Table of Contents

HMG Grid+

HMG Grid+ is a Harbour MiniGUI plugin library useful in printing or exporting two dimensional table data either from an array variable or from a Grid control.

It is provided as is and is free to use, modify, share even for commercial purposes.

Grid+ is a class based (OOPS) approach and it is easy to implement in report designing.

My email id for any suggestion/bug report/comment: srathinagiri@gmail.com

Properties

HMG Grid+ has various properties to configure the reports such as header, footer, font, page numbering etc., To make it easy for the user/developer, default values are set for these properties. It can be changed either at the time of designing by the developer or run time by the user if GUI method is called for. Inside the programs these properties can be used both for setting and retrieving data. For example,

```
oGP := GridPlus():New()
lHLine := oGP:HorizontalLines // for retrieving data
oGP:HorizontalLines := lHLine // for setting data
```

The expected data type of the property is given within bracket.

- C - Character
- N - Numeric
- L - Logical (.t. or .f.)
- A - Array

In Grid+ the Unit of Measurement is millimeters (mm).

Following are the list of Grid+ object Properties:

- **BottomMargin** (N)
  - Sets/Returns the bottom margin of the paper in mm. Default is 15 mm.
  - Usage: `oGP:BottomMargin := 20`
- **ColumnHeaders** (A)
  - Sets/Returns the column headers of the Grid/Array. By default, column headers are taken from the Grid control itself. For Array data, column headers are named by default as Column 1, Column 2, ... etc.,
  - Usage: `oGP:ColumnHeaders := { 'My First Column', 'My Second Column', 'My Third Column' }`
- **ColumnJustify** (A)
  - Sets/Returns the Justification of the data in every column. For array data, except for numeric data which is right justified, all the other columns are left justified. For Grid controls, default column justification from the grid is used.
  - Usage: `oGP:ColumnJustify := { 0, 1, 2, 1, 0 } // 0 for left, 1 for right and 2 for center column justification`
- **ColumnPicture** (A)
  - Sets/Returns the formatting mask used along with **ColumnSum** enabled columns.

The total value of the ColumnSum enabled columns will be formatted with the formatting mask specified for that column. By default ColumnPicture is not enabled. Without this property, if ColumnSum property is used, column totals are shown without any formatting in the report.

- Usage: `oGP:ColumnPicture := { '', '', '999,999,999,999.99' }`

- **ColumnSum** (A)
  - Sets/Returns an array of selected columns for summing up. By default ColumnSum is filled with an empty array.
  - Usage: `oGP:ColumnSum := { 3, 4 } // only 3rd and 4th columns are selected for summary`

- **ColumnWidths** (A)
  - Sets/Returns the ColumnWidths in mm. For array data, the default width is 30 mm per column. For Grid Data it is derived from the Grid Control. For better results, after loading the data into Grid+, the developer can fix the column widths according to the data to be printed. If any column width is entered as 0, that column will be omitted from printing.
  - Usage: `oGP:ColumnWidths := { 25, 30, 25, 50 }`

- **DatePrefix** (C)
  - Sets/Returns the prefix string used in the Date template of the Report Footer. Default string is 'Date :' It is used along with the **IncludeDate** property.
  - Usage: `oGP:DatePrefix := 'My Report Date :'`

- **DateTimeAlignment** (N)
  - Set/Returns the alignment position of the Date/Time template of the Report Footer. Default alignment is 0 (Left).
  - Usage: `oGP:DateTimeAlignment := 2 // 0 for Left, 1 for Right and 2 for Center`

- **DecimalSymbol** (C)
  - Sets/Returns the Decimal Point character used inside the formatted numeric columns. This property is effective if **ColumnSum** property is used. By default the decimal symbol is '.'
  - Usage: `oGP:DecimalSymbol := ','`

- **DefaultColumnWidth** (N)
  - Sets/Returns the default column width in mm for Array data which is 30 mm. This will be effective only if it is used before the array data being loaded inside Grid+.
  - Usage: `oGP:DefaultColumnWidth := 20`

- **GroupColumns** (A)
  - Grid+ is capable of grouping the data up to a maximum of three levels. Group/Sub-group headers are automatically inserted whenever there is a change in the data. It is necessary that before loading the data, it shall be sorted in the order of grouping viz., first, second and third column. The maximum number of entries can be 3.
  - Usage: `oGP:GroupColumns := { 2, 1 } // Second column data are grouped and inside`

```
that first column data are grouped
```

