**http://phpgrid.com/example/**

[**Introduction**](http://phpgrid.com/example/introduction/)

by [Richard](http://phpgrid.com/author/richard/) on 12:37 pm in [Examples](http://phpgrid.com/example/)

If you are new to phpGrid, we recommend that you go through the examples in the sequence provided for each section.

Before you begin, make sure that you have [installed and configured](http://phpgrid.com/documentation/installation/) phpGrid properly.

http://phpgrid.com/documentation/installation/

phpGrid utilizes [jQuery](http://www.jquery.com/) extensively, the popular cross-browser Javascript framework. The current phpGrid supports jQuery version 1.9.0. If you are already using jQuery but a different version, considering removing the existing jQuery to avoid version conflicts. If you have questions regarding this, please send us an [email](http://phpgrid.com/contact/).

Please note, certain features, such as **update, delete, and add new are disabled in online demo for security reason**. In addition to the examples, major phpGrid features can be seen in [Screenshots](http://phpgrid.com/screenshots/) page.

[Comments ( 0 )](http://phpgrid.com/example/introduction/#respond) [**A Basic PHP Datagrid**](http://phpgrid.com/example/example-1-a-basic-php-datagrid-2/)

by [Richard](http://phpgrid.com/author/richard/) on 12:40 pm in [Examples](http://phpgrid.com/example/)

The basic PHP datagrid requires only as little as **TWO** lines of code. First of all, always [create phpGrid object](http://phpgrid.com/documentation/construct-our-phpgrid/) in the first line; then call its methods to define properties, and lastly always call [display()](http://phpgrid.com/documentation/display/) to render to screen.

.

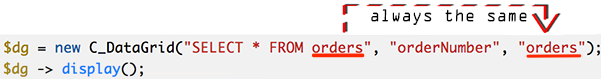
Please note in some databases, such as Firebird, MS Access, the fields name are case-sensitive. Make sure the name used in the code matches the case when you are using those type of databases.

.

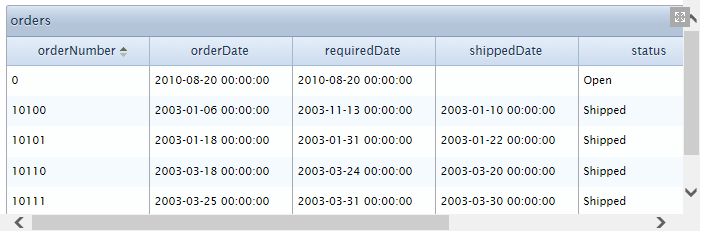
* parameter 1: SQL statement
* parameter 2: SQL primary key
* parameter 3: SQl table name

Do NOT include WHERE clause in $sql. Instead use [set\_query\_filter](http://phpgrid.com/documentation/set_query_filterwhere/) method (version 4.2+) to set query filter.

Note that the 3rd parameter is ALWAYS equals to the database table name, not alias or any arbitrary name (see below).



[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/basic_phpgrid.php?iframe=true&width=1040&height=700)

****

**Programming Tip: Database Naming Standards**

It’s generally not recommended to have dash or space character in naming database tables, or you must add surrounding quotes, or backticks to the table names or columns that have those characters in the query.

Underscore character, on the other hand, is generally accepted for naming convention.

Example:

|  |  |
| --- | --- |
| 1 2 | SELECT \* FROM `my-table` -- requires ` around table name SELECT \* FROM my\_table  -- does not require surrounding quotes. |

[Comments ( 0 )](http://phpgrid.com/example/example-1-a-basic-php-datagrid-2/#respond)

[**Descriptive Column Headers**](http://phpgrid.com/example/example-descriptive-column-headers/)

by [Richard](http://phpgrid.com/author/richard/) on 5:09 pm in [Examples](http://phpgrid.com/example/)

By default phpGrid pulls out the data field name defined in database as the header for each column. You can simply change the title using method [set\_col\_title()](http://phpgrid.com/documentation/set_col_title/).

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");  // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");  $dg -> display(); |

[Comments ( 0 )](http://phpgrid.com/example/example-descriptive-column-headers/#respond)

[**Hide Column**](http://phpgrid.com/example/hide-column/)

by [Richard](http://phpgrid.com/author/richard/) on 5:13 pm in [Examples](http://phpgrid.com/example/)

For columns need not to be shown, such as primary key, use method [set\_col\_hidden()](http://phpgrid.com/documentation/set_col_hidden/) to hide those columns.

**Very Important:** The data are still sent to web browser but hidden using CSS display:none. For sensitive data such as passwords and SSN etc, do not use this method, instead do not include those fields in your SQL at all.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");   $dg -> display(); |

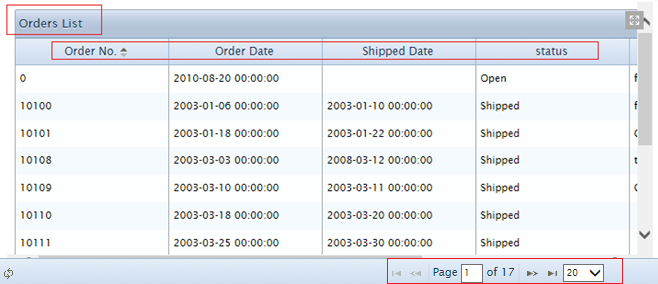
[Comments ( 0 )](http://phpgrid.com/example/hide-column/#respond)

[**Datagrid Caption**](http://phpgrid.com/example/datagrid-caption/)

by [Richard](http://phpgrid.com/author/richard/) on 5:16 pm in [Examples](http://phpgrid.com/example/)

By default, phpGrid displays table name as the caption. It can be changed using [set\_caption()](http://phpgrid.com/documentation/set_caption/). You can also hide the caption by using an empty string “” as the caption.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // change default caption $dg -> set\_caption("Orders List");   $dg -> display(); |

[Comments ( 0 )](http://phpgrid.com/example/datagrid-caption/#respond)

[**Excel, HTML, PDF, or CSV Export**](http://phpgrid.com/example/export-datagrid-to-excel-or-html/)

by [Richard](http://phpgrid.com/author/richard/) on 5:18 pm in [Examples](http://phpgrid.com/example/)

phpGrid currently supports export in native Excel format, CSV, PDF, and HTML format. When the export feature is enabled, phpGrid displays an export icon in the footer. Also see [enable\_export()](http://phpgrid.com/documentation/enable_export/).

**PDF and CSV formats are now supported** (Available to **Professional** License and up) !

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");  // hide a column $dg->set\_col\_hidden("requiredDate");  // change default caption $dg->set\_caption("Orders List");  // EXCEL / PDF / CSV  $dg->enable\_export('EXCEL');  $dg->display(); |

Comments ( 0 )[**Integrated Search**](http://phpgrid.com/example/integrated-search/)

by [Richard](http://phpgrid.com/author/richard/) on 5:20 pm in [Examples](http://phpgrid.com/example/)

phpGrid includes integrated search. By default, this feature is not enabled. To enable search use [enable\_search()](http://phpgrid.com/documentation/enable_search/) method with parameter set to true. Once enabled, the integrated search can be toggled with the search button on the footer.

Notice the “status” is automatically rendered as a drop-down in the integrated search(v5.5+). Talking about making your life easy? :)

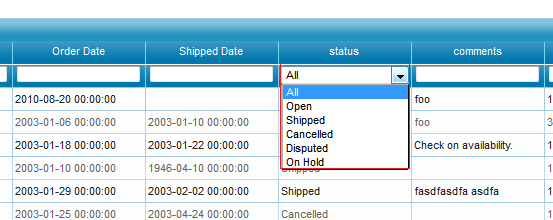
To always display the search toolbar, please see [Always display integrated search toolbar](http://phpgrid.uservoice.com/knowledgebase/articles/157294-always-display-integrated-search-toolbar-when-page) on KB.

Insert the following at the end of the page before ending body tag </body>.

<script type="text/javascript">  
$(function() {  
    var grid = jQuery("#TABLE\_NAME");  
    grid[0].toggleToolbar();  
});  
</script>

Replace the "TABLE\_NAME" with your own

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM Orders", "orderNumber", "Orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // change default caption $dg -> set\_captin("Orders List");  // set export type and edit select type  $dg -> enable\_export('EXCEL');  $dg -> enable\_edit('FORM', 'CRUD'); $dg -> set\_col\_edittype('status', 'select',  'Open:Open;  Shipped:Shipped;  Cancelled:Cancelled;  Disputed:Disputed;  On Hold:On Hold');  // enable integrated search $dg -> enable\_search(true);   $dg -> display(); |

[Comments ( 0 )](http://phpgrid.com/example/integrated-search/#respond)

[**Set Height and Width**](http://phpgrid.com/example/set-height-and-width/)

by [Richard](http://phpgrid.com/author/richard/) on 5:22 pm in [Examples](http://phpgrid.com/example/)

Use [set\_dimension()](http://phpgrid.com/documentation/set_dimension/) method to specify data grid initial height and width. The default height and width is 400 by 300 pixels. In this example, the width is set to 800 pixel, and height is set to 600 pixel.

Wait, It [looks funny](http://phpdatagrid.com/examples/set_dimension.php)! The datagrid height is taller than the space used by the number of rows in a page. To fix this, we can set a bigger pagination to accommodate a taller grid. See the next example on pagination.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // change default caption $dg -> set\_caption("Orders List");  // set export type $dg -> enable\_export('EXCEL');  // enable integrated search $dg -> enable\_search(true);  // set height and weight of datagrid $dg -> set\_dimension(800, 600);    $dg -> display(); |

[Comments ( 0 )](http://phpgrid.com/example/set-height-and-width/#respond)

[**Datagrid Pagination**](http://phpgrid.com/example/set-pagination/)

by [Richard](http://phpgrid.com/author/richard/) on 5:24 pm in [Examples](http://phpgrid.com/example/)

By default, the pagination size is 20. It means maximum of 20 records can be displayed in a page. This value can be easily changed with [set\_pagesize()](http://phpgrid.com/documentation/set_pagesize/) method. Be aware that pagination is disabled when [set\_scroll()](http://phpgrid.com/documentation/set_scroll/) is set to true. See the next example on set\_scroll() for more information.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // change default caption $dg -> set\_caption("Orders List");  // set export type $dg -> enable\_export('EXCEL');  // enable integrated search $dg -> enable\_search(true);  // set height and weight of datagrid $dg -> set\_dimension(800, 600);   // increase pagination size to 40 from default 20 $dg -> set\_pagesize(40);   $dg -> display(); |

[Comments ( 0 )](http://phpgrid.com/example/set-pagination/#respond)

[**Advanced Search**](http://phpgrid.com/example/advanced-search/)

by [Richard](http://phpgrid.com/author/richard/) on 5:25 pm in [Examples](http://phpgrid.com/example/)

phpGrid also includes advanced search. To enable search use [enable\_advanced\_search()](http://phpgrid.com/documentation/enable_advanced_search/) method with parameter set to true. Once enabled, the advanced search can be toggled with the advanced search button on the footer.

The advanced search can search on multiple fields simultaneously with different comparison operators.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->enable\_advanced\_search(true); $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/advanced_search.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/advanced-search/#respond)

[**Load Data with Vertical Scroll**](http://phpgrid.com/example/load-data-with-vertical-scroll/)

by [Richard](http://phpgrid.com/author/richard/) on 5:26 pm in [Examples](http://phpgrid.com/example/)

In contrast to traditional pagination method to browse through data, you can use vertical scroll as alternative way to load data. It is done by set parameter in [set\_scroll()](http://phpgrid.com/documentation/set_scroll/) to true. When scroll position changes, phpGrid makes ajax call in the background and refresh the content in the grid. As a result, pagination is disabled automatically when this method is used.

Note that it does not load everything at once and only hold all the items from the start through to the latest point ever visited. This improves page loading time while prevents memory leaks from happening.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // change default caption $dg -> set\_caption("Orders List");  // set export type $dg -> enable\_export('EXCEL');  // enable integrated search $dg -> enable\_search(true);  // set height and weight of datagrid $dg -> set\_dimension(800, 600);   // increase pagination size to 40 from default 20 $dg -> set\_pagesize(40);  // use vertical scroll to load data $dg -> set\_scroll(true);   $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/vertical_scroll.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/load-data-with-vertical-scroll/#respond)

[**Theme Roller \***](http://phpgrid.com/example/theme-roller/)

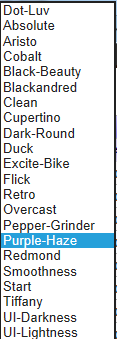
by [Richard](http://phpgrid.com/author/richard/) on 4:31 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*The overall look and feel of the phpGrid can be easily changed using [set\_theme()](http://phpgrid.com/documentation/set_theme/) method. The theme elements are completely customizable through CSS. Theme files, css and images, are stored in css folder.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // change default caption $dg -> set\_caption("Orders List");  // set export type $dg -> enable\_export('EXCEL');  // enable integrated search $dg -> enable\_search(true);  // set height and weight of datagrid $dg -> set\_dimension(800, 600);   // increase pagination size to 40 from default 20 // $dg -> set\_pagesize(40);  // use vertical scroll to load data $dg -> set\_scroll(true);  // change theme $dg -> set\_theme('dot-luv');   $dg -> display(); |

Premium Themes are now available to Enterprise and Universal Licenses customers

* **Aristo**
* **Cobalt**

[Comments ( 0 )](http://phpgrid.com/example/theme-roller/#respond)

[**Display Hyperlink**](http://phpgrid.com/example/display-hyperlink/)

by [Richard](http://phpgrid.com/author/richard/) on 4:36 pm in [Examples](http://phpgrid.com/example/)

You can display the text in column as static hyperlink by calling [set\_col\_link()](http://phpgrid.com/documentation/set_col_link/) and passing the column name. Note this method only display static URL stored in database field. To display dynamic URL links based in variables, see the next example on [set\_col\_dynalink()](http://phpgrid.com/documentation/set_col_dynalink/) method.

