Set Printer ADDTABLE

⚠ The ADDTABLE message should be used in place of all TABLE and TEXTCOL messages.

Description

The ADDTABLE message is used to print tables and columnar reports.

Remarks

The ADDTABLE message gives you better formatting control than the TABLE message and prints faster. The ADDTABLE message should be used, because it supports all table features and the TABLE and TEXTCOL messages will not be updated or enhanced in future releases.

The ADDTABLE message should be used instead of the TEXTCOL message to print columnar reports. Set the column formatting to No wrapping (~) for all columns to prevent word wrapping.

You can control where the table will split across pages by printing the table one row at a time and checking the position with the POS message and then use the CALCTABLE message to determine the height of the next row.

The Column Format parameter is used to set the column alignment and column width for each column in the table. The number of columns is determined by the number of values in the Column Format parameter. Each column in the Column Format parameter is separated with an @VM. The Column Format for each column must contain the width. The width is in units of Twips. One Inch is equal to 1440 Twips. For example, you can set the width of a column to 2880 to make the column 2 inches wide. You must insert the column alignment before the column width. The column alignment flags are listed in the Comments below. You can combine one Column Alignment flag with one Column Position flag. For example, "~<1440" would set the width to 1 inch and not allow word wrapping, and "_^2880" would set the width to 2 inches and have center alignment with the text in each cell positioned at the bottom.

The Header is printed at the top of the table and is repeated at the top of each page if the table does not fit on the current page. The column headers must be separated with an @VM. The number of Header values should be equal to the number of Column Format values. Use CHAR(13):CHAR(10) to create multiline headers.

The Body is printed with the formatting that is set with the Column Format parameter. The number of columns in the Body should be equal to the number of Column Format values. Rows are separated with an @FM. The Body parameter should not exceed 32 KB in size (use the Append parameter).

The HeaderShade and BodyShade parameters set the colors that are used in the background for the header and body. You should use the RGB function to set these color codes (see Using Colors).

Remarks

The Append parameter allows you to append to the previous table. All of the parameters, except the Body parameter, for the ADDTABLE must be the same for all table pieces that are printed with the Append property set to 1.

The Border parameter sets the border for the entire table. You have to print each row as a separate table to print different borders for each row. See the OIPRINT_EQUATES for a complete description of the border styles.

The Position parameter is used to set the vertical position of the table. The default vertical position is "L" for Left alignment. When you print a table with Left alignment, the text in the first column will line up with the left margin. When you print a table with Right alignment, the last column of the table will line up with the right margin. When you print a table with Center alignment, the table will be vertically centered on the page. The Flush position will print the table with the border of the left column aligned to the left column. The Flush position is useful when you want to print a table with a border, and you want the left border of the table to line up with the left margin. You can also specify the actual horizontal position by passing the horizontal position in the units (the default is inches) established by the INIT message.

Parameters

Parameters are as shown below

Parameter	Description of Fields	Comments	Default

Parm1 - Column Format	<1> Column formatting	Use @VM between column formatting. Column Alignment: < Left > Right ^ Center = Full Column Position: _ Vert bottom + Vert center ~ No wrapping ! Vert line	""
Parm2 - Header	<1> Header text	Use @VM between column headers	""
Parm3 - Body	<1> Table body	@FM between rows and @VM between columns	""
Parm4 - HeaderShade	<1> Shading color for the header	RGB color code for header shading	""
Parm5 - BodyShade	<1> Shading color for the body	RGB color code for body shading	""
Parm6 - Append	<1> Append to the end of previous table	If 1, then the Body is appended to the end of the previous table	0
Parm7 - Format	<1> Border See OIPRINT_EQUATES	Border style for the entire table	0 - None
	<2> Position	L Left R Right C Center F Flush Position	"L"

See also

[TABLE Set_Printer() message](#)

Example

See the OIPI_EXAMPLE2 program that was installed by the setup program for an example on how to use the ADDTABLE message.

This example shows how to use the ADDTABLE message to print a table. The column widths are 1.5 inches, and the first column is left justified with the text vertically centered, the second column is center justified with the text aligned to the bottom, the third column is right justified and prevents word wrapping on this column, and the forth column is full text justified. The body background color is light blue, and the header background color is bright red. The table is printed with borders around every cell. The second call to ADDTABLE is exactly the same as the first, except the Append parameter is set to 1 so the second table is appended to the first.

```
function AddTableTest
declare function Set_Printer, RGB
$insert OIPRINT_EQUATES
x = Set_Printer("INIT", "AddTableTest", "AddTableTest", "", "", 4)
if x < 0 then
call msg("","INIT Error = ":x)
return 0
end
table = "1,2,3,4/"
table := "5,6,7,8/"
table := "9,10,11,12/"
table := "Row Number 4 - Column Number 1,Row Number 4 - Column Number 2,Row Number 4 - Column Number 3,Row
Number 4 - Column Number 4"
convert "," to @vm in table
convert "/" to @fm in table
head = "Column 1":@VM: "Column 2":@VM: "Column 3":@VM: "Column 4"
x = Set_Printer("TEXT", @FM: "ADDTABLE":@FM)
x = Set_Printer("ADDTABLE", "+<2160":@VM: "_^2160":@VM: "~>2160":@VM: "=2160", head, table, RGB(255, 0, 0), RGB
(128, 128, 255), 0, TB_ALL)
* append

x = Set_Printer("ADDTABLE", "+<2160":@VM: "_^2160":@VM: "~>2160":@VM: "=2160", head, table, RGB(255, 0, 0), RGB
(128, 128, 255), 1, TB_ALL)
stat = Set_Printer("TERM")
return 1
```

This example shows how to use the ADDTABLE message to print a columnar report. Each column is 1 inch and the "~" flag is used to prevent word wrapping in all columns. The first two columns are left justified and the last two columns are right justified.

```
function ColumnTableTest
declare function Set_Printer, RGB
$insert OIPRINT_EQUATES
x = Set_Printer("INIT", "ColumnTableTest", "ColumnTableTest", "", "", 4)
if x < 0 then
call msg("","INIT Error = ":x)
return 0
end
table = "1,2,3,4/"
table := "5,6,7,8/"
table := "9,10,11,12/"
table := "Row Number 4 - Column Number 1,Row Number 4 - Column Number 2,Row Number 4 - Column Number 3,Row
Number 4 - Column Number 4"
convert "," to @vm in table
convert "/" to @fm in table
head = "Column 1":@VM: "Column 2":@VM: "Column 3":@VM: "Column 4"
x = Set_Printer("TEXT", @FM: "ADDTABLE":@FM)
x = Set_Printer("ADDTABLE", "~1440":@VM: "~1440":@VM: ">~1440":@VM: ">~1440", head, table, "", "", 0, 0)
stat = Set_Printer("TERM")
return 1
```