值。如果表达式是非数字字符串或NULL,会将其转换为0,填充,然后右对齐。

`$JUSTIFY`识别当前区域设置的DecimalSeparator字符。

它根据要添加或删除DecimalSeparator字符。

DecimalSeparator字符取决于区域设置;

通常,美式格式的语言环境使用句号(.),欧式格式的语言环境使用逗号(,)。

要为区域设置确定DecimalSeparator字符,请调用以下方法:

```
WRITE ##class(%SYS.NLS.Format).GetFormatItem("DecimalSeparator")
```

如果指定的参数无效,则会发出SQLCODE -380。

如果指定的参数无效,则会发出SQLCODE -381。

`$JUSTIFY`, `ROUND`和`TRUNCATE`

当舍入到固定数目的小数位数非常重要时(例如,表示货币金额时),请使用\$JUSTIFY,它将返回舍入操作后指定数目的尾随零。

当decimal大于表达式中的小数位数时,\$JUSTIFY的零位数。

\$JUSTIFY也右对齐数字,使DecimalSeparator字符在一列数字中对齐。

ROUND也舍入指定数目的小数位数,但它的返回值总是规范的,并删除后面的零。

例如,ROUND(10.004,2)返回10,而不是10.00。

与\$JUSTIFY不同,ROUND允许指定舍入(默认值)或截断。

TRUNCATE截断指定的小数位数。

与ROUND不同的是,如果截断的结果是尾随的零,那么这些尾随的零将被保留。

然而,与\$JUSTIFY不同的是,TRUNCATE不填零。

ROUND和TRUNCATE允许舍入(或截断)到小数分隔符的左边。

例如,ROUND(128.5,-1)返回130。

\$JUSTIFY and LPAD

LPAD的双参数形式和\$JUSTIFY的双参数形式都通过用前导空格填充字符串来对字符串进行右对齐。

这两个参数形式的不同之处是它们如何处理小于输入表达式长度的输出宽度:LPAD截断输入字符串以适应指定的输出长度。

\$JUSTIFY扩展输出长度以适应输入字符串。

如示例所示:

```
SELECT '>' || LPAD(12345,10) || '<' AS lpadplus,
       '>' || $JUSTIFY(12345,10) || '<' AS justifyplus,
       '>' || LPAD(12345,3) || '<' AS lpadminus,
       '>' || $JUSTIFY(12345,3) || '<' AS justifyminus
```

LPAD的三个参数形式允许使用空格以外的字符作为左pad。

参数

expression

右对齐的值,并可选地表示为具有指定小数位数的数字。

- 如果需要对字符串调整,请不要指定decimal。
表达式可以包含任意字符。
\$JUSTIFY右对齐表达式,如width中所述。
- 如果对数字对齐,请指定decimal。
如果指定了decimal,将表达式作为标准数字提供给\$JUSTIFY。
它解决了前面的加减号,并删除了前面和后面的零。
它在第一个非数字字符处截断表达式。
如果表达式以非数字字符(例如货币符号)开头,则将表达式值转换为0。
规范转换不识别NumericGroupSeparator字符、货币符号、多个DecimalSeparator字符或尾随的加号或减号。

在\$JUSTIFY接收到正则数表达式后,\$JUSTIFY执行其操作,将该正则数舍入或置零到小数位数的十进制数,然后右对齐结果,如width中所述。

width

要对转换后的表达式右对齐的宽度。

如果宽度大于表达式的长度(在数字和小数数字转换之后),将右对齐为宽度,并根据需要使用空白的左填充。

如果width小于表达式的长度(在数字和小数转换之后),将width设置为表达式值的长度。

将宽度指定为正整数。

宽度值为0、空字符串("")、NULL或非数字字符串将被视为宽度为0,这意味着将宽度设置为表达式值的长度。

decimal

小数位数的个数。

如果expression包含更多的小数位数,则\$JUSTIFY将小数部分舍入为该小数位数。

如果表达式包含更少的小数位数,\$JUSTIFY将小数部分用0填充到这个小数位数,并添加一个Decimal Separator字符(如果需要)。

如果decimal=0,\$JUSTIFY将表达式舍入为整数并删除decimal Separator字符。

如果表达式值小于1,\$JUSTIFY将在DecimalSeparator字符前插入前导零。

\$DOUBLE值INF、-INF和NAN不受进制值的影响,由\$JUSTIFY返回。

示例

下面的动态SQL示例对字符串执行右对齐。

不执行数值转换:

```
ClassMethod Justify()  
{  
    s myquery = "SELECT TOP 20 Age,$JUSTIFY(Name,18),DOB FROM Sample.Person"  
    s tStatement = ##class(%SQL.Statement).%New()  
    s qStatus = tStatement.%Prepare(myquery)  
    s rset = tStatement.%Execute()  
    d rset.%Display()  
    w !,"End of data"  
}
```

```
DHC-APP>d ##class(PHA.TEST.SQLCommand).Justify()
```