Please note we are using Products table in this example.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from products", "productCode", "productcode"); $dg -> set\_col\_title("productCode", "Product Code"); $dg -> set\_col\_title("productName", "Product Name"); $dg -> set\_col\_title("productLine", "Product Line");  // display static Url $dg -> set\_col\_link("productUrl");                                                                                                                                    $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/display_hyperlink.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/display-hyperlink/#respond)

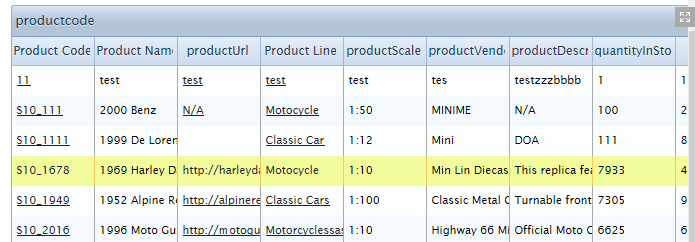
[**Display Dynamic URL**](http://phpgrid.com/example/display-dynamic-url/)

by [Richard](http://phpgrid.com/author/richard/) on 4:42 pm in [Examples](http://phpgrid.com/example/)

From example above, we learn that phpGrid can display simple, static URL using [set\_col\_link()](http://phpgrid.com/documentation/set_col_link/) method. However, it is often for database drive webpage that URL is dynamically formed based on parameters value. Method [set\_col\_dynalink()](http://phpgrid.com/documentation/set_col_dynalink/) is designed specially for this purpose.

The following demonstrates passing the column **orderNumber** as the dynamic value used as part of the URL.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from products", "productCode", "productCode"); $dg -> set\_col\_title("productCode", "Product Code"); $dg -> set\_col\_title("productName", "Product Name"); $dg -> set\_col\_title("productLine", "Product Line");            // display static Url $dg -> set\_col\_link("productUrl");                     // display dynamic url. e.g.http://www.example.com/?productCode=101&foo=bar $dg -> set\_col\_dynalink("productCode", "http://www.example.com/", "productCode", '&foo=bar');                                                                     $dg -> display(); |

[Comments ( 0 )](http://phpgrid.com/example/display-dynamic-url/#respond)

[**Data Format Display**](http://phpgrid.com/example/data-format-display/)

by [Richard](http://phpgrid.com/author/richard/) on 4:45 pm in [Examples](http://phpgrid.com/example/)

phpGrid comes with several frequently used data formatter through [set\_col\_format()](http://phpgrid.com/documentation/set_col_format/) method. Different data format has different format options. In most cases, users only need to set column formatting for “integer”, “number”, and “mail” using this method. For “currency’,”link”, and “showlink” formats, phpGrid provides a number of helper functions to make formatting simpler and easier. Refer to [documentations](http://phpgrid.com/documentation/) on column formatter helper methods.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from products", "productCode", "productCode"); $dg -> set\_col\_title("productCode", "Product Code"); $dg -> set\_col\_title("productName", "Product Name"); $dg -> set\_col\_title("productLine", "Product Line");  // display static Url $dg -> set\_col\_link("productUrl");                                               // dynamic url. e.g.http://www.example.com/?productCode=101&foo=bar $dg -> set\_col\_dynalink("productCode", "http://www.example.com/", "productCode", '&foo=bar');  // column format $dg -> set\_col\_currency("buyPrice"); $dg -> set\_col\_format("quantityInStock", "integer", [array](http://www.php.net/array)("thousandsSeparator" => ",","defaultValue" => "0"));   $dg -> set\_col\_format("MSRP", 'currency', [array](http://www.php.net/array)("prefix" => "$",                                                 "suffix" => '',                                                 "thousandsSeparator" => ",",                                                 "decimalSeparator" => ".",                                                 "decimalPlaces" => '2',                                                 "defaultValue" => '0.00'));  // the above line is equivalent to the following helper function                         $dg -> set\_col\_currency("MSRP", "$", '', ",",".", "2", "0.00");                                                                                         $dg -> display(); |



Didn’t see a formatter you need? You can also implement your own custom formatter such as phone number and SSN. Here’s a [custom formatter sample code](https://phpgrid.uservoice.com/knowledgebase/articles/252146-format-unix-timestamp-to-human-readable-time-forma).

[Comments ( 0 )](http://phpgrid.com/example/data-format-display/#respond) For UNIX timestamp, the **set\_col\_edittype** will not work due to the data format. A workaround is to use custom formatter javascript with a little bit "hack":  
  
The sample code below, last\_update is the the database table field in UNIX timestamp type. dateFormatter is the javascript function to convert UNIX timestamp to readable date time. Must surround the javascript function name with ### to be later escaped by phpGrid internal code.  
  
<?php  
// add your own code here....  
$dg->set\_col\_property('last\_update', array('formatter'=>'###**dateFormatter**###')); // must have ###

// more code here....  
 ?>

<script>

    function **dateFormatter** (cellValue, options, rowdata)

    {

        if (cellValue==0){

            return 'N/A';

        }

        var time = new Date();

        time.setTime (cellValue \* 1000);

        var day = time.getDate();

        var month = time.getMonth()+1;

        var year = time.getFullYear();

        var hours = time.getMinutes();

        var minutes = time.getMinutes();

        var niceDate = year+'/'+month+'/'+day;

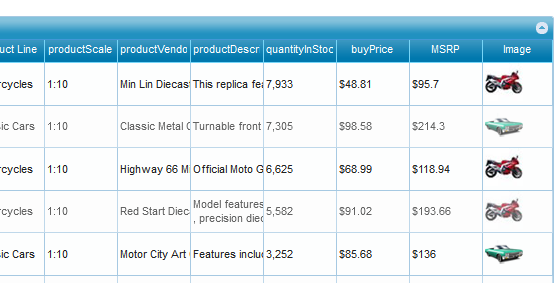
        return niceDate;

    }

</script>

[**Image Display**](http://phpgrid.com/example/image-display/)

by [Richard](http://phpgrid.com/author/richard/) on 4:49 pm in [Examples](http://phpgrid.com/example/)



Display image is easy in phpGrid. Simply set the column name in [set\_col\_img()](http://phpgrid.com/documentation/set_col_img/), phpGrid will take care of the rest.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from products", "productCode", "productCode"); $dg -> set\_col\_title("productCode", "Product Code"); $dg -> set\_col\_title("productName", "Product Name"); $dg -> set\_col\_title("productLine", "Product Line");  // display static Url $dg -> set\_col\_link("productUrl");                                               // dynamic url. e.g.http://www.example.com/?productCode=101&foo=bar $dg -> set\_col\_dynalink("productCode", "http://www.example.com/", "productCode", '&foo=bar');  // column format $dg -> set\_col\_currency("buyPrice"); $dg -> set\_col\_format("quantityInStock", "integer", [array](http://www.php.net/array)("thousandsSeparator" => ",",                                                           "defaultValue" => "0"));   $dg -> set\_col\_format("MSRP", 'currency', [array](http://www.php.net/array)("prefix" => "$",                                                 "suffix" => '',                                                 "thousandsSeparator" => ",",                                                 "decimalSeparator" => ".",                                                 "decimalPlaces" => '2',                                                 "defaultValue" => '0.00'));  // the above line is equivalent to the following helper function                         $dg -> set\_col\_currency("MSRP", "$", '', ",",".", "2", "0.00");                                        // display image $dg -> set\_col\_img("Image");                                                   $dg -> display(); |

You can also further adjust image display options such as width, height and border with CSS using [set\_col\_property](http://phpgrid.com/documentation/set_col_property/) method. The CSS will only apply the specific column.

|  |  |
| --- | --- |
|  | <[style](http://december.com/html/4/element/style.html)>     .image\_css img{width: 40pt;border:1px solid black;} </[style](http://december.com/html/4/element/style.html)> |
|  | $dg->set\_col\_property('Image', [array](http://www.php.net/array)("classes"=>'image\_css'); |

Alternatively, you [resize images for the entire grid with CSS](http://phpgrid.uservoice.com/knowledgebase/articles/253453).

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/image_display.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/image-display/#respond)

[**Interactive Grid Resizing**](http://phpgrid.com/example/resize-datagrid-with-the-mouse/)

by [Richard](http://phpgrid.com/author/richard/) on 4:53 pm in [Examples](http://phpgrid.com/example/)

When enabled, the small triangle icon is displayed at the right bottom corner of the php data grid. The grid can be resized by simply click and drag the icon using mouse. This is more convenient than setting the width and height in the code.

*This function is currently in beta.*

|  |  |
| --- | --- |
|  | $dg= new C\_DataGrid("select \* from products", "productCode", "productCode"); $dg ->set\_col\_title("productCode", "Product Code"); $dg ->set\_col\_title("productName", "Product Name"); $dg ->set\_col\_title("productLine", "Product Line");  // display static Url $dg ->set\_col\_link("productUrl");                                               // dynamic url. e.g.http://www.example.com/?productCode=101&foo=bar $dg -> set\_col\_dynalink("productCode", "http://www.example.com/", "productCode", '&foo=bar');  // column format $dg -> set\_col\_currency("buyPrice"); $dg -> set\_col\_format("quantityInStock", "integer", [array](http://www.php.net/array)("thousandsSeparator" => ",",                                                           "defaultValue" => "0"));    $dg -> set\_col\_format("MSRP", 'currency', [array](http://www.php.net/array)("prefix" => "$",                                                 "suffix" => '',                                                 "thousandsSeparator" => ",",                                                 "decimalSeparator" => ".",                                                 "decimalPlaces" => '2',                                                 "defaultValue" => '0.00'));  // the above line is equivalent to the following helper function                         $dg -> set\_col\_currency("MSRP", "$", '', ",",".", "2", "0.00");                                        // display image $dg -> set\_col\_img("Image");  // enable resize by dragging mouse $dg -> enable\_resize(true);                                                    $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/resize_by_mouse.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/resize-datagrid-with-the-mouse/#respond)

[**Editable Datagrid \***](http://phpgrid.com/example/edit-datagrid/)

by [Richard](http://phpgrid.com/author/richard/) on 4:58 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*

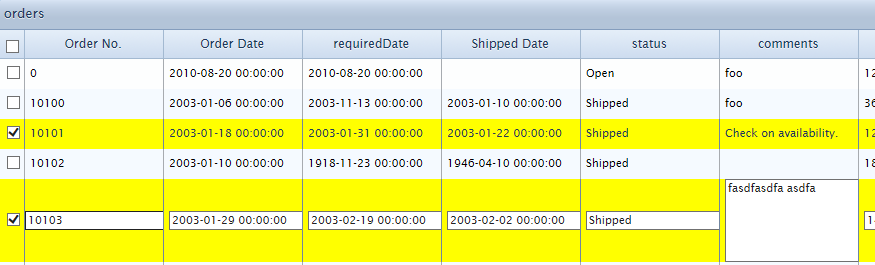
The PHP datagrid is not editable by default. You can enable edit by simply calling [enable\_edit()](http://phpgrid.com/documentation/enable_edit/). Currently, two types of edit modes are supported, **FORM** edit and **INLINE**. When Form edit mode is used, additional icons appear in the data grid footer for editing. When set to inline mode, the cells in the selected row become editable, and **must press Enter key to save changes**. Only a single row can be edited for either mode.

When edit is enabled in a grid, all of the CRUD operations, Create, Read, Update, and Delete, are supported by default. You can restrict types of edit operations permitted by specifying the second parameter in [enable\_edit()](http://phpgrid.com/documentation/enable_edit/). For example, to disallow Delete, change it to *$operations=’CRU’*.

Please note save is disabled in the online demo for security reason.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // enable edit $dg -> enable\_edit("FORM", "CRUD");   $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/enable_edit_datagrid.php?iframe=true&width=1040&height=700)



**Update:**Beginning version 4.4, you can double click to edit whenever edit mode is enabled.  
The toolbar can also be moved to top with a little javascript. See KB “[Move Toolbar to Top of the Grid](http://phpgrid.uservoice.com/knowledgebase/articles/174744)“.

[Comments ( 0 )](http://phpgrid.com/example/edit-datagrid/#respond)

[**Read only fields**](http://phpgrid.com/example/read-only-fields/)

by [Richard](http://phpgrid.com/author/richard/) on 4:59 pm in [Examples](http://phpgrid.com/example/)

Sometimes we don’t want certain fields to be editable, then we can use [set\_col\_readonly()](http://phpgrid.com/documentation/set_col_readonly/) method.

Note that you can set more than one columns to be read only using this method.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM Orders", "orderNumber", "Orders");   // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");  // enable edit $dg ->enable\_edit("INLINE", "CRUD");    // hide a column $dg -> set\_col\_hidden("requiredDate");  // read only columns, one or more columns delimited by comma $dg -> set\_col\_readonly("orderDate, customerNumber");    $dg -> display(); |

Click on row to see the fields that are set to read only.

