

Bazaar – php example code – part 10 – pagination in list output

Article will focus on way how to create pagination functionality for larger list of displayed data (list of items for sell, list of registered users, cart items, ...). Our implementation will be portable among a pages or other projects, visually attractive and ergonomic.

Expectation from pagination functionality

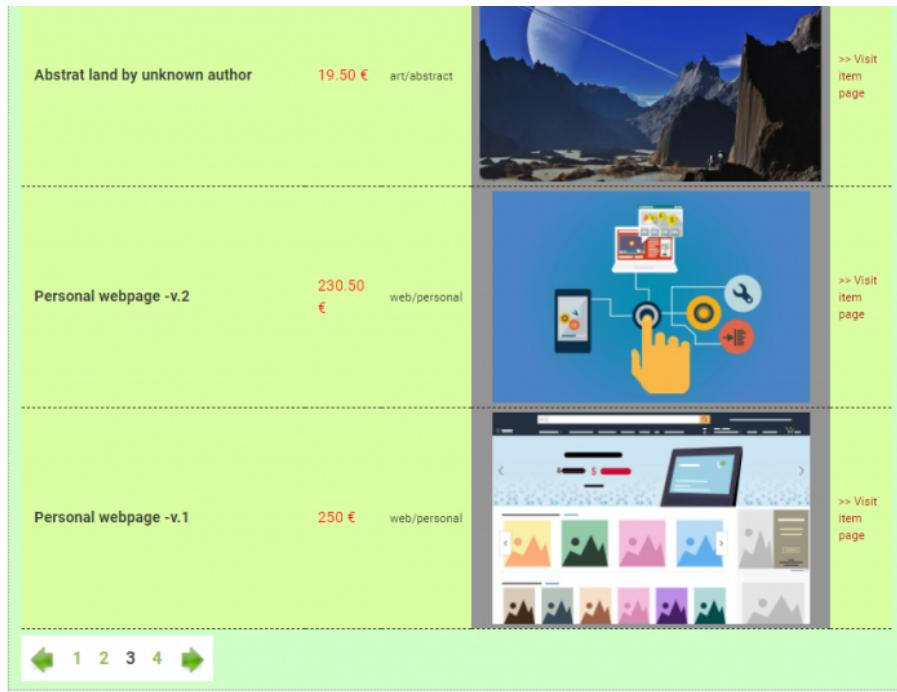
Larger list output of data can be for user a bit overwhelming. Best way how to break output into a separate chunks of data is paginate them with support of separate numbered page links.

For our implementation of pagination mechanism is expected:

- user can define or in code is hardcoded number of items per page, this is held in \$number_per_page variable
- page links must be shown as numbered hyperlinks in bottom left part of data lists
- arrows for next and previous page must be present with its graphical representation
- separate function for generating pagination link will be created (its name is functions.php and appropriate function is named
`function generate_page_links($user_search, $sort_by, $order, $cur_page, $num_pages) { //($user_search, $sort_by, $order, $cur_page, $num_pages);`

Our expectation how pagination links will look like show next

picture:



Pagination links in bottom part of a item list on main Bazaar page

Code for page links generation

Function for paging links creation is inserted into a functions.php script. Full content of mentioned function follows:

```
function generate_page_links($user_search, $sort_by, $order, $cur_page, $num_pages) { //($user_search, $sort_by, $order, $cur_page, $num_pages);
    $page_links = "";
    echo "<br>";

    // if this is not first in row, we need generate the „previous“ link
    if ($cur_page > 1) {
```

```

    $page_links .= ,<a id="pagination" href=" . $_SERVER[PHP_SELF'] . ,?usersearch='
        .$user_search . ,&sort_by=' . $sort_by . ,&order=' . $order . '&page=' . ($cur_page - 1) . ,></a>';
    } else {
        $page_links .= ,<span id="pagination"></span> ;
    }
    // Loop through the pages generating the page numbered links
    for($i = 1; $i <= $num_pages; $i++) {
        if ($cur_page == $i) {
            $page_links .= ,<span id="pagination">' . $i. ,</span>';
        // span inline element mark non a tag (unlinked number) as pagination for further formating by css
        } else {
            $page_links .= ,<a id="pagination" href=" . $_SERVER[PHP_SELF'] . ,?usersearch='
                .$user_search . ,&sort_by=' . $sort_by . ,&order=' . $order . '&page=' . $i . ,>' . $i . ,</a>';
        }
    }
    // If this page is not last in row, generate „next“ link
    if ($cur_page < $num_pages) {
        $page_links .= ,<a id="pagination" href=" . $_SERVER[PHP_SELF'] . ,?usersearch='
            .$user_search . ,&sort_by=' . $sort_by . ,&order=' . $order . '&page=' . ($cur_page + 1) . ,></a>';
    } else {
        $page_links .= , <span id="pagination"></span>';
    }

    return $page_links;
}

```

Our function can generate also user_search if needed for providing optional chunk of information through GET able links.