Age	Expression_2	DOB
31	yaoxin	54536
	xiaoli	
8	??	63189
8	??	63189
44	??	50066
	??	
	??	
	Isaacs,Roberta Z.	
44	Chadwick,Zelda S.Chadwick,Zelda S.Chadwick,Zelda S.Chadwick,Zelda S.Chadwick,Zelda S.	50066
89	Fives,James D.	33527
48	Vonnegut,Jose P.	48488
94	Chadbourne,Barb B.	31690
74	Quigley,Barb A.	38960
8	O'Rielly,Chris H.	63189

79 Willeke, Alvin L. 37099
4 Orwell, John V. 64375

下面的动态SQL示例使用指定的小数位数执行数字右对齐:

```

/// d ##class(PHA.TEST.SQLCommand).Justify1()
ClassMethod Justify1()
{
  s myquery = 2
  s myquery(1) = "SELECT TOP 20 $JUSTIFY(Salary,10,2) AS FullSalary,"
  s myquery(2) = "$JUSTIFY(Salary/7,10,2) AS SeventhSalary FROM Sample.Employee"
  s tStatement = ##class(%SQL.Statement).%New()
  s qStatus = tStatement.%Prepare(.myquery)
  s rset = tStatement.%Execute()
  d rset.%Display()
  w !,"End of data"
}

```

```

DHC-APP>d ##class(PHA.TEST.SQLCommand).Justify1()
FullSalary            SeventhSalary
  2662.00              380.29
  21796.00             3113.71
  87539.00             12505.57
  53891.00             7698.71
  53682.00             7668.86
  50822.00             7260.29
  60625.00             8660.71
  49575.00             7082.14
  31914.00             4559.14
  49912.00             7130.29
  36968.00             5281.14
  85770.00             12252.86
  43806.00             6258.00
  19669.00             2809.86
  54810.00             7830.00
  21430.00             3061.43
  74680.00             10668.57
  28570.00             4081.43
  79082.00             11297.43
  58713.00             8387.57

```

20 Rows(s) Affected
End of data

下面的动态SQL示例对指定数量的小数进行数字右对齐,并对相同的数值进行字符串右对齐:

```

ClassMethod Justify2()
{
  s myquery = 2
  s myquery(1) = "SELECT $JUSTIFY({fn ACOS(-1)},8,3) AS ArcCos3,"
  s myquery(2) = "$JUSTIFY({fn ACOS(-1)},8) AS ArcCosAll"
  s tStatement = ##class(%SQL.Statement).%New()
  s qStatus = tStatement.%Prepare(.myquery)
  s rset = tStatement.%Execute()

```

```
    d rset.%Display()  
}
```

```
DHC-APP> d ##class(PHA.TEST.SQLCommand).Justify2()  
ArcCos3 ArcCosAll  
    3.142          3.141592653589793238
```

1 Rows(s) Affected

下面动态SQL示例使用\$DOUBLE值INF和NAN执行数字右对齐:

```
ClassMethod Justify3()  
{  
    d ##class(%SYSTEM.Process).IEEEEError(0)  
    s x = $DOUBLE(1.2e500)  
    s y = x - x  
    s myquery = 2  
    s myquery(1) = "SELECT $JUSTIFY(?,12,2) AS INFtest,"  
    s myquery(2) = "$JUSTIFY(?,12,2) AS NANtest"  
    s tStatement = ##class(%SQL.Statement).%New()  
    s qStatus = tStatement.%Prepare(.myquery)  
    s rset = tStatement.%Execute(x,y)  
    d rset.%Display()  
}
```

```
DHC-APP> d ##class(PHA.TEST.SQLCommand).Justify3()  
INFtest NANtest  
    INF          NAN
```

1 Rows(s) Affected

[#SQL #Caché](#)

源 URL: <https://cn.community.intersystems.com/post/%E7%AC%AC%E4%B8%83%E5%8D%81%E7%AB%A0-sql%E5%87%BD%E6%95%B0-justify>