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/readonly_fields.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/read-only-fields/#respond)

[**Required fields**](http://phpgrid.com/example/required-fields/)

by [Richard](http://phpgrid.com/author/richard/) on 5:02 pm in [Examples](http://phpgrid.com/example/)

To set fields as required in an editable grid, use [set\_col\_required()](http://phpgrid.com/documentation/set_col_required/) method. If required fields are left blank, an “\* required” message will be displayed. One more more columns can be specified as required fields in this method.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM Orders", "orderNumber", "Orders");   // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");   // enable edit $dg -> enable\_edit("INLINE", "CRUD");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // read only columns, one or more columns delimited by comma $dg -> set\_col\_readonly("orderDate, customerNumber");   // required fields $dg -> set\_col\_required("orderNumber, customerNumber");   $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/required_fields.php?iframe=true&width=1040&height=700)

To test, click on any row to toggle inline edit, leave a required fields blank, then hit Enter.

[Comments ( 0 )](http://phpgrid.com/example/required-fields/#respond)

[**Select Multiple Records**](http://phpgrid.com/example/select-multiple-records/)

by [Richard](http://phpgrid.com/author/richard/) on 5:07 pm in [Examples](http://phpgrid.com/example/)

You can select multiple records with [set\_multiselect()](http://phpgrid.com/documentation/set_multiselect/) method. When multiselect is enabled, a checkbox is shown to the left of each row. It is currently designed to delete multiple records. More functionalities will be added to multiselect in the future.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM Orders", "orderNumber", "Orders");   // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");   // enable edit $dg -> enable\_edit("INLINE", "CRUD");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // read only columns, one or more columns delimited by comma $dg -> set\_col\_readonly("orderDate, customerNumber");   // required fields $dg -> set\_col\_required("orderNumber, customerNumber");  // multiple select $dg -> set\_multiselect(true);   $dg -> display(); |

Optional JavaScript to retrieve selected rows

|  |  |
| --- | --- |
|  | <script type="text/javascript"> function ShowSelectedRows(){     var rows = getSelRows();     // replace your own javascript here     alert(rows); } </script> |

To get value from a different cell of the select row:

|  |  |
| --- | --- |
|  | $('#TABLE\_NAME').jqGrid('getCell',row\_id,'COLUMN NAME'); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/select_multiple_rows.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/select-multiple-records/#respond)

[**Set Edit HTML Control**](http://phpgrid.com/example/set_col_edittype-2/)

by [Richard](http://phpgrid.com/author/richard/) on 5:11 pm in [Examples](http://phpgrid.com/example/)

After enabling edit, you can set the type of HTML control with [set\_col\_edittype()](http://phpgrid.com/documentation/set_col_edittype/) for any editable column used. List of Available controls are:

* text
* textarea
* select
* checkbox
* password
* **autocomplete** (version 6+) – New!

Text and textarea are set automatically based on the database access library [ADOdb metatype](http://phplens.com/lens/adodb/docs-adodb.htm#metatype) as the default edit control.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from employees", "employeeNumber", "Employees"); $dg -> set\_col\_title("employeeNumber", "Emp No."); $dg -> set\_col\_title("lastName", "Last Name"); $dg -> set\_col\_title("firstName", "First Name"); $dg -> set\_col\_title("isActive", "Active?"); $dg -> set\_col\_format("email", "email"); $dg -> enable\_edit("FORM", "CRUD"); $dg -> set\_row\_color("","","#DDEEF5"); $dg -> set\_col\_hidden('employeeNumber',false);  $dg -> set\_col\_edittype("isActive", "checkbox","1:0"); $dg -> set\_col\_edittype("reportsTo", "select", "Select employeeNumber, concat(firstName, ' ', lastName) from employees",false); $dg -> set\_col\_edittype("officeCode", "autocomplete", "Select officeCode,city from offices",false);  // alternatively, for select type, you can hard-code the key value pair value // $dg -> set\_col\_edittype("officeCode", "select", "1:San Francisco;2:Boston;3:NYC;4:Paris;5:Tokyo;6:Sydney;7:London");  $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/html_edit_control.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/set_col_edittype-2/#respond)

[**Master Detail Grid \***](http://phpgrid.com/example/master-detail-grid/)

by [Richard](http://phpgrid.com/author/richard/) on 9:40 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*

The phpGrid went extra miles to make creating the master detail datagrid a super simple task. There is essentially a simple function to [set\_masterdetail()](http://phpgrid.com/documentation/set_masterdetail/) to set the master detail relationship between two datagrids, and have them work interactively.

[one or more detail datagrids](http://phpgrid.com/example/masterdetail-master-grid-with-multiple-detail-grids/).

Note that you can have master and detail grids from the same database table as long as different [SQL alias](http://www.w3schools.com/sql/sql_alias.asp) are used.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM Orders", "orderNumber", "Orders");   // change column titles $dg->set\_col\_title("orderNumber", "Order No."); $dg->set\_col\_title("orderDate", "Order Date"); $dg->set\_col\_title("shippedDate", "Shipped Date"); $dg->set\_col\_title("customerNumber", "Customer No.");   // enable edit $dg->enable\_edit("INLINE", "CRUD");   // hide a column $dg -> set\_col\_hidden("requiredDate");  // read only columns, one or more columns delimited by comma $dg -> set\_col\_readonly("orderDate, customerNumber");   // required fields $dg -> set\_col\_required("orderNumber, customerNumber");  // multiple select $dg -> set\_multiselect(true);   // second grid as detail grid. Notice it is just another regular phpGrid with properites. $sdg = new C\_DataGrid("SELECT orderNumber,productCode,quantityOrdered,priceEach FROM OrderDetails", "orderNumber", "OrderDetails"); $sdg->set\_col\_title("orderNumber", "Order No."); $sdg->set\_col\_title("productCode", "Product Code"); $sdg->set\_col\_title("quantityOrdered", "Quantity"); $sdg->set\_col\_title("priceEach", "Unit Price"); $sdg->set\_col\_dynalink("productCode", "http://www.example.com/", "orderLineNumber", '&foo'); $sdg->set\_col\_format('orderNumber','integer', [array](http://www.php.net/array)('thousandsSeparator'=>'','defaultValue'=>'')); $sdg->set\_col\_currency('priceEach','$');  // define master detail relationship by passing the detail grid object as the first parameter, then the foriegn key name. $dg->set\_masterdetail($sdg, 'orderNumber');  $dg->display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/master_detail.php?iframe=true&width=1040&height=900)

Comments ( 0 )[**Master Detail with Column Alias**](http://phpgrid.com/example/master-detail-with-column-alias/)

by [Richard](http://phpgrid.com/author/richard/) on 9:41 pm in [Examples](http://phpgrid.com/example/)

It can be tricky working with master detail grids with column name alias, especially during edit. Using our sample database, here’s a working snippet.

|  |  |
| --- | --- |
|  | $sg = new C\_DataGrid("SELECT employeeNumber, employeeNumber as salesRepEmployeeNumber, lastName, firstName, email, jobTitle FROM  employees","employeeNumber","employees"); $sg\_d1 = new C\_DataGrid("SELECT customerNumber, customerName, city, state, salesRepEmployeeNumber, phone FROM customers","customerNumber","customers");  $sg->set\_masterdetail($sg\_d1, 'salesRepEmployeeNumber'); $sg->enable\_edit()->set\_col\_readonly("salesRepEmployeeNumber"); $sg\_d1->enable\_edit();  $sg->display();﻿ |

In master table “employeeNumber” is the FK reference to the detail table that has the column named “salesRepEmployeeNumber”﻿. Create an alias that matches the corresponding name used in detail table, in this case it’s “salesRepEmployeeNumber”; it’s important to also keep the original column name because in some database such as MySQL does not allow alias in WHERE clause.

Lastly, use set\_col\_readonly() to set the alias name “salesRepEmployeeNumber” to read only so it won’t be passed to database for editing.

The only limitation is that the SQL alias must be in master grid, or the edit will again fail.

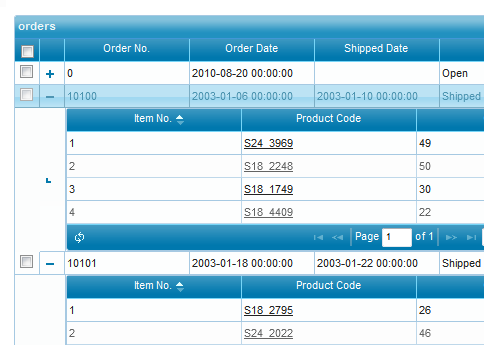
[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/master_detail_aliasfk.php?iframe=true&width=1590&height=1000)

Also available on KB: <http://phpgrid.uservoice.com/knowledgebase/articles/411029>

Comments ( 0 ) [**Subgrid \***](http://phpgrid.com/example/subgrid/)

by [Richard](http://phpgrid.com/author/richard/) on 9:43 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*



Defining a subgrid is essentially the same as setting up the master detail grid as described in our previous example. Again, only a single line of code change is required by calling [set\_subgrid()](http://phpgrid.com/documentation/set_subgrid/).

**Starting version 6, nested, drill-down subgrids are supported!**

Due to the calling sequence, the enable\_edit() should be called BEFORE set\_subgrid() method, or edit properties will be ignored.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");  // enable edit $dg->enable\_edit("INLINE", "CRUD");  // second grid as detail grid. Notice it is just another regular phpGrid with properites. $sdg = new C\_DataGrid("SELECT \* FROM orderdetails", [array](http://www.php.net/array)("orderLineNumber", "productCode"), "orderdetails"); $sdg->enable\_edit("INLINE", "CRUD");  // second grid as detail grid. Notice it is just another regular phpGrid with properites. $sdg2 = new C\_DataGrid("SELECT \* FROM products", [array](http://www.php.net/array)("productCode"), "products"); $sdg2->enable\_edit("INLINE", "CRUD");  // define master detail relationship by passing the detail grid object as the first parameter, then the foriegn key name. $sdg->set\_subgrid($sdg2, 'productCode'); $dg->set\_subgrid($sdg, 'orderNumber');  $dg->display(); |

[**See 3-Level Nested Subgrid Example!**](http://phpdatagrid.com/examples/phpGridx/examples/subgrid.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/subgrid/#respond)

[**Multiple PHP Editable Datagrid**](http://phpgrid.com/example/multiple-editable-php-datagrids/)

by [Richard](http://phpgrid.com/author/richard/) on 9:48 pm in [Examples](http://phpgrid.com/example/)

Did you know it is possible to have multiple editable grids in a single page? No need to separate your data management into multiple pages. You can now performa all your content administrative tasks in one page.

Note that you can have multiple datagrids of the same SQL table on a single page. However, you must give different [SQL alias](http://www.w3schools.com/sql/sql_alias.asp) to each table.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 | $dg = new C\_DataGrid("SELECT \* FROM Orders", "orderNumber", 'Orders'); $dg->display();  $dg2 = new C\_DataGrid("select \* from employees", "employeeNumber", "Employees"); $dg2->display();  $dg3 = new C\_DataGrid("select \* from offices", "officeCode", "Offices"); $dg3->display();  $dg4 = new C\_DataGrid("select \* from productlines", "productLine", "ProductLines"); $dg4->display();  $dg5 = new C\_DataGrid("select \* from customers", "customerNumber", "Customers"); $dg5->display(); |

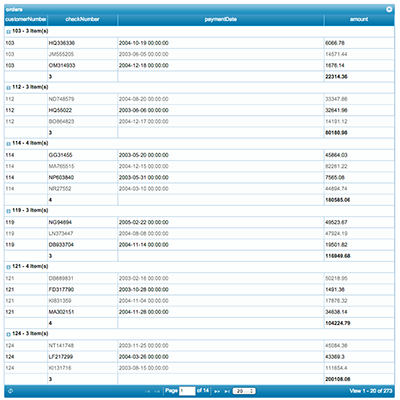
[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/multiple_datagrids.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/multiple-editable-php-datagrids/#respond)

[**Column Grouping with Summary \***](http://phpgrid.com/example/column-grouping-with-summary/)

by [Richard](http://phpgrid.com/author/richard/) on 6:11 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*

[](http://phpgrid.com/wp-content/uploads/2011/02/Simple-Grouping-with-data.png)

phpGrid can group data by columns. Simply define a column name on which grouping occur using method [set\_group\_properties()](http://phpgrid.com/documentation/set_group_properties/).

In addition to define grouping column, set summary field using method [set\_group\_summary()](http://phpgrid.com/documentation/set_group_summary/) by passing the column name and summary type, which can be sum, count, avg, min, or max.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 | $dg = new C\_DataGrid("SELECT \* FROM payments", "customerNumber", "orders");  $dg->set\_group\_properties('customerNumber'); $dg->set\_group\_summary('amount','sum');  $dg->set\_group\_summary('checkNumber','count');  $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/group_grid.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/column-grouping-with-summary/#respond)

[**Master with Multiple Detail Grids \***](http://phpgrid.com/example/masterdetail-master-grid-with-multiple-detail-grids/)

by [Richard](http://phpgrid.com/author/richard/) on 5:54 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*

phpGrid supports a master grid with multiple detail grids. There is no limit on how many detail grids a master datagrid can have. Detail grids are defined the way as the master datagrid does. Click on example link to see it in action!