- **GroupIndent** (N)
  - Sets/Returns the Group Header/Sub header indent space. Default is 3.
  - Usage: `oGP:GroupIndent := 5`
- **HorizontalLines** (L)
  - Set/Returns whether Row separation lines are printed or not. By default Grid Lines are enabled.
  - Usage: `oGP:HorizontalLines := .f.`
- **IncludeDate** (L)
  - Sets/Returns whether the current Reporting Date to be printed at the Report Footer. By default, this property is disabled.
  - Usage: `oGP:IncludeDate = .t.`
- **IncludeTime** (L)
  - Sets/Returns whether the current Reporting Time to be printed at the Report Footer. By default, this property is disabled.
  - Usage: `oGP:IncludeTime = .t.`
- **LeftMargin** (N)
  - Sets/Returns the left margin of the paper in mm. Default is 15 mm.
  - Usage: `oGP:LeftMargin := 20`
- **MergeHeaders** (A)
  - Grid+ can be used to merge some of the column headers and additional merged header(s) can be designed. A two dimensional array with the structure of { nColumnFrom, nColumnTo, cMergeHeader } shall be passed on to this property. If the nColumnFrom and nColumnTo are overlapping with other merge headers, they won't be displayed.
  - Usage: `oGP:MergeHeaders := { { 1, 3, 'Sales Details' }, { 4, 6, 'Product Details' } }`
- **Orientation** (N)
  - Sets/Returns the paper orientation. Default orientation is 1 (Portrait). Use 2 for Landscape.
  - Usage: `oGP:Orientation := 2 // Paper orientation is now landscape`
- **PageCenter** (L)
  - The content of the report can be printed in the center of the page if possible. (Left and Right margin are not considered)
  - Usage: `oGP:PageCenter := .t.`
- **PageNumberAlignment** (N)
  - Sets/Returns the Page number display alignment position in the report. Default alignment is 1 (Right) Use 0 - Left, 1 - Right, 2 - Center This property is used along with **PageNumbering**
  - Usage: `oGP:PageNumberAlignment := 0 // left aligned`

- **PageNumbering** (L)
  - Enable/Disable the page numbering option. If disabled page numbers are not printed. By default this property is enabled.
  - Usage: `oGP:PageNumbering := .f.`
- **PageNumberPosition** (N)
  - Sets/Returns the Page number display position in the report. Page number can be displayed as a Header or Footer or Both. By default it is set to be printed in the Header. Use 1 - Header only, 2 - Footer only, 3 - Both Header and Footer. This property is used along with **PageNumbering**
  - Usage: `oGP:PageNumberPosition := 3 // page number enabled in top and bottom of the page`
- **PageNumberPrefix** (C)
  - Sets/Returns the prefix string used in the page number template of the Report Header/Footer/Both. Default string is 'Page No. :'. This property is used along with **PageNumbering**
  - Usage: `oGP:PageNumberPrefix := 'This Page Number: '`
- **PaperHeight** (N)
  - Sets/Returns the printer paper height. Useful only when 'Custom Paper Size' is selected. Paper Height is specified in mm.
  - Usage: `oGP:PaperHeight := 296`
- **PaperSize** (N)
  - Sets/Returns the printer paper size. Any HMG system defined paper size constant can be used.
  - Usage: `oGP:PaperSize := PRINTER_PAPER_A4`
- **PaperWidth** (N)
  - Sets/Returns the printer paper width. Useful only when 'Custom Paper Size' is selected. Paper Width is specified in mm.
  - Usage: `oGP:PaperWidth := 216` *
- **PenWidth** (N)
  - Sets/Returns the Horizontal/Vertical line width in mm. Default value is 0.10 mm
  - Usage: `oGP:PenWidth := 0.05`
- **Preview** (L)
  - Sets/Retuns whether print preview is enabled. Default is .t.
  - Usage: `oGP:preview := .f.`
- **Printer** (C)
  - Sets/Returns the name of the system printer. Default system printer name is selected by default.
  - Usage: `oGP:Printer := 'PDF Creator'`
- **ReportFont** (A)
  - Sets/Returns the main report font specification in HMG Array font format. Default is

Arial, 12 for array data and default Grid font and size for grid data.

- Usage: `oGP:ReportFont := ARRAY FONT 'Courier New' SIZE 10`

- **ReportFooter** (C)
  - Sets/Returns a string to be printed at the end of the Report. By default it is empty.
  - Usage: `oGP:ReportFooter := 'User S. Rathinagiri'`
- **ReportFooterFont** (A)
  - Sets/Returns the Report Footer array font. This property Used along with **ReportFooter** property.
  - Usage: `oGP:ReportFooterFont := ARRAY FONT 'Courier New' SIZE 10 .t.`
- **ReportHeader1** (C)
  - Sets/Returns the report header to be printed in every page. By default it is empty.
  - Usage: `oGP:ReportHeader1 := 'This is header 1'`
- **ReportHeader2** (C)
  - Sets/Returns the report header to be printed in every page. By default it is empty.
  - Usage: `oGP:ReportHeader2 := 'This is header 2'`
- **ReportHeader3** (C)
  - Sets/Returns the report header to be printed in every page. By default it is empty.
  - Usage: `oGP:ReportHeader3 := 'This is header 3'`
- **ReportHeaderFont** (A)
  - Sets/Returns the Report Footer array font. This property Used along with **ReportHeader1/2/3** property.
  - Usage: `oGP:ReportHeaderFont := ARRAY FONT 'Courier New' SIZE 10 .t.`
- **RightMargin** (N)
  - Sets/Returns the right margin of the printer paper. Default is 15 mm.
  - Usage: `oGP:RightMargin := 20`
- **SpreadWhiteSpace** (L)
  - Sets/Returns whether proportionate white spacing is enabled or not. Grid+ has a facility to spread white space inside the cells (if available), proportionate to the column widths to fill the report to the paper width (leaving only left and right margins). By default this property is disabled.
  - Usage: `oGP:SpreadWhiteSpace := .t.`
- **TextMaxRow** (N)
  - Sets/Returns the maximum number of lines to be printed in 'Text' mode exporting. This property is useful only for 'Text' file export. Default is 60.
  - Usage: `oGP:TextMaxRow := 50`
- **TimePrefix** (C)
  - Sets/Returns the prefix string used in the Time template of the Report Footer. Default string is 'Time :' It is used along with the **IncludeTime** property.
  - Usage: `oGP:TimePrefix := 'My Report Time :'`
- **TopMargin** (N)