In main implementation page we take closer look in a way how to display paginated outputs.

Way how to implement pagination in a main page code

For first approach we will show you code implementing pagination with calling generate_page_links(\$user_search, \$sort_by, \$order, \$cur_page, \$num_pages) function.

As a comments are displayed necessary leadings for understanding our code. Next code sniped is selected from main Bazaar page, from bottom list of items for sell (permanently displayed part, not part generated by user selected category – upper part is optionally displayed).

```
*****
*/
/* Output in paginated form
 */
*****
*/
/**
 * Display pagination on the page – part included to listening in this area
 */
/* Attempt MySQL server connection. Assuming you are running M
```

```

ySQL
*/
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME);
//GET data for pagination send to page herself
//calculate pagination information
$cur_page = isset($_GET[,page']) ? $_GET[,page'] : 1;
// results per page default declarer as 5 on top of page and c
hanged in submitt part after reset button handling $results_pe
r_page = 5;
$skip = (( $cur_page -1) * $results_per_page);
// first question to database table for obtaining number of p
ublished items in a database – obtain value for $total
$sql = „SELECT * FROM bazaar_item WHERE published=“.,1’’.” AN
D cart_number=“.,0’’.” ORDER BY item_id DESC „; // read in r
everse order of score – highest score first
$output_for_number_rows_count = mysqli_query($dbc, $sql); // q
uery database
$total = mysqli_num_rows($output_for_number_rows_count); // get
number of rows in database

... omitted part of code enabling ordering along selected
category
if(( $sort_by == „name“) && ($order == „1“)) { // along name an
d ASC order
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1’’.” AND cart_number=“.,0’’.” ORDER BY name_of_item ASC LIMIT
$skip, $results_per_page“;
}
if(( $sort_by == „name“) && ($order == „-1“)) { // along na
me and DESC order
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1’’.” AND cart_number=“.,0’’.” ORDER BY name_of_item DESC LIMIT
$skip, $results_per_page“;
}
if(( $sort_by == „price“) && ($order == „1“)) { // along pr
ice and ASC order
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1’’.” AND cart_number=“.,0’’.” ORDER BY price_eur ASC LIMIT $skip,
$results_per_page“;
}
if(( $sort_by == „price“) && ($order == „-1“)) { // along p

```

```

rice and DESC order
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1‘.“ AND cart_number=“.,0‘.“ ORDER BY price_eur DESC LIMIT
$skip, $results_per_page“;
}
if(($sort_by == „category“) && ($order == „1“)) { // along
category and ASC order
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1‘.“ AND cart_number=“.,0‘.“ ORDER BY subcategory_id ASC LI
MIT $skip, $results_per_page“;
}
if(($sort_by == „category“) && ($order == „-1“)) { // alon
g category and DESC order
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1‘.“ AND cart_number=“.,0‘.“ ORDER BY subcategory_id DESC L
IMIT $skip, $results_per_page“;
}
if(($sort_by == „default“)) { // along category and DESC o
rder
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,
1‘.“ AND cart_number=“.,0‘.“ ORDER BY item_id DESC LIMIT $s
kip, $results_per_page“;
}
} else { // first run without ordering – no get link generate
d
    $sql = „SELECT * FROM bazaar_item WHERE published=“.,1‘“.
“ AND cart_number=“.,0‘.“ ORDER BY item_id DESC LIMIT $skip,
$results_per_page“; // read in reverse order of score – high
est score first
}

*****
*/
/* Output in Table – solution 1 – for debugging data from
database */
*****
*****
```

echo „

“;

echo „<h4>Latest added items for you! </h4>“;

echo „
“;

```

/***
 * Obtaining wished number of item per page – option for select
 */
?>
<form method="post" action="php echo $_SERVER[ ,PHP_SELF' ];
?&gt;"&gt;

&lt;div class="form-group"&gt;
&lt;label&gt; Set expected number of items per page -5 is default:&lt;/label&gt;
&lt;input list="number_per_page" name="number_per_page" placeholder="please select or write nr."&gt;
    &lt;datalist id="number_per_page" &gt; <!-- must be converted in subcategory_id in script – marked with (*) --&gt;
        &lt;option value="5"&gt;
        &lt;option value="10"&gt;
        &lt;option value="15"&gt;
        &lt;option value="20"&gt;
        &lt;option value="50"&gt;
        &lt;option value="100"&gt;
    &lt;/datalist&gt;

    &lt;!-- users_id from session obtaining – for debugging and testing is set as hidden --&gt;

    &lt;button type="submit" name="nr_of_pages" class="btn btn-warning"&gt; Use selected number of pages! &lt;/button&gt;
&lt;/div&gt;
&lt;/form&gt;
&lt;?php
echo „&lt;br&gt;“; echo „&lt;br&gt;“;
if($output = mysqli_query($dbc, $sql)){
    if(mysqli_num_rows($output) &gt; 0){ // if any record obtained from SELECT query
        // create table output
        echo „&lt;table&gt;“; //head of table
        echo „&lt;tr&gt;“;
            //echo „&lt;th&gt;id&lt;/th&gt;“;
</pre

