

SQLCLR函数清单

2013年6月24日
11:38

函数名	参数	返回值	说明
fnpbConcatString	@value [nvarchar] (MAX)	RETURNS [nvarchar] (MAX)	返回字符串串联结果(, 作为分隔符)
fnpbConcatStringEx	@value [nvarchar] (MAX), @split [nvarchar] (20))	RETURNS [nvarchar] (MAX)	返回字符串串联结果(可指定分隔符)
fnpbConcatStringDistinct	@value [nvarchar] (MAX)	RETURNS [nvarchar] (MAX)	返回字符串串联结果(去除重复值)
fnpbConcatStringDistinctEx	@value [nvarchar] (MAX), @split [nvarchar] (20))	RETURNS [nvarchar] (MAX)	返回字符串串联结果(可指定分隔符, 去除重复值)
fnpbConvertStringToTable	@input [nvarchar] (MAX), @splitor [nvarchar] (MAX))	RETURNS TABLE ([Item] [nvarchar] (2000) NULL)	将字符串@input拆分成表(按@splitor拆分)
fnpbConvertStringToTwoFieldTable	@input [nvarchar] (MAX), @splitor [nvarchar] (200), @splitor2 [nvarchar] (200)	RETURNS TABLE ([sField1] [nvarchar] (2000) NULL, [sField2] [nvarchar] (2000) NULL)	将字符串拆分成两列的表 @splitor用于拆分行 @splitor2用于拆分行内列(为空则 =@splitor)
fnpbDecryptionDES	@sDescriptionString [nvarchar] (4000), @sKey [nvarchar] (4000)	RETURNS [nvarchar] (4000)	按Des解密文本
fnpbEncryptionDES	@sEncryptionString [nvarchar] (4000), @sKey [nvarchar] (4000)	RETURNS [nvarchar] (4000)	按Des加密文本
fnpbEncryptionMD5	@sEncryptionString [nvarchar] (4000)	RETURNS [nvarchar] (32)	按MD5加密文本
fnpbConvertToSimplifiedChinese	@sInputString [nvarchar] (max)	RETURNS [nvarchar] (max)	转换为简体
fnpbConvertToTraditionalChinese	@sInputString [nvarchar] (max)	RETURNS [nvarchar] (max)	转换为繁体
fnpbGetPinYinFirstLetter	@sInputString [nvarchar] (50)	RETURNS [nvarchar] (50)	返回文本拼音首字母
fnpbCharCalc_AToZ	@sOldChar [nvarchar] (100), @iLength int	RETURNS [nvarchar] (100)	字符递增的下一字符 (A到Z)
fnpbCharCalc_AToZEx	@sOldChar [nvarchar] (100), @iLength INT, @iLoopCount INT	RETURNS [nvarchar] (100)	字符递增的下一字符 (A到Z, 可指定 递增量@iLoopCount)
fnpbCharCalc_CharOneToNinetySix	@sOldChar [nvarchar] (100), @iLength INT	RETURNS [nvarchar] (100)	字符递增的下一字符 (0到96对应的字 符)
fnpbCharCalc_CharOneToNinetySixEx	@sOldChar [nvarchar] (100), @iLength INT, @iLoopCount INT	RETURNS [nvarchar] (100)	字符递增的下一字符 (0到96对应的字 符, 可指定递增量@iLoopCount)
fnpbCharCalc_ZeroToNine	@sOldChar [nvarchar] (100), @iLength int	RETURNS [nvarchar] (100)	字符递增的下一字符(0-9)
fnpbCharCalc_ZeroToNineEx	@sOldChar [nvarchar] (100), @iLength INT, @iLoopCount INT	RETURNS [nvarchar] (100)	字符递增的下一字符(0-9, 可指定递增 量@iLoopCount)
fnpbCharCalc_ZeroToZ	@sOldChar [nvarchar] (100), @iLength int	RETURNS [nvarchar] (100)	字符递增的下一字符(0-Z)
fnpbCharCalc_ZeroToZEx	@sOldChar [nvarchar] (100), @iLength INT, @iLoopCount INT	RETURNS [nvarchar] (100)	字符递增的下一字符(0-Z, 可指定递增 量@iLoopCount)
fnpbCharCalc_IntToChar	@iValue INT, @sBaseChar NVARCHAR (200)	RETURNS [nvarchar] (1000)	将数字转换成字符
fnpbCharCalc_CharToInt	@sText NVARCHAR (1000), @sBaseChar NVARCHAR (200)	RETURNS INT	将字符转换成数字
fnpbCharCalc_NextChar	@sBaseChar NVARCHAR (200), @sText NVARCHAR (1000), @iReturnLength INT, @iStep INT	RETURNS NVARCHAR (1000)	字符递增的下一字符
sppbPivot	@sSqlText [nvarchar] (MAX), @sFixedFields [nvarchar] (300) @sNameField [nvarchar] (50), @sValueField [nvarchar] (50), @sGroupFunction [nvarchar] (50), @bReturnData bit, @sOrderFields [nvarchar] (50), @sNameSQLText [nvarchar] (MAX), @sNameCaptionField [nvarchar] (MAX), @sNullDefaultValue [nvarchar] (50), @sNameFieldNullValue [nvarchar] (50)		纵转横
fnpbPadLeft	@sInput NVARCHAR (200),	RETURNS NVARCHAR (300)	在字符串前面填充字符到指定位数