Note that you can have master and detail grids from the same database table as long as different [SQL alias](http://www.w3schools.com/sql/sql_alias.asp) are used.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 | //suppliers master-detail $sg = new C\_DataGrid("SELECT \* FROM suppliers","supplierCode","suppliers"); $sg->set\_sql\_key("supplierName");   //supplier detail 1: product lines $sg\_d1 = new C\_DataGrid("SELECT \* FROM supplierproductlines","supplierCode","supplierproductlines");   $sg->set\_masterdetail($sg\_d1, 'supplierName');   //supplier detail2: products $sg\_d2 = new C\_DataGrid("SELECT productCode,productName,productDescription,quantityInStock,MSRP,productVendor FROM products","productCode","products");   //set detail 2 for suppliers $sg->set\_masterdetail($sg\_d2, 'productVendor'); $sg->display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/master_detail_multiple.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/masterdetail-master-grid-with-multiple-detail-grids/#respond)

[**Nested Master Detail Datagrid \***](http://phpgrid.com/example/nested-master-detail-datagrid/)

by [Richard](http://phpgrid.com/author/richard/) on 4:02 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*

The master detail datagrid can be nested to show relationships for multilevel hierarchal data. The example shows only three levels, though the number of nested levels can be unlimited.

|  |  |
| --- | --- |
|  | //suppliers master-detail $sg = new C\_DataGrid("SELECT \* FROM suppliers","supplierCode","suppliers"); $sg->set\_sql\_key("supplierName");  //supplier detail2: products $sg\_d1 = new C\_DataGrid("SELECT productCode,productName,productDescription,quantityInStock,MSRP,productVendor FROM products","productCode","products");  //nested grid-level 3 for products $sg\_d1\_n1 = new C\_DataGrid("SELECT \* FROM productparts","productCode","productparts");  //set detail for products $sg\_d1->set\_masterdetail($sg\_d1\_n1, 'productCode');  //set detail 2 for suppliers $sg->set\_masterdetail($sg\_d1, 'productVendor');  $sg->display(); |

Note:  
in example, click on #4 on suppliers, then S10\_4698 in the first detail grid to reveal the third nested detail grid.  
[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/master_detail_nested.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/nested-master-detail-datagrid/#respond)

[**Locale Setting**](http://phpgrid.com/example/locale-setting/)

by [Richard](http://phpgrid.com/author/richard/) on 11:19 am in [Examples](http://phpgrid.com/example/)

If you are using phpGrid in a language rather English, you may want the datagrid to display appropriate system text used by phpGrid such as “Submit”, “Next”, “Page”, “Find” etc. The language locale files are located in js\src\i18n directory.

|  |  |
| --- | --- |
| 1 2 3 | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->set\_locale('fr');         $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/locale_setting.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/locale-setting/#respond)

[**Reference to Multiple Databases \***](http://phpgrid.com/example/multiple-databases-support/)

by [Richard](http://phpgrid.com/author/richard/) on 11:23 am in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is only available in paid versions.*

phpGrid nows supports data source from more than one database on a single page. Simply use the 4th parameter in constructor (C\_DataGrid) to overwrite the initial database settings in conf.php.

In the following example, grid A binds a table from one database, while grid B binds another table from a completely different database, all on the same page.

|  |  |
| --- | --- |
|  | $gridA = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $gridA->enable\_edit("INLINE","CRUD"); $gridA -> display();  // overwrite database settings with another database. $gridB = new C\_DataGrid("SELECT \* FROM names", "id", "names",                         [array](http://www.php.net/array)("hostname"=>"localhost",                             "username"=>"USER2",                             "password"=>"PASSWORD",                             "dbname"=>"utf8db",                              "dbtype"=>"mysql",                              "dbcharset"=>"utf8"));  $gridB->enable\_edit("INLINE","CRUD");  $gridB -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/multiple_database.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/multiple-databases-support/#respond)

[**Display Non-English Characters**](http://phpgrid.com/example/display-foreign-language-characters/)

by [Richard](http://phpgrid.com/author/richard/) on 6:30 pm in [Examples](http://phpgrid.com/example/)

If database is created in charset other than default charset, e.g. “latin1″, some characters may be displayed as “?” in the grid. To properly display non-English characters, such as Spanish, Fresh, or Chinese, in MySQL, PostgreSQL, and Oracle you can define the character set value in ‘DB\_CHARSET’ variable in conf.php.

In conf.php, set the DB\_CHARSET to the corresponding charsets in your database.

version 6+ (with “PHPGRID\_” prefix):

|  |  |
| --- | --- |
|  | [define](http://www.php.net/define)('PHPGRID\_DB\_HOSTNAME','hostname'); // database host name [define](http://www.php.net/define)('PHPGRID\_DB\_USERNAME', 'username'); // database user name [define](http://www.php.net/define)('PHPGRID\_DB\_PASSWORD', 'password'); // database password [define](http://www.php.net/define)('PHPGRID\_DB\_NAME', 'sampledb'); // database name [define](http://www.php.net/define)('PHPGRID\_DB\_TYPE', 'mysql'); // database type [define](http://www.php.net/define)('PHPGRID\_DB\_CHARSET','utf8'); // OPTIONAL. Leave blank to use the default charset |

version 5.5.x and below:

|  |  |
| --- | --- |
|  | [define](http://www.php.net/define)('DB\_HOSTNAME','hostname'); // database host name [define](http://www.php.net/define)('DB\_USERNAME', 'username'); // database user name [define](http://www.php.net/define)('DB\_PASSWORD', 'password'); // database password [define](http://www.php.net/define)('DB\_NAME', 'sampledb'); // database name [define](http://www.php.net/define)('DB\_TYPE', 'mysql'); // database type [define](http://www.php.net/define)('DB\_CHARSET','utf8'); // OPTIONAL. Leave blank to use the default charset |

Add the Meta element to the HTML head node and set character encoding to UTF-8. **Note it’s important that the actual file must also be** [**saved as UTF-8 encoding**](http://www.w3.org/International/questions/qa-setting-encoding-in-applications).

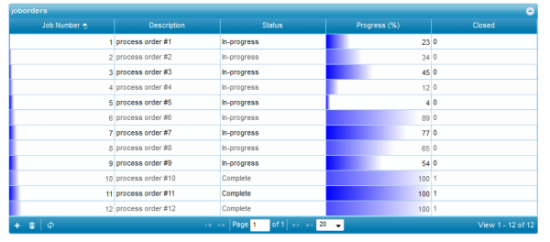
|  |  |
| --- | --- |
| 1 | <[meta](http://december.com/html/4/element/meta.html) http-equiv="Content-Type" content="text/html; charset=UTF-8" /> |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/foreign_language_characters.php?iframe=true&width=1040&height=550)

[Comments ( 0 )](http://phpgrid.com/example/display-foreign-language-characters/#respond)

[**In-cell Data Bar \***](http://phpgrid.com/example/in-cell-data-bar/)

by [Richard](http://phpgrid.com/author/richard/) on 12:04 pm in [Examples](http://phpgrid.com/example/)



Bar chart is a great way to visualize numeric data. phpGrid now supports bar chart natively without 3rd party plugin using [set\_databar()](http://phpgrid.com/documentation/set_databar/) method. You can have multiple data bar in a datagrid.

For complex data visualization, we recommend [PHP Chart](http://phpchart.net/). Please visit [PHP Chart](http://phpchart.net/) for live demo.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM joborders", "jobNumber", "joborders"); $dg -> set\_col\_title("jobNumber", "Job Number"); $dg -> set\_col\_title("jobDescription", "Description"); $dg -> set\_col\_title("status", "Status");                $dg -> set\_col\_title("percentComplete", "Progress (%)");  $dg -> set\_col\_title("isClosed", "Closed");    $dg -> set\_databar("percentComplete","blue"); $dg -> set\_databar("jobNumber","blue"); $dg -> enable\_edit("INLINE", "CRUD");    $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/data_bar.php?iframe=true&width=1040&height=500)

[Comments ( 0 )](http://phpgrid.com/example/in-cell-data-bar/#respond)

[**Conditional Value \***](http://phpgrid.com/example/conditional-value/)

by [Richard](http://phpgrid.com/author/richard/) on 12:08 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is not available in Lite and Basic versions.*

Conditional Value is similar to conditional format (see [set\_conditional\_format](http://phpgrid.com/documentation/set_conditional_format/) but with simplier set of features. Use set\_conditional\_value() to dynamically display a value when specific condition is met. The conditional value can be text, HTML, or even CSS style.

You can use conditional value with data bar in the same grid.

|  |  |
| --- | --- |
|  | .tstyle { display:block;background-image:none;margin-right:-2px;margin-left:-2px;height:14px;padding:5px;background-color:green;color:navy;font-weight:bold } .fstyle {  display:block;background-image:none;margin-right:-2px;margin-left:-2px;height:14px;padding:5px;background-color:yellow;color:navy } |

**PHP Code**

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM joborders", "jobNumber", "joborders"); $dg -> set\_col\_title("jobNumber", "Job Number"); $dg -> set\_col\_title("jobDescription", "Description"); $dg -> set\_col\_title("status", "Status");                $dg -> set\_col\_title("percentComplete", "Progress (%)");  $dg -> set\_col\_title("isClosed", "Closed");    $dg->set\_conditional\_value("isClosed", "==1", [array](http://www.php.net/array)(     "TCellValue"=>"<img src='SampleImages/checked.gif' />",     "FCellValue"=>"<img src='SampleImages/unchecked.gif' />"));   $dg->set\_conditional\_value("status", "=='Complete'", [array](http://www.php.net/array)(     "TCellStyle"=>"tstyle",     "FCellStyle"=>"fstyle"));   $dg->enable\_edit('INLINE', 'CRUD');   $dg->set\_multiselect(true); $dg -> set\_databar("percentComplete","red"); $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/conditional_value.php?iframe=true&width=1040&height=500)

**Note:**  
For even more complex conditions, please refer to [row level permission example](http://phpgrid.com/example/row-level-permission-edit-condtion/) by using [set\_grid\_property()](http://phpgrid.com/documentation/set_grid_property/) and [add\_event()](http://phpgrid.com/documentation/add_event/) method.

[Comments ( 0 )](http://phpgrid.com/example/conditional-value/#respond)

[**Create Excel-Like, Responsive Grid**](http://phpgrid.com/example/grid-auto-width/)

by [Richard](http://phpgrid.com/author/richard/) on 12:10 pm in [Examples](http://phpgrid.com/example/)

Use [enable\_autowidth()](http://phpgrid.com/documentation/enable_autowidth/) and [enable\_autoheight()](http://phpgrid.com/documentation/enable_autoheight/) to set datagrid responsive with the page dimension to fill the entire screen, similar to Excel spreadsheet.

|  |  |
| --- | --- |
| 1 2 3 4 | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->enable\_edit('INLINE', 'CRUD'); $dg->enable\_autowidth(true)->enable\_autoheight(true); $dg->display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/auto_fit_to_screen.php) (Try to resize the window)

Hint:  
Call [enable\_kb\_nav()](http://phpgrid.com/documentation/enable_kb_nav-beta/) method to move between rows using only keyboard.

[Comments ( 0 )](http://phpgrid.com/example/grid-auto-width/#respond)

[**Horizontal Scroll**](http://phpgrid.com/example/horizontal-scroll/)

by [Richard](http://phpgrid.com/author/richard/) on 12:13 pm in [Examples](http://phpgrid.com/example/)

Set parameter **$shrinkToFit** in [set\_dimension()](http://phpgrid.com/documentation/set_dimension/) to false to enable horizontal scroll when the container width is less than the grid width. The scroll bar will not show when the grid width is equal or less than the container width.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");  // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber",  "Customer No."); // hide a column $dg -> set\_col\_hidden("requiredDate"); // change default caption $dg -> set\_caption("Orders List"); // set export type $dg -> enable\_export('EXCEL');  // enable integrated search $dg -> enable\_search(true); // set height and weight of datagrid $dg -> set\_dimension(800, 600, false);  // increase pagination size to 40 from default 20 $dg -> set\_pagesize(40);  $dg -> display(); |

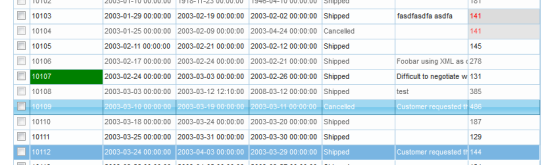
[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/horizontal_scroll.php?iframe=true&width=950&height=700)

[Comments ( 0 )](http://phpgrid.com/example/horizontal-scroll/#respond)

[**Conditional Format \***](http://phpgrid.com/example/conditional-format-2/)

by [Richard](http://phpgrid.com/author/richard/) on 12:51 pm in [Examples](http://phpgrid.com/example/)

*\* Please note this feature is not available in Lite and Basic versions.*



Conditional formatting example using [set\_conditional\_format()](http://phpgrid.com/documentation/set_conditional_format/) method.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");  //Format a cell based on the specified condition $dg->set\_conditional\_format("orderNumber","CELL",[array](http://www.php.net/array)(     "condition"=>"eq","value"=>"10107","css"=> [array](http://www.php.net/array)("color"=>"#ffffff","background-color"=>"green")));  $dg->set\_conditional\_format("customerNumber","CELL",[array](http://www.php.net/array)(     "condition"=>"eq","value"=>"141","css"=> [array](http://www.php.net/array)("color"=>"red","background-color"=>"#DCDCDC")));  // Format a row based on the specified condition $dg->set\_conditional\_format("comments","ROW",[array](http://www.php.net/array)(     "condition"=>"cn","value"=>"request","css"=> [array](http://www.php.net/array)("color"=>"white","background-color"=>"#4297D7")));                            $dg->set\_multiselect(true); $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/conditional_formatting.php?iframe=true&width=1040&height=700)

**Note:**  
For even more complex conditions, please refer to [row level permission example](http://phpgrid.com/example/row-level-permission-edit-condtion/) by using [set\_grid\_property()](http://phpgrid.com/documentation/set_grid_property/) and [add\_event()](http://phpgrid.com/documentation/add_event/) method.

