

# SHOW/HIDE SECTION BASED ON VALUE

## OVERVIEW

There may be cases where a user wants to show or hide a section of a report based on a specific value. This functionality can be implemented via script within the report. The approach is to test the value and then set the sections "Visible" property to True or False.

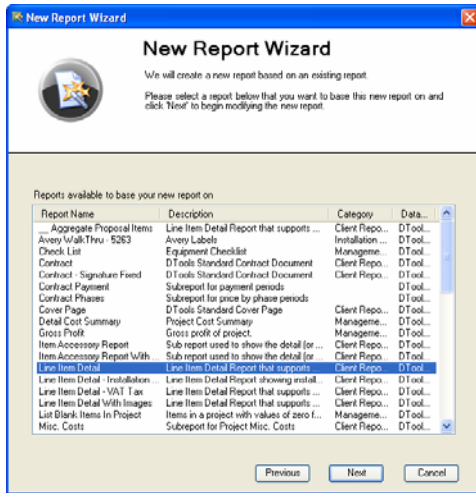
The "ReportUtilities" class contains two methods that will help greatly in accomplishing this process: "ReturnTextBoxValue" and "SetSectionVisibility".

ReturnTextBoxValue returns the value of the "Value" property on a textbox control in a report. This is an excellent way to get the value to test as the section one wishes to control may be a group header or group footer. The textboxes in these sections allow for data summing so one can use the value of the textbox as opposed to writing code to sum values from the recordset data the report is executing against.

For purposes of this discussion, the "LineItem" report will be modified to hide sections where the price is equal to \$0.00. The example shows the basic concepts for showing and hiding sections based on values. Depending on the needs, the user may need to control the visibility of multiple sections to attain their desired results. The same concepts described below can be applied.

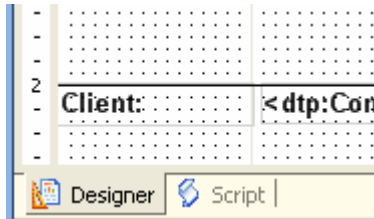
## INSTRUCTIONS

1. Open the Reporting Center
2. Enter the report designer
3. Select File > New from the menu to start the "New Report Wizard"
4. Select "New Report Based on Existing Report". Click Next.
5. Select the "Line Item Detail" report as the template. Click Next.

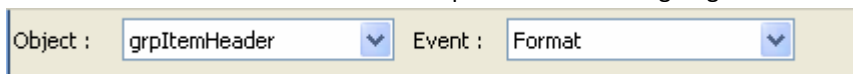


6. Give Your Report a Name. Click Next.

7. Continue Clicking Next until the wizard is complete.
8. Make note of the name of the section to be hidden and the name of the textbox whose value will be used for the test. In this example, the section name is "grpItemHeader" and the textbox controls name is "txtPrice".
9. Click the "Script" tab at the bottom of the designer.



10. At the top of the script window, select the correct section name from the "Object" dropdown and select the "Format" event from the "Event" dropdown. This will highlight that section of the script.



11. Add the following line to the "Format" event of the section.

```
' Return the "Value" from the Price TextBox
Dim val As Object = ReportUtilities.ReturnTextBoxValue(rpt,"grpItemHeader","txtPrice")
```

The parameters of this helper method are defined as such:

The first parameter (rpt) is the instance of the report.

The second is the name of the section the textbox control exists in.

The third is the name of the textbox control.

Add the following code to the script event handler:

```
' If it is not Null, test the value, otherwise, default to true
If val IsNot Nothing Then

    ' surround with try..catch in case the value does not convert.
    try

        ' Convert value to Decimal, if it throws an exception, do nothing
        Dim price As decimal = ctype(val, decimal)

        ' Set the visibility of the section based on whether the value = 0 or not

        ' If the value = 0 then the visibility = false
        ' If the value <> 0 then the visibility = true
        ReportUtilities.SetSectionVisibility(rpt,"grpItemHeader",price <> 0)

    catch

        ' ***** Eat the exception... *****
        ' default to True
        ReportUtilities.SetSectionVisibility(rpt,"grpItemHeader",true)

    End Try

Else

    ' default to True
    ReportUtilities.SetSectionVisibility(rpt,"grpItemHeader",true)

End If
```

The Code in the Script editor should look as such:

```
Sub grpItemHeader_Format

ReportUtilities.SetManufacturerModelTextBox(rpt,"grpItemHeader","txtModel")

' *****
' Show/Hide Section based on value - Begin
' *****

' Return the "Value" from the Price TextBox
Dim val As Object = ReportUtilities.ReturnTextBoxValue(rpt,"grpItemHeader","txtPrice")

' If it is not Null, test the value, otherwise, default to true
If val IsNot Nothing Then

    ' surround with try..catch in case the value does not convert.
    try
        ' Convert value to Decimal, if it throws an exception, do nothing
        Dim price As decimal = ctype(val, decimal)

        ' Set the visibility of the section based on whether the value = 0 or not
        ' If the value = 0 then the visibility = false
        ' If the value <> 0 then the visibility = true
        ReportUtilities.SetSectionVisibility(rpt,"grpItemHeader",price <> 0)

    catch
        ' ***** Eat the exception... *****

        ' default to True
        ReportUtilities.SetSectionVisibility(rpt,"grpItemHeader",true)
    End Try
Else
    ' default to True
    ReportUtilities.SetSectionVisibility(rpt,"grpItemHeader",true)
End If

' *****
' Show/Hide Section based on value - End
' *****
```

The second code snippet does the following:

Tests the value returned from "ReturnTextBoxValue" for null.

If the value is null, the section is set to visible, else the value is converted to the proper type and tested.

If the value does not convert, an exception is thrown and caught. In this demo, the section is simply set to visible.

If the value converts, the returned value from it's test is passed to the "Visible" paramater of the "SetSectionVisibility" method.

12. Click the "Designer" tab at the bottom of the window (like step 9 in reverse).
13. Click the "Save" toolbar
14. Click the "Publish" toolbar.