- Sets/Returns the top margin of the printer paper. Default is 15 mm.
- Usage: `oGP:TopMargin := 20`

- **TruncateColumns** (L)
  - Grid+ spreads the content of the report into tiles (new paper) if the all the columns could not be fit in a single page. If this option is to be disabled, then those columns which are beyond the right margin will be truncated. By default this option is disabled (.f.).
  - Usage: `oGP:TruncateColumns := .t.`

- **VerticalLines** (L)
  - Set/Returns whether Column separation lines are printed or not. By default Grid Lines are enabled.
  - Usage: `oGP:VerticalLines := .f.`

- **WordWrap** (L)
  - Sets/Returns whether cell content is wrapped to a next line if it doesn't fit to be printed within the cell width. By default this option is disabled. If disabled, the cell content will be truncated beyond the printable area.
  - Usage: `oGP:WordWrap := .t.`

Methods

Grid+ class has the following methods which can be called by the developer inside his HMG programs.

- **AddArrayData(aArray, [lFormatted])**
  - This method is used to load two dimensional array data into Grid+ for printing/exporting. The array may contain any type of data. If it is already having data formatted into strings, you can use the second parameter (lFormatted) for speeding up. Default lFormatted is .f. First parameter for this method is compulsory.
  - Usage : `oGP:AddArrayData( aData, .t. )`
- **AddGridData(cForm, cGrid)**
  - This method is used to load HMG Grid control data into Grid+. This method is not suitable for Virtual Grids. For virtual grids you can use the **AddArrayData()** method to load the data directly from the source array. This method requires two parameters viz., FormName and the Grid Control name.
  - Usage: `` `oGP:AddGridData( 'form1`, 'grid1' ) ``
- **Export2CSV([cFileName], [lHeaderRow])**
  - After loading data by either **AddArrayData()** or **AddGridData()** method, this method can be used to export the data in CSV (Comma Separated Value) format. If cFileName is not specified, a Save File Dialogue will be opened. lHeaderRow is an optional parameter to specify whether to include the column headers as a CSV row or not. CSV files are useful to quickly export grid/array data into Spreadsheet software like MS Excel/LibreOffice Calc. It is to be noted that ColumnSum, MergeHeaders, Report Header, Footer, Grouping etc don't have any effect in using this method. This method will export only raw data.
  - Usage: `oGP:Export2CSV( 'exportedfile.csv', .t. )`
- **Export2HTML([cFileName])**
  - After loading data by either **AddArrayData()** or **AddGridData()** method, this method can be used to export the data in HTML format. If cFileName is not specified, a Save File Dialogue will be opened. PageNumbering doesn't have any effect on using this method as whole HTML file content is treated as a single page. Other than that, Grouping, MergeHeaders, ColumnSum, Headers and Footers can be used with this method.
  - Usage: `oGP:Export2HTML( 'purchasereport.html' )`
- **Export2TXT([cFileName])**
  - After loading data by either **AddArrayData()** or **AddGridData()** method, this method can be used to print the data in good old plain text format. This method also doesn't consider MergeHeaders, Grouping, ColumnSum.

- Usage: `oGP:Export2TXT( 'Report.txt' )`
- **New()**
    - This is the Grid+ initialization method which is to be called before calling any method or using any property. This method doesn't need any parameter.
    - Usage: `oGP := GridPlus():New()`
- **PrintGrid()**
    - After loading data by either **AddArrayData()** or **AddGridData()** method, this method can be used to directly print data to the System Printer after considering various other properties. Most of the Grid+ properties are used by this method. By controlling those parameters, Grid+ can be used as a versatile Report Designer.
    - Usage: `oGP:PrintGrid()`
- **ShowGUI()**
    - After loading data by either **AddArrayData()** or **AddGridData()** method, this method allows to show a GUI report designing window to configure all the Grid+ properties in an intuitive way by the user himself.
    - Usage: `oGP:ShowGUI()`