[Comments ( 0 )](http://phpgrid.com/example/conditional-format-2/#respond)

[**Method Chaining**](http://phpgrid.com/example/method-chaining/)

by [Richard](http://phpgrid.com/author/richard/) on 10:22 am in [Examples](http://phpgrid.com/example/)

Method chaining is now supported starting version 5.0. It’s not required, but it provides more readable code and reduce amount of source code in a fluent interface.

For example:

With method chaining (v5.0 and later):

|  |  |
| --- | --- |
|  | $dg->set\_col\_title(“lastName”, “Last Name”)->set\_col\_required(“lastName”, true)->set\_co\_default(“lastName”, “Robert”); $dg->enable\_edit(“FORM”)->enable\_export(true)->enable\_resize(true); |

Without method chaining:

|  |  |
| --- | --- |
|  | $dg -> set\_col\_title(“lastName”, “Last Name”); $dg -> set\_col\_required(“lastName”, true); $dg -> set\_co\_default(“lastName”, “Robert”); $dg -> enable\_edit(“FORM”); $dg -> enable\_export(true); $dg -> enable\_resize(true); |

[Comments ( 0 )](http://phpgrid.com/example/method-chaining/#respond)

[**Custom Event Handler**](http://phpgrid.com/example/custom-event-handler/)

by [Richard](http://phpgrid.com/author/richard/) on 10:09 pm in [Examples](http://phpgrid.com/example/)

phpGrid supports custom event handler using [add\_event()](http://phpgrid.com/documentation/add_event/) method. The event handlers are essentially JavaScript functions so they must be enclosed with PHP [heredoc](http://us2.php.net/manual/en/language.types.string.php#language.types.string.syntax.heredoc) syntax. This is a very powerful function once you are familiar with events.

Below are the events used in this example:

* jqGridSelectRow
* jqGridrowattr
* jqGridAddEditAfterSubmit

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");  $onSelectRow = <<<ONSELECTROW function(status, rowid) {     alert('event 1');     console.log(rowid);     console.log(status);          /\* example to redirect a new URL when select a row \*/     // orderNumber = $('#orders').jqGrid('getCell',rowid,'orderNumber');     // customerNumber = $('#orders').jqGrid('getCell',rowid,'customerNumber');     // window.location = encodeURI("http://example.com/" + "?" + "orderNumber=" + orderNumber + "&customerNumber="+customerNumber); } ONSELECTROW;  $onSelectRow2 = <<<ONSELECTROW2 function(status, rowid) {     alert('event 2');     console.log("here"); } ONSELECTROW2;  $rowattr = <<<ROWATTR function (rowData, inputRowData)  {     return rowData.status === "OnHold" ? {style: "background-color:blue"} : {}; } ROWATTR;  // post data another page after submit $afterSubmit = <<<AFTERSUBMIT function (event, status, postData) {     console.log(postData);     alert(postData.customerNumber);     $.ajax({ url: '/my/site',         data: {custNum: postData.customerNumber}, // replace customerNumber with your own field name         type: 'post',         success: function(output) {                     alert(output);                 }         }); } AFTERSUBMIT;  $dg->add\_event("jqGridSelectRow", $onSelectRow); $dg->add\_event("jqGridSelectRow", $onSelectRow2); $dg->add\_event("jqGridrowattr", $rowattr); $dg->add\_event("jqGridAddEditAfterSubmit", $afterSubmit); $dg->enable\_edit('FORM'); $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/set_custom_event_handler.php?iframe=true&width=1040&height=700) (click on any row to trigger event)

[Comments ( 0 )](http://phpgrid.com/example/custom-event-handler/#respond)

[**Column Property**](http://phpgrid.com/example/column-property/)

by [Richard](http://phpgrid.com/author/richard/) on 12:33 pm in [Examples](http://phpgrid.com/example/)

You can now directly manipulate individual column properties without using helper set\_col\_\* functions such as [set\_col\_format()](http://phpgrid.com/documentation/set_col_format/) and [set\_col\_readonly()](http://phpgrid.com/documentation/set_col_readonly/) .  This is suitable for users who are already familiar with jqGrid colMdel API (<http://www.trirand.com/jqgridwiki/doku.php?id=wiki:colmodel_options>).

The new column property method does not replace existing helper functions. It should be used beside by side with the existing set\_col\_\* helper functions.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg -> set\_row\_color('yellow', 'blue', 'lightgray'); $dg -> set\_col\_property("orderNumber", [array](http://www.php.net/array)("name"=>;"Order Number", "width"=>;40)); // display only time $dg -> set\_col\_property("orderDate",                          [array](http://www.php.net/array)("formatter"=>;"date",                               "formatoptions"=>;[array](http://www.php.net/array)("srcformat"=>;"ISO8601Short","newformat"=>;"g:i A")); $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/column_property.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/column-property/#respond)

[**Hyperlink to Call JavaScript Function**](http://phpgrid.com/example/call-javascript-function-on-hyperlink-click/)

by [Richard](http://phpgrid.com/author/richard/) on 1:59 pm in [Examples](http://phpgrid.com/example/)

You can show hyperlink using [set\_col\_link](http://phpgrid.com/documentation/set_col_link/) for static link or [set\_col\_dynalink](http://phpgrid.com/documentation/set_col_dynalink/) for dynamic link. What about if you want to do something even fancier, like open a new window, or trigger a JavaScript event when hyperlink is clicked?

To call JavaScript function on hyperlink click, use “showlink” in [set\_col\_format](http://phpgrid.com/documentation/set_col_format/) method.

In the following example, the first set\_col\_format displays text in alert window using JavaScript function “myFunction” to retrieved text from a different cell in the same row. The second does a phpgrid.com site search using jQuery Ajax function “phpGridSiteAjaxSearch”, and displays the result in a alert message window.

PHP:

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from products", "productCode", "products"); $dg -> set\_col\_title("productCode", "Product Code"); $dg -> set\_col\_title("productName", "Product Name"); $dg -> set\_col\_title("productLine", "Product Line");  $dg->set\_col\_format("productLine", "showlink", [array](http://www.php.net/array)("baseLinkUrl"=>"javascript:", "target"=>"\_self",     "showAction"=>"myFunction(jQuery('#products'),'",      "addParam"=>"');"));                                                                                                                             $dg->set\_col\_format("productName", "showlink", [array](http://www.php.net/array)("baseLinkUrl"=>"javascript:", "target"=>"\_self",     "showAction"=>"phpGridSiteAjaxSearch(jQuery('#products'),'",      "addParam"=>"');"));                                                                                                                            $dg -> display(); |

JavaScript:

|  |  |
| --- | --- |
|  | myFunction = function (grid,param) {     var ar = param.split('=');     if (grid.length > 0 && ar.length === 2 && ar[0] === '?id') {         var rowid = ar[1];         var prodDesc = grid.getCell(rowid, 'productDescription');          alert(prodDesc);     } };  phpGridSiteAjaxSearch = function (grid,param) {     var ar = param.split('=');     if (grid.length > 0 && ar.length === 2 && ar[0] === '?id') {         var rowid = ar[1];         var productName = grid.getCell(rowid, 'productName');          jQuery.ajax({             url: 'http://phpgrid.com',             data: { s: productName },             type: 'GET',             dataType: 'text',             success:function(data,status) {                 // your code goes here for a successul Ajax call .e.g open a new window                 alert(data);             },             error:function(data,status,err){                 alert(data + status);             }         });     } }; |

[See Live Example](http://phpdatagrid.com/examples/phpGridx/examples/hyperlink_click_event.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/call-javascript-function-on-hyperlink-click/#respond)

[**PHP Datagrid AJAX File Upload**](http://phpgrid.com/example/php-datagrid-ajax-file-upload/)

by [Richard](http://phpgrid.com/author/richard/) on 10:43 pm in [Examples](http://phpgrid.com/example/)

\* *The file upload is supported in Enterprise and Universal license.*

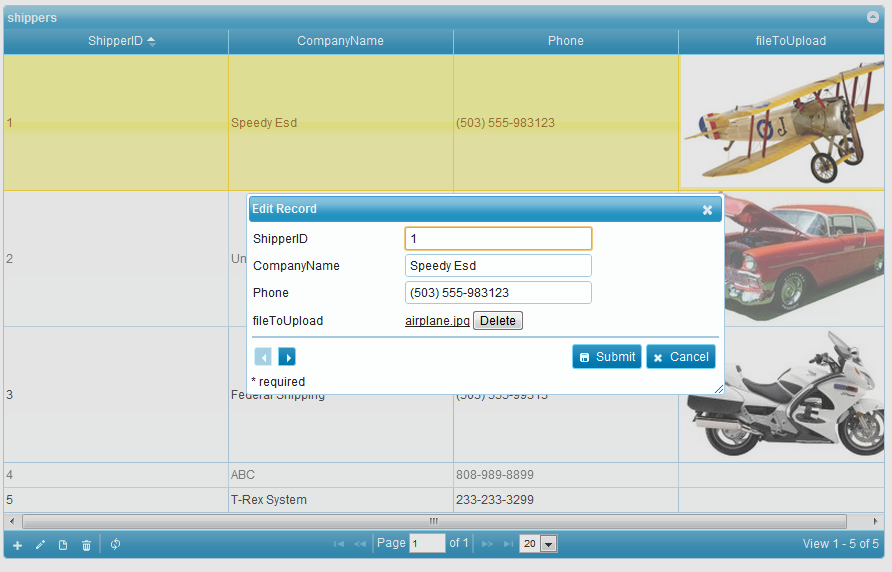
File upload is one of the most requested features. And we are finally able to delivery this new feature. It requires only a SINGLE LINE of code, with only a single function call to [set\_col\_fileupload()](http://phpgrid.com/documentation/set_col_fileupload/). Since PHP is installed as [Apache module](http://php.net/manual/en/security.apache.php) in our host server, the 3rd parameter is optional that the physical path is automatically mapped based on 2nd parameter, the base URL.

All the file operations (Update, Delete) are Ajax-based, without refreshing the page.

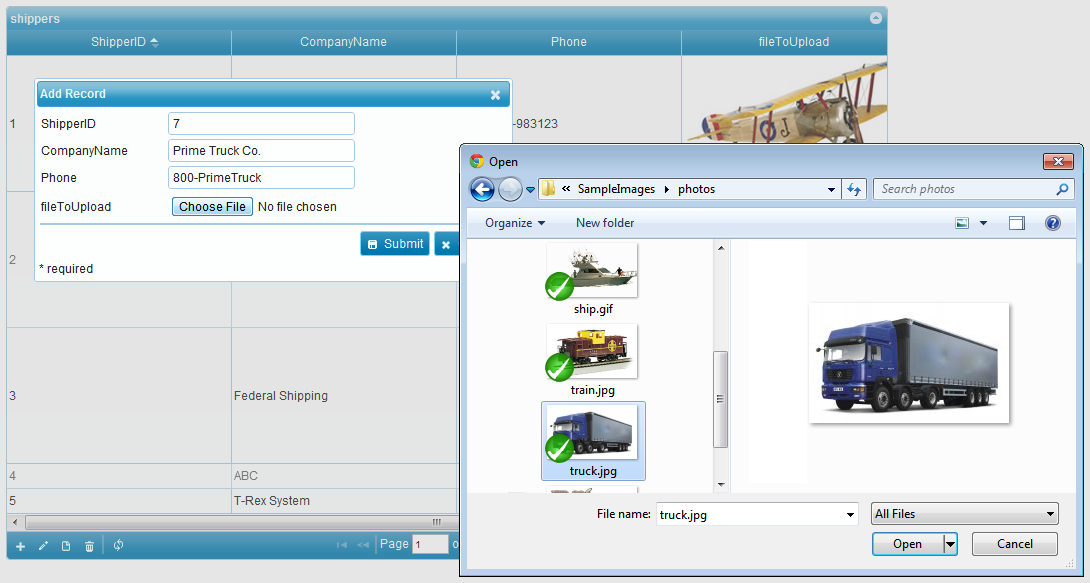
|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM shippers", "ShipperID", "shippers"); $dg -> set\_col\_fileupload("fileToUpload", '/phpGridx/photos/'); $dg -> enable\_edit('FORM'); $dg->set\_col\_img("fileToUpload", "/phpGridx/photos/"); $dg -> display(); |

You should consider resizing images displayed so large images won’t blow your datagrid out of proportion. Read more about [Image resize using CSS](http://phpgrid.uservoice.com/knowledgebase/articles/253453).

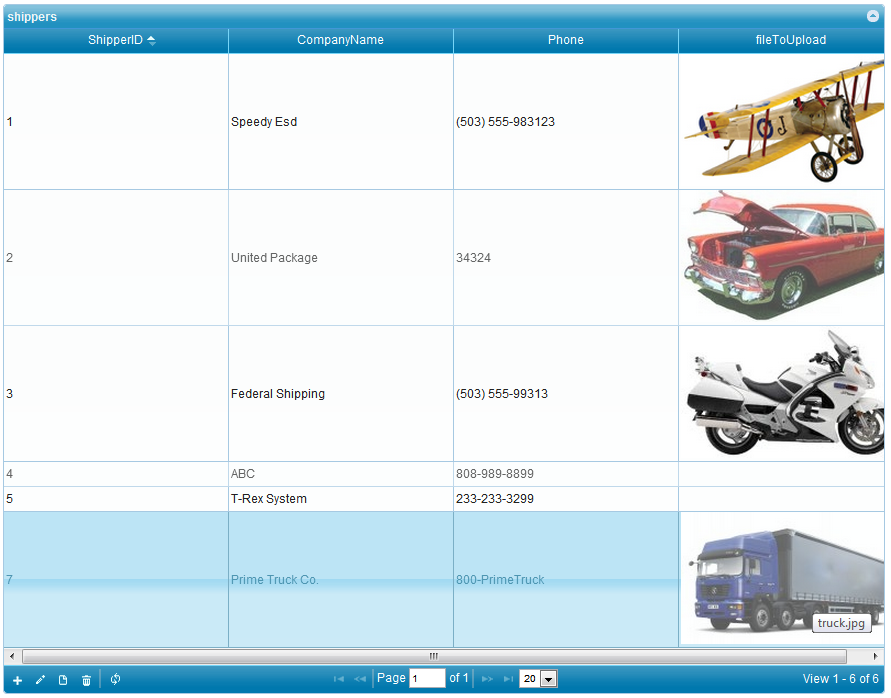
The below screenshot (click to enlarge) illustrates the file edit operations in edit form. It has the “delete” button to remove the existing file before uploading a new one. The file name is saved to table “shippers”.

