



Crystal Reports

How to Display Parameter Values on Reports

Overview

This paper describes how to display all the values for multiple-value and ranged parameters on a report.

Contents

INTRODUCTION	2
DISPLAYING PARAMETERS	2
<i>Multiple discrete string values</i>	2
Crystal Syntax.....	2
Basic Syntax	3
<i>Multiple discrete non-string values</i>	3
Crystal Syntax.....	3
Basic Syntax	3
<i>Ranged parameters</i>	4
Crystal Syntax.....	4
Basic Syntax	4

Introduction

One of the most frequent uses of parameters in reports is to create a dynamic record selection formula, so you can specify the type of records you would like to see in the report.

The next logical step is to display the parameter entries on the report (usually in the Report Header) so you know what data is contained in the report. While straightforward for discrete parameters with a single entry, multiple-value and ranged parameters are a bit trickier. Crystal Reports treats multiple-value parameters as real arrays, while ranged-value parameters require that the minimum and maximum values of each range be displayed individually.

This paper describes how to display all the values for multiple-value and ranged parameters on reports so you know what kind of data the report contains.

NOTE	Ranged and multiple-value parameters are only available in Crystal Reports 7 or higher.
-------------	---

Displaying Parameters

To display all entries in multiple-value or ranged parameters, a formula must be created to extract each value entered and then concatenate the values into a string.

NOTE	The maximum output of any formula is 255 characters in Crystal Reports 8.5 and earlier. In Crystal Reports 9 and later, the maximum output of any formula is 64K.
-------------	---

Because the result of the concatenation formula is a string, all non-string data should be converted to text using the **ToText** function. When converting parameter values to text it may be necessary to use format strings to control the appearance of the formula output.

For more information on format strings, search for “ToText Function” on the **Index** tab of the Crystal Reports Online Help. Note that in these examples the formula objects should be set to “Can Grow” since the parameter values are separated by carriage returns.

Multiple discrete string values

For string parameters with multiple discrete values, the **Join** function can be used to concatenate all entries in the parameter. For detailed information, search for “Join Function” on the **Index** tab of the Crystal Reports Online Help.

Crystal Syntax

```
WhilePrintingRecords;
```

```
Join({?Parameter}, Chr(10));
```

Basic Syntax

```
WhilePrintingRecords
```

```
formula = Join({?Parameter}, Chr(10))
```

Multiple discrete non-string values

For non-string parameters with multiple discrete values, the formula must loop through all entries in the parameter and concatenate the values into a single string.

Crystal Syntax

```
WhilePrintingRecords;
```

```
Shared numberVar counter;
```

```
//increments the loop so that all parameter entries can be displayed
```

```
Shared stringVar display;
```

```
//creates a string "running total" so that all entries can be displayed
```

```
for counter := 1 to Count({?Parameter}) do
```

```
(
```

```
display := display + ToText({?Parameter}[counter],0,"") + Chr(10);
```

```
);
```

```
display;
```

Basic Syntax

```
WhilePrintingRecords
```

```
Shared counter As Number
```

```
'increments the loop so that all parameter entries can be displayed
```

```
Shared display As String
```

```
'creates a string "running total" so that all entries can be displayed
```

```
Do While Count({?Parameter}) > counter
```

```
counter = counter + 1
```

```
display = display + ToText({?Parameter}(counter),0,"") + Chr(10)
```

```
Loop
```

```
formula = display
```

Ranged parameters

For parameters that may contain ranged values, the formula must loop through all entries in the parameter to determine the minimum and maximum values of any ranges present then concatenate the values into a single string.

Crystal Syntax

```
WhilePrintingRecords;

Shared numberVar increment;

//increments the loop so that all parameter entries can be displayed

Shared stringVar output;

//creates a string "running total" so that all entries can be displayed

for increment := 1 to Count({?Parameter}) do

(

if Minimum({?Parameter}[increment]) = Maximum({?Parameter}[increment])

then (output := output + ToText(Minimum({?Parameter}[increment]),0,"") +

Chr(10);)

else (output := output + ToText(Minimum({?Parameter}[increment]),0,"") + ' to

' + ToText(Maximum({?Parameter}[increment]),0,"") + Chr(10););

);

output;
```

Basic Syntax

```
WhilePrintingRecords

Shared increment As Number

'increments the loop so that all parameter entries can be displayed

Shared output As String

'creates a string "running total" so that all entries can be displayed

Do While Count({?Parameter}) > increment

increment = increment + 1

If Maximum({?Parameter}(increment)) <> Minimum({?Parameter}(increment))

-
```

```
Then output = output + ToText(Minimum({?Parameter}(increment)),0,"") + " to  
" + ToText(Maximum({?Parameter}(increment)),0,"") + Chr(10) _
```

```
Else output = output + ToText(Minimum({?Parameter}(increment)),0,"") +  
Chr(10)
```

```
Loop
```

```
formula = output
```