	@iTotalWidth INT, @sPadChar NVARCHAR(1)		
fnpbPadRight	@sInput NVARCHAR(200), @iTotalWidth INT, @sPadChar NVARCHAR(1)	RETURNS NVARCHAR(300)	在字符串后面填充字符到指定位数

纵转横参数说明:

```
CREATE PROCEDURE [dbo].[sppbPivot]
(@sSqlText [nvarchar] (MAX) /*提供纵表数据的SQL或者表名*/
,@sFixedFields [nvarchar] (300) /*固定列名, 多列通过逗号分隔*/
,@sNameField [nvarchar] (50) /*名称列, 只能为一列*/
,@sValueField [nvarchar] (50) /*值列, 只能为一列*/
,@sGroupFunction [nvarchar] (50) /*汇总函数, 如果为NULL或者'', SQLCLR默认会不分组, 用max*/
,@bReturnData bit /*是否直接返回纵转横数据, 若为否, 纵转横数据写入##VToH中*/
,@sOrderFields [nvarchar] (50) /*排序列, 用于##VToH数据的排序, 多列通过逗号分隔*/
,@sNameSQLText [nvarchar] (MAX) /*名称列生成语句, 可以包含@sNameField、@sNameCaptionField两列, 也可仅仅只包含@sNameField列*/
,@sNameCaptionField [nvarchar] (MAX) /*名称标题字段, 与sNameField对应显示为转换后的字段名, 可以用, 串联Caption列名, 也可以用SQL返回列*/
,@sNullDefaultValue [nvarchar] (50) /*空值取默认值(一般为0或NULL), SQLCLR默认为NULL*/
,@sNameFieldNullValue [nvarchar] (50) /*名称字段值为空时的默认值, SQLCLR默认为' NULL' */
)

```

```
<?xml version="1.0" encoding="utf-8" ?>
<Sql10jects>
  <SqlItems>
    <SqlItem sName="fnpbConcatString" />
    <SqlItem sName="fnpbConcatStringDistinct" />
    <SqlItem sName="fnpbConvertStringToTable" />
    <SqlItem sName="fnpbDecryptionDES" />
    <SqlItem sName="fnpbEncryptionDES" />
    <SqlItem sName="fnpbEncryptionMD5" />
    <SqlItem sName="fnpbConvertToSimplifiedChinese" />
    <SqlItem sName="fnpbConvertToTraditionalChinese" />
    <SqlItem sName="fnpbGetPinYinFirstLetter" />
    <SqlItem sName="fnpbCharCalc_AToZ" />
    <SqlItem sName="fnpbCharCalc_CharOneToNinetySix" />
    <SqlItem sName="fnpbCharCalc_ZeroToNine" />
    <SqlItem sName="fnpbCharCalc_ZeroToZ" />
    <SqlItem sName="fnpbCharCalc_Compare" />
    <SqlItem sName="fnpbConcatStringDistinctEx" />
    <SqlItem sName="fnpbConcatStringEx" />
    <SqlItem sName="sppbPivot" />
    <SqlItem sName="fnpbConvertStringToTwoFieldTable" />
    <SqlItem sName="fnpbCharCalc_AToZEx" />
    <SqlItem sName="fnpbCharCalc_CharOneToNinetySixEx" />
    <SqlItem sName="fnpbCharCalc_ZeroToNineEx" />
    <SqlItem sName="fnpbCharCalc_ZeroToZEx" />
    <SqlItem sName="fnpbCharCalc_IntToChar" />
    <SqlItem sName="fnpbCharCalc_CharToInt" />
    <SqlItem sName="fnpbCharCalc_NextChar" />
    <SqlItem sName="fnpbPadLeft" />
    <SqlItem sName="fnpbPadRight" />
  </SqlItems>
</Sql10jects>

```