[](http://phpgrid.com/wp-content/uploads/2012/07/fileupload_edit.png)

The below screenshot (click to enlarge) shows uploading new file as part of creating a new record by click on “choose file”, then pick file “truck.jpg” from the local system. Then the file “truck.jpg” is saved to file system under “/phpGridx/foo/” folder, and the file name is stored in table “shippers” alone with rest of data.

[](http://phpgrid.com/wp-content/uploads/2012/07/fileupload_addnew.png)

The follow screen (click to enlarge) shows that new record has been successfully created and the file “truck.jpg” has been successfully uploaded.

[](http://phpgrid.com/wp-content/uploads/2012/07/fileupload_addnew_done.png)

[Comments ( 0 )](http://phpgrid.com/example/php-datagrid-ajax-file-upload/#respond)

[**Customize Edit Form Layout**](http://phpgrid.com/example/customize-edit-form-layout/)

by [Richard](http://phpgrid.com/author/richard/) on 12:21 pm in [Examples](http://phpgrid.com/example/)

By default, the edit form is displayed a single column table. This is fine for table with small numbers of fields. When you got a large number of fields, the chances are that you want to modify the layout to display multiple columns. Use **formoptions** property “**rowpos**” and “**colpos**” for this purpose.

The following example demonstrates a 2-column edit form.

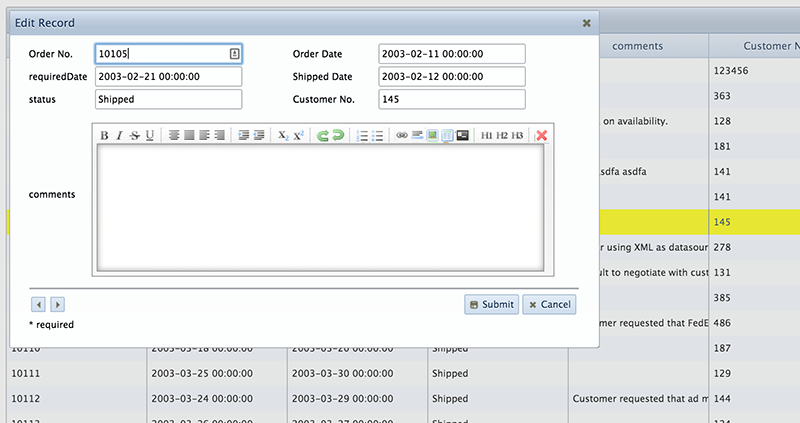
|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");   // change column titles $dg -> set\_col\_title("orderNumber", "Order No."); $dg -> set\_col\_title("orderDate", "Order Date"); $dg -> set\_col\_title("shippedDate", "Shipped Date"); $dg -> set\_col\_title("customerNumber", "Customer No.");  // hide a column $dg -> set\_col\_hidden("requiredDate");  // enable edit $dg -> enable\_edit("FORM", "CRUD");   $dg -> set\_col\_property("orderNumber", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>1,"colpos"=>1))); $dg -> set\_col\_property("orderDate", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>1,"colpos"=>2))); $dg -> set\_col\_property("requiredDate", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>2,"colpos"=>1))); $dg -> set\_col\_property("shippedDate", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>2,"colpos"=>2))); $dg -> set\_col\_property("status", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>3,"colpos"=>1))); $dg -> set\_col\_property("customerNumber", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>3,"colpos"=>2))); $dg -> set\_col\_property("comments", [array](http://www.php.net/array)("formoptions"=>[array](http://www.php.net/array)("rowpos"=>4,"colpos"=>1))); $dg -> set\_col\_property("comments", [array](http://www.php.net/array)("editoptions"=>[array](http://www.php.net/array)("style"=>"width:95%;")));   $dg->set\_form\_dimension(700, 400);  $dg->enable\_debug(false); $dg -> display(); |

**Bonus! Set comments field to always have the full width of the form. Make sure change “orders” to your own table name used for grid.**

|  |  |
| --- | --- |
|  | <script> // Make sure nothing else is sharing the same row! $(document).ready(function(){     var grid=$("#orders");      // your jqGrid (the <table> element)     var orgEditGridRow = grid.jqGrid.editGridRow; // save original function     $.jgrid.extend ({editGridRow : function(rowid, p){         $.extend(p,             {                 beforeShowForm : function(form) {                     form = $(form);                     $("tr", form).each(function() {                         var inputs = $(">td.DataTD:has(textarea)",this);                         if (inputs.length == 1) {                             var tds = $(">td", this);                             tds.eq(1).attr("colSpan", tds.length - 1);                             tds.slice(2).hide();                         }                     });                 }             });         orgEditGridRow.call (this,rowid, p);     }}); }); </script> |

**Form Layout Screenshot**

**Hint**: the edit form is resizable by click and drag bottom-right corner using mouse.

[](http://phpgrid.com/wp-content/uploads/2012/08/custom-form-edit-layout.png)

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/edit_form_layout.php?iframe=true&width=1040&height=700) (double click a row to see 2-column edit form)

[Comments ( 0 )](http://phpgrid.com/example/customize-edit-form-layout/#respond)

[**phpGrid, phpChart Integration**](http://phpgrid.com/example/phpgrid-phpchart-integration-with-live-example/)

by [Richard](http://phpgrid.com/author/richard/) on 5:19 pm in [Examples](http://phpgrid.com/example/)

[Run Demo](http://phpdatagrid.com/examples/phpgrid_phpchart_integration.php)



It’s easy to combine both phpGrid and phpChart into a seamless integrated, interactive, and completely web-based ad-hoc content and report management tool.

**Setup**

Before start, You must acquire both [phpGrid](http://phpgrid.com/downloads/) and [phpChart](http://phpchart.net/downloads/). You can download both for free.

It’s important to install phpGrid and phpChart in separate folder. Below is the folder structure used in our example. phpGrid and phpChart each resides in separate folder inside phpGrid\_phpChart folder under web root folder htdocs. **This is RECOMMENDED folder hierarchy.**

htdocs

+-- phpGrid\_phpChart

| |-- phpGrid

| | +-- conf.php

| |-- phpChart

| | +-- conf.php

| +-- phpgrid\_phpchart\_integration.php

Once folder structure is ready. Set up [phpChart SCRIPTPATH value](http://phpchart.net/examples/introduction/) in conf.php file.

phpChart conf.php

|  |  |
| --- | --- |
| 1 | [define](http://www.php.net/define)('SCRIPTPATH','/phpGrid\_phpChart/phpChart/'); |

**Programming**

Now, we can move on to the actual integration. First all, include both conf.php files on top of the script.

|  |  |
| --- | --- |
| 1 2 | require\_once("phpGrid/conf.php");  require\_once("phpChart/conf.php"); |

Then, create the chart with phpChart. Here in this example we have two charts. The first chart has a single series of data. The second one has two series and renders bar chart instead of the default line chart. All the phpChart methods and properties [documents are available on phpChart.net](http://phpchart.net/phpChart/docs/output/!!MEMBEROVERVIEW_C_PhpChartX.html). Pass “null” as the series data if you don’t want to have any initial data in the chart.

|  |  |
| --- | --- |
|  | // first phpChart with one series $pc = new C\_PhpChartX([array](http://www.php.net/array)([array](http://www.php.net/array)(null)), 'Graph'); $pc->set\_axes([array](http://www.php.net/array)(         'xaxis'=> [array](http://www.php.net/array)('label'=>'Buy Price'),         'yaxis'=> [array](http://www.php.net/array)('label'=>'$')     )); $pc->draw(800,500);   // second phpChart with two series $pc2 = new C\_PhpChartX([array](http://www.php.net/array)([array](http://www.php.net/array)(null), [array](http://www.php.net/array)(null)), 'Graph2'); $pc2->set\_axes([array](http://www.php.net/array)(         'xaxis'=> [array](http://www.php.net/array)(             'label'=>'Buy Price/MSRP (by Product Code)'),         'yaxis'=> [array](http://www.php.net/array)('label'=>'$')     )); $pc2->set\_animate(true); $pc2->set\_legend([array](http://www.php.net/array)('show'=>true,'location'=>'nw')); $pc2->draw(800,500); |

Finally the last part is to code the datagrid. In our example, we use “products” database table. **The most important part of the datagrid is the** [**custom event handler**](http://phpgrid.com/example/custom-event-handler/) **“jqGridLoadComplete” in which the phpGrid, phpChart integration happens.** Here, values in both “buyPrice” and “MSRP” columns are retrieved and stored in d1 and d2 as array. You can replace those columns with columns with your own choice in your table. GraphData1 and GraphData2 are used to store the series of data that later will be used to plot our charts created earlier.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM products", "productCode", "products"); $dg->enable\_edit('FORM'); $onGridLoadComplete = <<<ONGRIDLOADCOMPLETE function(status, rowid) {     var GraphData1 = [];     var GraphData2 = [];      d1 = $('#products').jqGrid('getCol', 'buyPrice', false);     d2 = $('#products').jqGrid('getCol', 'MSRP', false);     npoints = d1.length;     for(var i=0; i < npoints; i++){         GraphData1[i] = [i+1, parseInt(d1[i])];     }     \_Graph.series[0].data = GraphData1;     \_Graph.replot({resetAxes:true});      for(var i=0; i < npoints; i++){         GraphData1[i] = [i+1, parseInt(d1[i])];         GraphData2[i] = [i+1, parseInt(d2[i])];     }     \_Graph2.series[0].data = GraphData1;     \_Graph2.series[1].data = GraphData2;     \_Graph2.replot({resetAxes:true}); } ONGRIDLOADCOMPLETE; $dg->add\_event("jqGridLoadComplete", $onGridLoadComplete); $dg->set\_col\_property('MSRP', [array](http://www.php.net/array)('classes'=>'msrp\_bg'));   // change column background color $dg->set\_col\_property('buyPrice', [array](http://www.php.net/array)('classes'=>'buyprice\_bg'));  // change column background color $dg->enable\_search(true); $dg->display(); |

To easily recognize the data source, we set the datagrid column background color to match the series color in phpChart using CSS class

|  |  |
| --- | --- |
| 1 2 | .msrp\_bg { background-color: #4BB2C5 !important; } .buyprice\_bg {  background-color: #F8C05B !important; } |

**The COMPLETE example**

|  |  |
| --- | --- |
|  | <?php require\_once("phpGrid/conf.php"); require\_once("phpChart/conf.php"); ?> < !DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"> < html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en"> < head> < meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> < title>phpGrid phpChart Integration Example</title> < /head> < body> < STYLE> .msrp\_bg {     background-color: #4BB2C5 !important; } .buyprice\_bg {     background-color: #F8C05B !important; } < /STYLE>  < p>Try do a quick search or do a sort. Observe the graph updates in real-time.</p>  < table><tr><td> <?php // first phpChart with one series $pc = new C\_PhpChartX([array](http://www.php.net/array)([array](http://www.php.net/array)(null)), 'Graph'); $pc->set\_axes([array](http://www.php.net/array)(         'xaxis'=> [array](http://www.php.net/array)('label'=>'Buy Price'),         'yaxis'=> [array](http://www.php.net/array)('label'=>'$')     )); $pc->draw(600,300); ?> < /td> < td> <?php // second phpChart with two series $pc2 = new C\_PhpChartX([array](http://www.php.net/array)([array](http://www.php.net/array)(null), [array](http://www.php.net/array)(null)), 'Graph2'); $pc2->set\_axes([array](http://www.php.net/array)(         'xaxis'=> [array](http://www.php.net/array)(             'label'=>'Buy Price/MSRP (by Product Code)'),         'yaxis'=> [array](http://www.php.net/array)('label'=>'$')     )); $pc2->set\_animate(true); $pc2->set\_legend([array](http://www.php.net/array)('show'=>true,'location'=>'nw')); $pc2->draw(600,300); ?> < /td></tr></table>  <?php $dg = new C\_DataGrid("SELECT \* FROM products", "productCode", "products"); $dg->enable\_edit('FORM');  $onGridLoadComplete = <<<ONGRIDLOADCOMPLETE function(status, rowid) {     var GraphData1 = [];     var GraphData2 = [];      d1 = $('#products').jqGrid('getCol', 'buyPrice', false);     d2 = $('#products').jqGrid('getCol', 'MSRP', false);     npoints = d1.length;     for(var i=0; i < npoints; i++){         GraphData1[i] = [i+1, parseInt(d1[i])];     }     \_Graph.series[0].data = GraphData1;     \_Graph.replot({resetAxes:true});      for(var i=0; i < npoints; i++){             GraphData1[i] = [i+1, parseInt(d1[i])];             GraphData2[i] = [i+1, parseInt(d2[i])];     }     \_Graph2.series[0].data = GraphData1;     \_Graph2.series[1].data = GraphData2;     \_Graph2.replot({resetAxes:true}); } ONGRIDLOADCOMPLETE;  $dg->add\_event("jqGridLoadComplete", $onGridLoadComplete); $dg->set\_col\_property('MSRP', [array](http://www.php.net/array)('classes'=>'msrp\_bg')); $dg->set\_col\_property('buyPrice', [array](http://www.php.net/array)('classes'=>'buyprice\_bg')); $dg->enable\_search(true); $dg->display(); ?> < /body> < /html> |

That’s it! Do a quick search or sort. Observe the graph updating in real-time.

