[26] ExecuteSQLNoReturn()

Arguments: (ODBC DSN, User ID, Password, sql1, sql2, sql3, sql4, sql5)

This function executes a SQL statement against any ODBC DSN (even one that is not used by the Crystal report). If the SQL statement is longer than 254 characters, break it into segments across up to 5 sql "segments".

The idea is to Update, Delete, or Insert records as a by product of viewing a Crystal report. A typical use is to update a status column to "eMailed" when a Visual CUT process runs a Crystal report and emails the information (Invoice, Order Confirmation, PO, …).

Returns:
"OK" or an error message if failed.

If the sql statement is not properly constructed (e.g., missing single or double quotes) you may get a "Memory Full" message from Crystal. In such a case, examine and adjust the sql statement.

If you email me a request for a sample report, I will email you an rpt file that demonstrates this functionality. Below are two formulas from that report as examples you can follow:

The formula increments the "Last Year's Sales" column in the Customer table by $1:

```
ExecuteSQLNoReturn ("Xtreme Sample Database","","","Update
""Customer"" SET ""Last Year's Sales"" = ""Last Year's Sales""
+ 1 WHERE TRUE" ,"" ,"" ,"" ,"" )
```

This formula sets Customer 7 Address2 column to its Last Year's Sales:

```
ExecuteSQLNoReturn ("Xtreme Sample Database","","","Update
""Customer"" SET ""Address2"" = ""Last Year's Sales"" WHERE
""Customer ID"" = 7" ,"" ,"" ,"" ,"" )
```

This formula demonstrates using information from the current section in the report:

```
whileprintingrecords;
ExecuteSQLNoReturn ("Vision Offline","SYSDBA","sesame","Update
ENTRY SET PRINTED =1 WHERE ENTRY_ID =" +
cstr({ENTRY.ENTRY_ID},0,'') ,"" ,"" ,"" ,"" )
```

[27] ExecuteSQLReturnValue()

Arguments: (ODBC DSN, User ID, Password, sql1, sql2, sql3, sql4, sql5)

This function executes a SQL statement against any ODBC DSN (even one that is not used by the Crystal report). If the SQL statement is longer than 254 characters, break it into segments across up to 5 sql "segments".

The SQL statement must be of a type that returns a single value rather than a result set. A typical use is to look up or get information from a data source that is not included in the report.

Returns:
A string containing the result, or an error message if failed.

If the sql statement is not properly constructed (e.g., missing single or double quotes) you may get a "Memory Full" message from Crystal. In such a case, examine and adjust the sql statement.

If you email me a request for a sample report, I will email you an rpt file that demonstrates this functionality. Below is a formula from that report as an example you can follow:

The formula returns the number of employees in the EMPLOYEE table (even if that table or even the ODBC data source is not included in the report:

```
ExecuteSQLReturnValue ("Xtreme Sample Database","","","Select
Count(*) from Employee" ,"" ,"" ,"" ,"" )
```

Here's an example that combines both types of SQL Functions:

```
WhilePrintingRecords;
IF ExecuteSQLReturnValue ("Vision
Offline","SYSDBA","myPass","Select PRINTED from ENTRY WHERE
ENTRY_ID ="+cstr({ENTRY.ENTRY_ID},0,'') ,"" ,"" ,"" ,"" ) ="0"

THEN
ExecuteSQLNoReturn ("Vision Offline","SYSDBA","myPass","Update
ENTRY SET PRINTED = 1 WHERE ENTRY_ID
="+cstr({ENTRY.ENTRY_ID},0,'') ,"" ,"" ,"" ,"" )
```

[28] ExecuteSQLReturnDelimited()

Arguments: (ODBC DSN, User ID, Password, Delimiter, sql1, sql2, sql3, sql4, sql5)

This function executes a SQL statement against any ODBC DSN (even one that is not used by the Crystal report). If the SQL statement is longer than 254 characters, break it into segments across up to 5 sql "segments".

The SQL statement can be of a type that returns multiple records. The function takes all the values in the first column of the result set and concatenates them into a single delimited string.

Returns:
A string containing the result, or an error message if failed.

If the sql statement is not properly constructed (e.g., missing brackets, single or double quotes) you may get a "Memory Full" message from Crystal. In such a case, examine and adjust the sql statement.

If you email me a request for a sample report, I will email you an rpt file that demonstrates this functionality. Below is a formula from that report as an example you can follow:

The formula returns the last names of all employees delimited with a semi-colon (";") from the EMPLOYEE table in the Xtreme Sample Database (even if that table or even the ODBC data source is not included in the report:

```
ExecuteSQLReturnDelimited ("Xtreme Sample Database","","",";",
"Select [Last Name] from Employee" ,"" ,"" ,"" ,"" )
```

The result is:

```
Davolio;Fuller;Leverling;Peacock;Buchanan;Suyama;King;Callahan;Dodsworth;Hellstern;
Smith;Patterson;Brid;Martin
```