```

```

        // functionality for ordering result
        /**
         * SORTING – PART I. Here are generated GET
links for UP/DOWN ordering by appropriate category – not
pertinent to our explantation related to pagination
functionality
        */
echo ,<th>Name <br /><a id="SORT" href="index.php?sort_by=name&order=1">  </a>
                <a id="SORT" href="index.php?sort_by=name&order=-1">  </a> </th>';
//order 1 up -1 down
echo ,<th>Price <br /><a id="SORT" href="index.php?sort_by=price&order=1">  </a>
                <a id="SORT" href="index.php?sort_by=price&order=-1">  </a></th>';
echo ,<th>Category <br /><a id="SORT" href="index.php?sort_by=category&order=1">  </a>
                <a id="SORT" href="index.php?sort_by=category&order=-1">  </a> </th>';
echo „<th>Screenshot</th>“;
echo „<th>More info</th>“;

echo „</tr>“;
while($row = mysqli_fetch_array($output)){ //next rows
outputed in while loop
    echo " <div class=\"mailinglist\"> " ;
    echo „<tr>“;
        //echo „<td>“ . $row[,item_id'] . „</td>“;
        echo „<td class=\"item_name\">“ . $row[,name_of_item'] . „</td>“;
            echo „<td class=\"price\">“ . $row[,price_eur'] . “ € </td>“;
                    /* convert category_id in to category and subcategory */

```



```

                // Free result set
                mysqli_free_result($output_supl);
            } else {
                echo „There is no source
h
subcategory in category table. Please correct your error.“; // if no records in table
            }
        } else{
            echo „ERROR: Could not able to execute $sql. “ . mysqli_error($dbc); // if database query problem
        }
        echo „<td>“ . $category_idsupl.”/”. $subcategory_idsupl.”</td>“;

                $image_location = IMAGE_PATH.$row[,screenshot1'];
                echo „<td id=\“gray_under_picture\“> <img src=\“$image_location\“ alt=\“ screenshot of product primary \“ height=\“250\“> </td>“;
                echo ,<td colspan=“1”><a id=“DEL“ href=“item.php?item_id=’. $row[,item_id’]. ,“><img id=“next“ src=“./images/next.png“> </a></td></tr>‘; //construction of GETable link
                echo „</tr>“;
                echo “ </div> “ ;
            }
            echo „</table>“;
            //count nuber of pages total
            $num_pages = ceil($total / $results_per_page);

            //generate navigational page links if we have more than one page

            if($num_pages > 1) {
                $user_search = „“; // not implemented yet, then set as clear values
                if(empty($sort_by)) { // if not obtained by get then default order is applied
                    $sort_by=“default“;

```

```

    };
    if(empty($order)) { // if not obtained by get then
default order is applied
        $order="1";
    };

        // included function for pagination generation fun
ction stored in functions.php page
        echo generate_page_links($user_search, $sort_by, $
order, $cur_page, $num_pages);
        echo „<br><br>“;
    }
    // Free result set
    mysqli_free_result($output);
} else{
    echo „There is no item for sell. Please add one.“; //
if no records in table
}
} else{
    echo „ERROR: Could not able to execute $sql. “ . mysqli_e
rror($dbc); // if database query problem
}
// Close connection
mysqli_close($dbc);
?>

```

By this way we can implement pagination also in admin page. If two or more ordering are present on page, only one selected number of page is mandatory for all lists. it means, if we selected page nr. 2 in one list, also other list are on page 2 after reloading page with appropriate GET links is generated for requesting that page for display.

This is not bad solution of most of our needs for simplifying information displayed in one time.

Conclusion

Pagination is a way how to limit number off information visible to a user in given time. For our solution we used selection with LIMIT functionality and further mechanism for reading total number of items for display. After necessary calculation, links for pagination are generated and with help of GET request, page obtain all information important for display only that part of selection, that is needed for that moment.

Full working code of bazaar training app can be downloaded from github [here](#).

Main parts are implemented in as function in functions.php where is function for link generation. Index.php or admin.php are great examples how to implement two or more pagination per single page. This simple solution has its own limitation as it was mentioned before.