[Give it a try!](http://phpdatagrid.com/examples/phpgrid_phpchart_integration.php) | [Download Integration Example File [](http://phpgrid.com/grid/download/phpgrid_phpchart_integration.zip)](http://phpgrid.com/grid/download/phpgrid_phpchart_integration.zip)

**Technical Notes**

* The Javascript used in the custom event handler in phpGrid is the bread and butter of integration.
* Set array to null for initial graph.
* On initial plot creation you can pass series of data in a 1-D array. jqPlot internally converts it to a 2-D array. However, this doesn’t happen on .replot().
* If the page throws an error

|  |  |
| --- | --- |
| 1 | Notice: [Constant](http://www.php.net/constant) DEBUG already [defined](http://www.php.net/defined) in /phpGrid/conf.php on line 2 |

* This is because both phpChart and phpGrid have DEBUG constant, you need to comment out of them in conf.php

|  |  |
| --- | --- |
| 1 | // define('DEBUG', true); |

[Comments ( 0 )](http://phpgrid.com/example/phpgrid-phpchart-integration-with-live-example/#respond)

[**Virtual/Calculated Column**](http://phpgrid.com/example/virtual-column-aka-calculated-column/)

by [Richard](http://phpgrid.com/author/richard/) on 3:18 pm in [Examples](http://phpgrid.com/example/)

Starting version 5.5, you can now add virtual column, AKA calculated column, to your existing datagrid. Virtual, by definition, is that it doesn’t exist in the database table. It’s a calculated field created from other columns. The virtual columns are added to the END of the existing datagrid.

phpGrid only adds a “virtual” column and does NOT change the database table structure. Front-end user should not be able to add/change a column in your database table structure. It should be done only by a very smaller number of people such as DBA, and ultimately through a back-end database administration program.

It’s important that the virtual column name is NOT one of the existing database column names used in PHP datagrid.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->enable\_edit('INLINE');  // calculated value to be displayed in the virtual column // n1 stores column 1 value, n2 stores column 7 value..and so on. $col\_formatter = <<<COLFORMATTER function(cellvalue, options, rowObject){     var n1 = parseInt(rowObject[0],10),               n2 = parseInt(rowObject[6],10);           return n1+n2; } COLFORMATTER;  $dg -> add\_column(         'total',          [array](http://www.php.net/array)('name'=>'total',              'index'=>'total',              'width'=>'360',              'align'=>'right',              'sortable'=>false,             'formatter'=>$col\_formatter),         'Total (Virtual)'); $dg->display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/virtual_column.php?iframe=true&width=1040&height=700) (The last column “Total” is a virtual column)

[Comments ( 0 )](http://phpgrid.com/example/virtual-column-aka-calculated-column/#respond)

[**Custom Data Validation**](http://phpgrid.com/example/custom-data-validation/)

by [Richard](http://phpgrid.com/author/richard/) on 3:40 pm in [Examples](http://phpgrid.com/example/)

phpGrid automatically does data validation based on database data type such as a string cannot be used when the data type is integer and non-null field must have a value. It’s sufficient in most everyday use cases.

**Client Side Validation**

Starting version 5.5, users can use their own validation javascript function for more complex data edit rules. Use [set\_col\_customrule()](http://phpgrid.com/documentation/set_col_customrule/) method is created for this purpose. Below is an example of three Javascript functions used for custom validation. You can even compare data among multiple columns. The Javascript functions in example is rather over simplified, but you get the picture. Once obtained the cell value, you can develop even more complex Javascript function to validation your data.

**Server Side Validation**

For server side data validation, in your javascript function, use jQuery.ajax to call your server side validation routine. An example of Ajax call can be found on [Hyperlink to Call JavaScript Function](http://phpgrid.com/example/call-javascript-function-on-hyperlink-click/) example

Make sure that you play around with the live example after the code snippet!

**Javascript Functions:**

|  |  |
| --- | --- |
|  | // validation (FORM and INLINE) function price\_validation1(value, colname) {     if(value < 0){        return [false,colname + " must be zero a positive integer."];     }     return [true, ""]; }  // validation (INLINE only). Note the technique to obtain a specific cell value function price\_validation2(value, colname) {     var rowId = jQuery("#products").jqGrid('getGridParam','selrow');     if(parseFloat(jQuery('#' + rowId + '\_' + 'buyPrice').val()) >  parseFloat(jQuery('#' + rowId + '\_' + 'MSRP').val()))         return [false,"buyPrice must be equal or less than MSRP."];     else         return [true,""]; }  // validation (FORM only). Note the technique to obtain a specific cell value is different from INLINE edit. function price\_validation3(value, colname) {     if(parseFloat(jQuery('#buyPrice').val()) >  parseFloat(jQuery('#MSRP').val()))         return [false,"buyPrice must be equal or less than MSRP."];     else         return [true,""]; } |

**PHP**

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM products", "productCode", "products"); $dg->enable\_edit('FORM'); $dg->set\_col\_customrule('quantityInStock', 'price\_validation1'); $dg->set\_col\_customrule('buyPrice', 'price\_validation3'); $dg->display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/custom_validation.php?iframe=true&width=1040&height=700) (Try set quantityInStock to a negative number, and buyPrice < MSRP.)

[Comments ( 0 )](http://phpgrid.com/example/custom-data-validation/#respond)

[**Local Array Data Source**](http://phpgrid.com/example/local_array_data_source/)

by [Richard](http://phpgrid.com/author/richard/) on 11:44 am in [Examples](http://phpgrid.com/example/)

phpGrid now supports local array data source (version 5.5+). No database is required for local data. So **it’s NOT necessary to define** [**PHPGRID\_DB\_\* variables**](http://phpgrid.com/documentation/installation/) **in conf.php when using local array**. Simply pass a PHP array as the first parameter to the [phpGrid constructor](http://phpgrid.com/documentation/construct-our-phpgrid/). Everything else is virtually the same.

In the below example, the first segment creates a local PHP array, namely $data1, will be used as the data source for phpGrid. The second segment demonstrates passing the $data1 to the [phpGrid constructor](http://phpgrid.com/documentation/construct-our-phpgrid/) and call its methods. All existing phpGrid methods can be used the same way as a database-driven datagrid\*.

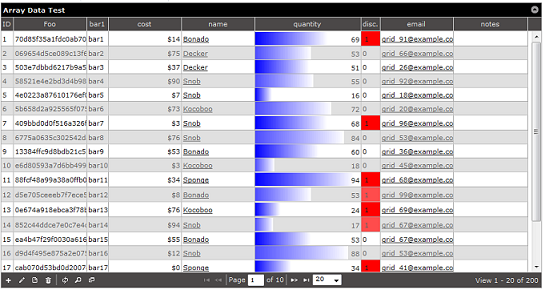
Make sure to check out the live example!

**Local Array (PHP)**

|  |  |
| --- | --- |
|  | $name = [array](http://www.php.net/array)('Bonado', 'Sponge', 'Decker', 'Snob', 'Kocoboo'); for ($i = 0; $i < 200; $i++) {     $data1[$i]['id']    = $i+1;     $data1[$i]['foo']    = [md5](http://www.php.net/md5)([rand](http://www.php.net/rand)(0, 10000));     $data1[$i]['bar1']    = 'bar'.($i+1);     $data1[$i]['bar2']    = 'bar'.($i+1);     $data1[$i]['cost']    = [rand](http://www.php.net/rand)(0, 100);     $data1[$i]['name']    = $name[[rand](http://www.php.net/rand)(0, 4)];     $data1[$i]['quantity'] = [rand](http://www.php.net/rand)(0, 100);     $data1[$i]['discontinued'] = [rand](http://www.php.net/rand)(0, 1);     $data1[$i]['email'] = 'grid\_'. [rand](http://www.php.net/rand)(0, 100) .'@example.com';     $data1[$i]['notes'] = ''; } |

**phpGrid Code** (**$data1** is the local array created above)

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid($data1, "id", "data1");   // $data1 is the local array $dg->set\_col\_title("id", "ID")->set\_col\_width('id', 20); $dg->set\_col\_title("foo", "Foo"); $dg->set\_col\_title("bar", "Bar"); $dg->set\_col\_title('discontinued', 'disc.')     ->set\_col\_width('discontinued', 35); $dg->set\_col\_align('cost', 'right')     ->set\_col\_currency('cost', '$'); $dg->set\_col\_width('bar1', 40); $dg->set\_col\_width('quantity', 220); $dg->set\_row\_color('lightblue', 'yellow', 'lightgray'); $dg->set\_databar('quantity', 'blue'); $dg->enable\_search(true); $dg->enable\_edit('FORM', 'CRUD'); $dg->enable\_resize(true); $dg->set\_col\_format('email', 'email'); $dg->set\_col\_dynalink('name', 'http://example.com', [array](http://www.php.net/array)("id", "name")); $dg->set\_caption('Array Data Test'); $dg->set\_col\_hidden('bar2'); $dg->set\_col\_property('notes', [array](http://www.php.net/array)('edittype'=>'textarea','editoptions'=>[array](http://www.php.net/array)('cols'=>40,'rows'=>10)))     ->set\_col\_wysiwyg('notes'); $dg->set\_dimension(900, 400); //$dg->set\_multiselect(true); $dg->set\_conditional\_value('discontinued', '==1',  [array](http://www.php.net/array)("TCellStyle"=>"tstyle")); $dg->set\_theme($theme\_name); $dg->display(); |

**Screenshot**  


**\* Please note that master detail, subgrid, export and file uploads are not yet available when using local array data source.**

**\*\* Local array data source currently is only available to Enterprise and Universal Licensees.**

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/local_array.php?iframe=true&width=940&height=500)

**JSON Data Source**

It’s also possible to use JSON string as a data source with *one* extra step: use [json\_decode](http://php.net/manual/en/function.json-decode.php) and set the second parameter to true to return the decoded JSON string to an associative array.

Once you have the array, it can be passed to the phpGrid constructor as if it’s a local array data source. This is useful when your data is real-time or loaded from a remote source such as stock quote and RSS etc.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 | $url = "http://myurl.com/json\_string"; $json = [file\_get\_contents](http://www.php.net/file_get_contents)($url); $json\_output = [json\_decode](http://www.php.net/json_decode)($json, true);  $dg = new C\_DataGrid($json\_output['items'], "id", "items"); $dg->display(); |

[Comments ( 0 )](http://phpgrid.com/example/local_array_data_source/#respond)

[**Column Freeze**](http://phpgrid.com/example/column-freeze/)

by [Richard](http://phpgrid.com/author/richard/) on 3:20 pm in [Examples](http://phpgrid.com/example/)

You can now use set\_col\_frozen method to set column freeze method. It’s useful when working with a big table with many columns. The freezing column must start from the very left and then one by one to the right.

|  |  |
| --- | --- |
| 1 2 3 4 | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->set\_dimension(600, 400, false); $dg->set\_col\_frozen('orderNumber'); $dg -> display(); |

Note that it is recommended to set text not to wrap with CSS “nowrap”. See <http://phpgrid.uservoice.com/knowledgebase/articles/154972>

[See Live Example!](http://phpdatagrid.com/examples/phpGridx/examples/column_freeze.php?iframe=true&width=750&height=500)

[Comments ( 0 )](http://phpgrid.com/example/column-freeze/#respond)

[**Column Methods**](http://phpgrid.com/example/column-methods/)

by [Richard](http://phpgrid.com/author/richard/) on 3:34 pm in [Examples](http://phpgrid.com/example/)

Version 5.5.5 introduced [set\_grid\_method()](http://phpgrid.com/documentation/set_grid_method/) method. You can use this method to call any jqGrid javascript method that can perform actions on the grid as a whole. However, it’s not possible to manipulate the grid on a row or cell level using [set\_grid\_method](http://phpgrid.com/documentation/set_grid_method/).

The example below demonstrates set\_grid\_method to grouping header by calling the jqGrid “[setGroupHeader](http://www.trirand.com/jqgridwiki/doku.php?id=wiki:groupingheadar)” method.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 13 14 | $dg = new C\_DataGrid("SELECT \* FROM payments", "customerNumber", "orders");  $dg->set\_col\_width("customerNumber", 30); $dg->set\_col\_width("checkNumber",50); $dg->set\_col\_width("amount",50);  $dg->set\_grid\_method('setGroupHeaders', [array](http://www.php.net/array)(                                 [array](http://www.php.net/array)('useColSpanStyle'=>true),                                 'groupHeaders'=>[array](http://www.php.net/array)(                                         [array](http://www.php.net/array)('startColumnName'=>'customerNumber',                                               'numberOfColumns'=>2,                                               'titleText'=>'Numbers Header') )));  $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/grouping_header.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/column-methods/#respond)

[**Inline Edit Actions Column**](http://phpgrid.com/example/inline-edit-actions-column/)

by [Richard](http://phpgrid.com/author/richard/) on 10:44 am in [Examples](http://phpgrid.com/example/)

Inline edit mode also comes with actions column with edit and delete buttons. Use [add\_column](http://phpgrid.com/documentation/add_column/) method to add “actions”. In formatoptions, specify the following property to display or hide edit and delete buttons:

* delbutton – true or false
* editbutton – true or false

Note that when use actions column, it is not necessary to specify edit options in enable\_edit method. An advantage using actions column is that it’s possible to have row-level permission. Please see [datagrid row-level edit permission example](http://phpgrid.com/example/row-level-permission-edit-condtion/).

