

Forum:[Hack Forums](#)

Thread: [My PHP table generation class.](#)

glitchgxd - August 5, 2012, 7:25 am

It's probably not that professional, but I use it often.

class_table.php

Code <?php

```
class Table
{
    public $_head = array();

    public $_cells = array();

    public $_rows = array();

    function create_html($info=array())
    {
        $table = "<table";
        if(!empty($info) && is_array($info))
        {
            foreach($info as $a => $b) $table .= " $a=\"$b\"";
        }
        $table .= ">";
        if(!empty($this->_head))
        {
            $table .= '<thead';
            if(!empty($this->_head['extra']) && is_array($this->_head['extra']))
            {
                foreach($this->_head['extra'] as $c => $d) $table .= " $c=\"$d\"";
            }
            $table .= ">";
            foreach($this->_head['cells'] as $j => $cell)
            {
                $table .= "<td";
                if(!empty($cell['extra']) && is_array($cell['extra']))
                {
                    foreach($cell['extra'] as $e => $f) $table .= " $e=\"$f\"";
                }
                $table .= ">";
                $table .= "$cell[data]";
                $table .= "</td>";
            }
            $table .= "</thead>";
        }
        if(!empty($this->_rows))
        {
            $table .= '<tbody>';
```

```

        foreach($this->_rows as $i => $data)
        {
$stable .= "<tr";
if(!empty($data['extra']) && is_array($data['extra']))
{
foreach($data['extra'] as $c => $d) $stable .= " $c=\"\$d\"";
}
$stable .= ">";
foreach($this->_rows[$i]['cells'] as $j => $cell)
{
$stable .= "<td";
if(!empty($cell['extra']) && is_array($cell['extra']))
{
foreach($cell['extra'] as $e => $f) $stable .= " $e=\"\$f\"";
}
$stable .= ">";
$stable .= "$cell[data]";
$stable .= "</td>";
}
$stable .= "</tr>";
        }
        $stable .= '</tbody>';
    }
    $stable .="</table>";
    $this->_head = $this->_rows = $this->_cells = array(); // Clear current cells and rows.
    return $stable;
}

```

```

function make_head($extra=array(), $style=array())
{
    if(!empty($style) && is_array($style))
    {
        $extra['style'] = "";
        foreach($style as $k => $v)
$extra['style'] .= "{$k}: {$v};";
    }
    $this->_head = array("cells" => $this->_cells, "extra" => $extra);
    $this->_cells = array(); // Clear the current cells.
}

```

```

// _rows[][cells][data]
function make_row($extra=array(), $style=array())
{
    if(!empty($style) && is_array($style))
    {
        $extra['style'] = "";
        foreach($style as $k => $v)
$extra['style'] .= "{$k}: {$v};";
    }
    $this->_rows[] = array("cells" => $this->_cells, "extra" => $extra);
    $this->_cells = array(); // Clear the current cells.
}

```

```

// _cells[['data']]
// _cells[['extra']]
function make_cell($data="", $extra=array(), $style=array())
{
    if(!empty($style) && is_array($style))
    {
        $extra['style'] = "";
        foreach($style as $k => $v)
        $extra['style'] .= "{$k}: {$v};";
    }
    $this->_cells[] = array("data" => $data, "extra" => $extra);
}
}

```

?>

So, to use this, you'll have to include the table script into your script. Then you can start creating your table. For instance, if I wanted to make a table like this:

```

Code <table class="cool">
  <thead>
    <td style="font-weight: bold">Title</td>
    <td>Details</td>
  </thead>
  <tr>
    <td colspan="2">No information</td>
  </tr>
  <tr>
    <td>Some title</td>
    <td>Some details</td>
  </tr>
</table>

```

You would use the following code:

Code <?php

```

require_once 'class_tables.php';
$table = new Table;

$table->make_cell('Title', array('style' => 'font-weight: bold'));
$table->make_cell('Details');
$table->make_head();

$table->make_cell('No information', array('colspan' => 2));
$table->make_row();

$table->make_cell('Some title');
$table->make_cell('Some details');
$table->make_row();

echo $table->create_html(array('class' => 'cool'));

```

?>

Now of course you could implement this with a variety of things (maybe a loop to generate your

tables). It's up to you. Let me know what you think!

Enjoy.