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 10 11 12 | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders");  // enable edit - but no need to specify 2nd parameter because we will be using actions column $dg -> enable\_edit("INLINE");   $dg->add\_column("actions", [array](http://www.php.net/array)('name'=>'actions',     'index'=>'actions',     'width'=>'70',     'formatter'=>'actions',     'formatoptions'=>[array](http://www.php.net/array)('keys'=>true, 'editbutton'=>true, 'delbutton'=>false)),'Actions');  $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/inline_edit_action_column.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/inline-edit-actions-column/#respond)

[**Row-level Edit Permission**](http://phpgrid.com/example/row-level-permission-edit-condtion/)

by [Richard](http://phpgrid.com/author/richard/) on 11:05 am in [Examples](http://phpgrid.com/example/)

Using [actions column](http://phpgrid.com/example/inline-edit-actions-column/) in the earlier example, it’s now possible to specify row-level permission for editing using additional javascript.

The Javascript used here hides the delete and edit buttons based condition that when status equals to “Shipped”. The example shown is for demo purpose only. It’s NOT 100% secure. Programmer should still validate the permission on the server side.

For simpler condition without using Javascript, you can consider [set\_edit\_condition()](http://phpgrid.com/documentation/set_edit_condition/) method.

|  |  |
| --- | --- |
|  | // note this should not replace database role-based or user-based permissions. $onGridLoadComplete = <<<ONGRIDLOADCOMPLETE function(status, rowid) {     var ids = jQuery("#orders").jqGrid('getDataIDs');     for (var i = 0; i < ids.length; i++)     {         var rowId = ids[i];         var rowData = jQuery('#orders').jqGrid ('getRowData', rowId);          if($("#orders").jqGrid("getCell", rowId, "status") == "Shipped"){             $("#orders").jqGrid("setCell", rowId, "actions", " zzz ", {"display":"none"}); // not possible to set value for virtual column         }     } } ONGRIDLOADCOMPLETE;  $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->set\_col\_hidden('comments');  $dg->add\_column("actions", [array](http://www.php.net/array)('name'=>'actions',     'index'=>'actions',     'width'=>'80',     'formatter'=>'actions',     'formatoptions'=>[array](http://www.php.net/array)('keys'=>true)),'Actions'); $dg->set\_grid\_property([array](http://www.php.net/array)('onSelectRow'=>'')); // remove onSelect event $dg->add\_event("jqGridLoadComplete", $onGridLoadComplete); $dg->enable\_edit('INLINE');  $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/custom_edit_condition.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/row-level-permission-edit-condtion/#respond)

[**Multiple Datagrid Instances From the Same Table**](http://phpgrid.com/example/multiple-datagrid-instances-from-the-same-table/)

by [Richard](http://phpgrid.com/author/richard/) on 11:37 am in [Examples](http://phpgrid.com/example/)

You can certain have more than one datagrid instances from the same database table. Just make sure to give each unique name using [set\_jq\_gridName()](http://phpgrid.com/documentation/set_jq_gridname/) method.

|  |  |
| --- | --- |
| 1 2 3 4 5 6 7 8 9 | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg->enable\_edit(); $dg -> display();  $dg2 = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); $dg2->set\_jq\_gridName('order2'); $dg2->set\_caption("order 2"); $dg2->enable\_edit(); $dg2 -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/multiple_grids_single_dbtable.php?iframe=true&width=1040&height=700) (scroll to the right view the 2nd grid)

[Comments ( 0 )](http://phpgrid.com/example/multiple-datagrid-instances-from-the-same-table/#respond)

[**Column Chooser & Sortable Row**](http://phpgrid.com/example/column-chooser-sortable-row/)

by [Richard](http://phpgrid.com/author/richard/) on 12:33 pm in [Examples](http://phpgrid.com/example/)

The example below demonstrates column chooser and sortable row features. Click on the column chooser icon in the footbar to reveal the column chooser dialog. To test sortable row, drag a row and move it around.

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("select \* from products", "productCode", "products");  $dg -> enable\_columnchooser(true); $dg -> set\_sortablerow(true);  $dg -> display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/enable_columnchooser_sortablerow.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/column-chooser-sortable-row/#respond)

[**Composite Primary Key Support \***](http://phpgrid.com/example/composite-primary-key-support/)

by [Richard](http://phpgrid.com/author/richard/) on 5:47 pm in [Examples](http://phpgrid.com/example/)

*\* An exclusive feature for phpGrid Enterprise and Universal License*

In the past, phpGrid supports database SQL that only has a single primary key. If multiple primary keys exist, the recommended workaround was to create an auto-increment column as an alternative single primary key.

Though most cases the above solution is sufficient, as the database grows and its structure evolves, the need to support composite primary key becomes essential, especially in an enterprise environment.

Starting phpGrid 6 (Enterprise & Universal only), you can pass an array of string as the primary key as the $sql\_key in data grid [constructor](http://phpgrid.com/documentation/construct-our-phpgrid/). e.g.

|  |  |
| --- | --- |
|  | [array](http://www.php.net/array)("KEY1", "KEY2", "KEY3"...); |

Lots of resources devoted into developing this important feature while making it simple at the same time. In stead of passing a single string variable in the constructor as the primary key, you can now pass an array of strings as the composite primary key. For a single primary key, you can still use a string or an array with a single string value. e.g.

Complete composite primary key example:

|  |  |
| --- | --- |
|  | $dg = new C\_DataGrid("SELECT \* FROM orderdetails", [array](http://www.php.net/array)("productCode", "orderNumber"), "orderdetails"); $dg->enable\_edit("FORM", "CRUD"); $dg->display(); |

[**See Live Example!**](http://phpdatagrid.com/examples/phpGridx/examples/composite_pk.php?iframe=true&width=1040&height=700)

For single primary key, you can still pass it as string type, or an array with just a single string element.

|  |  |
| --- | --- |
| 1 | $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", "orders"); |

– OR –

|  |  |
| --- | --- |
| 1 | $dg = new C\_DataGrid("SELECT \* FROM orders", [array](http://www.php.net/array)("orderNumber"), "orders"); |

[Comments ( 0 )](http://phpgrid.com/example/composite-primary-key-support/#respond)

[**Multiple Datagrids Side by Side**](http://phpgrid.com/example/multiple-datagrids-side-by-side/)

by [Richard](http://phpgrid.com/author/richard/) on 3:02 pm in [Examples](http://phpgrid.com/example/)

When having multiple datgagrids on a page, it’s common you need to change the default top to bottom layout. The following code snippet shows how to use simple CSS to change layout to side by side.

Make sure to give enough width to datagrid container so that the grids won’t wrap.

|  |  |
| --- | --- |
|  | <div id="bigcontainer">      <div id="dg1">         <?php         $dg = new C\_DataGrid("SELECT \* FROM orders", "orderNumber", 'orders');         $dg->enable\_edit("FORM", "CRUD");         $dg->display();         ?>     </div>      <div id="dg2">         <?php         $dg2 = new C\_DataGrid("select \* from employees", "employeeNumber", "employees");         $dg2->enable\_edit("FORM", "CRUD");         $dg2->display();         ?>     </div>       <div id="dg3">         <?php         $dg3 = new C\_DataGrid("select \* from offices", "officeCode", "offices");         $dg3->enable\_edit("FORM", "CRUD");         $dg3->display();         ?>     </div>  < /div>  < style>     div#bigcontainer{         width: 5000px;     }     div#dg1, div#dg2, div#dg2, div#dg3{         float:left;     } < /style> |

[**Multiple datagrids side by side live demo!**](http://phpdatagrid.com/examples/phpGridx/examples/multiple_datagrids_sidebyside.php?iframe=true&width=1200&height=700)

[Comments ( 0 )](http://phpgrid.com/example/multiple-datagrids-side-by-side/#respond)

[**iOS, Android Mobile Support**](http://phpgrid.com/example/ios-android-mobile-support/)

by [Richard](http://phpgrid.com/author/richard/) on 8:33 pm in [Examples](http://phpgrid.com/example/)

phpGrid works well in iOS and Android mobile devices.

[](http://phpgrid.com/wp-content/uploads/2015/01/phpgrid_mobile_support.png)

Visit the following demo on your mobile device:  
[**Live demo**](http://www.phpdatagrid.com/examples/phpGridx/examples/php-datagrid-mobile.php?iframe=true&width=1040&height=700)

[Comments ( 0 )](http://phpgrid.com/example/ios-android-mobile-support/#respond)

[**Nested/Cascade/Dependent Drodown**](http://phpgrid.com/example/nested-dropdown/)

by [Richard](http://phpgrid.com/author/richard/) on 8:07 pm in [Examples](http://phpgrid.com/example/)

Use “[autocomplete](http://phpgrid.com/example/set_col_edittype-2/)” control and set\_nested\_dropdown() method, dependent dropdown, something called cascaded filter or nested dropdown, are also supported.

|  |  |
| --- | --- |
|  | $countryStateData =     [array](http://www.php.net/array)(         'usa' => [array](http://www.php.net/array)(             [array](http://www.php.net/array)('id'=>'',   'text'=>''),             [array](http://www.php.net/array)('id'=>'ca', 'text'=>'CA'),             [array](http://www.php.net/array)('id'=>'al', 'text'=>'AL'),             [array](http://www.php.net/array)('id'=>'nj', 'text'=>'NJ')          ),         'canada' => [array](http://www.php.net/array)(             [array](http://www.php.net/array)('id'=>'',   'text'=>''),             [array](http://www.php.net/array)('id'=>'ab', 'text'=>'AB'),             [array](http://www.php.net/array)('id'=>'qc', 'text'=>'QC'),             [array](http://www.php.net/array)('id'=>'bc', 'text'=>'BC')         )     );  $stateCityData =     [array](http://www.php.net/array)(         'ca' => [array](http://www.php.net/array)(             [array](http://www.php.net/array)('id'=>'',   'text'=>''),             [array](http://www.php.net/array)('id'=>'ca', 'text'=>'San Francisco'),             [array](http://www.php.net/array)('id'=>'al', 'text'=>'Los Angeles'),             [array](http://www.php.net/array)('id'=>'nj', 'text'=>'San Diego')          ),         'al' => [array](http://www.php.net/array)(             [array](http://www.php.net/array)('id'=>'',   'text'=>''),             [array](http://www.php.net/array)('id'=>'ab', 'text'=>'test'),             [array](http://www.php.net/array)('id'=>'qc', 'text'=>'test2'),             [array](http://www.php.net/array)('id'=>'bc', 'text'=>'city2')         ),         'bc' => [array](http://www.php.net/array)(             [array](http://www.php.net/array)('id'=>'',   'text'=>''),             [array](http://www.php.net/array)('id'=>'ab', 'text'=>'2342344'),             [array](http://www.php.net/array)('id'=>'qc', 'text'=>'bc city2'),             [array](http://www.php.net/array)('id'=>'bc', 'text'=>'bc city333')         )     ); echo '<script>'; echo 'var countryStateData = '. [json\_encode](http://www.php.net/json_encode)($countryStateData).";\n"; echo 'var stateCityData = '. [json\_encode](http://www.php.net/json_encode)($stateCityData).";\n"; echo '</script>';  $dg = new C\_DataGrid('select \* from customers', 'customerNumber', 'customers'); $dg->enable\_edit('INLINE'); $dg->set\_col\_edittype('country', 'autocomplete', ':;usa:USA;canada:Canada;France:France;Germany:Germany;Norway:Norway;Poland:Poland;Australia:Australia;Spain:Spain;Denmark:Denmark;Singapore:Singapore;Belgium:Belgium;Finland:Finland;New Zealand:New Zealand;Italy:Italy;Japan:Japan;Irelan:Ireland;Hong Kong:Hong Kong;Russia:Russia;Israel:Israel'); $dg->set\_col\_edittype('state',   'autocomplete', ':;bc:bc;ab:ab;al:al;bc:bc;nj:NJ;CA:CA;Co. Cork:Co. Cork;CT:CT;Isle of Wight:Isle of Wight;MA:MA;NH:NH;NSW:NSW;NV:NV;NY:NY;Osaka:Osaka;PA:PA;Pretoria:Pretoria;Québec:Québec;Queensland:Queensland;Tokyo:Tokyo;Victoria:Victoria'); $dg->set\_nested\_dropdown('country', 'state', 'countryStateData');  $dg->display(); |

Programming Note:

* The example demostrate dependent dropdown, or sometimes called nested dropdown. It can be nested in unlimited level.
* It’s important that all the dropdowns must be set to “autocomplete” edit type before calling set\_nested\_dropdown() method.
* Due to the way the select is implemented in jqGrid, the 3rd parameter of [set\_col\_edittype()](http://phpgrid.com/documentation/set_col_edittype/) method must contain the ENTIRE string of all the key:value pair of that column, or the data will appear to be missing when display. There is no way around due to the jqGrid limitation.
* The dependency logic is essentially embedded in the javascript data source (e.g. $countryStateData) used to populate the dropdown dynamically based on its selected parent value.
* Nested dropdown only works in INLINE edit at the moment.

In a nutshell, the phpGrid displays the datagrid rendered by jqGrid. Select2 does the autocomplete and nested dropdown completely on the client side.   
which is also the reason why the data javascript data source must be structured correctly.