

## Randomize group order in LimeSurvey 1.7x

---

In [this thread](#), a user suggested a method to randomize the order of groups in a survey. He or she gave some clues about how to do so, but did not go into every detail with respect to actual code changes.

When one of my surveys required the exact same functionality, I sat down and implemented a working solution. Below is what I've done to randomize groups in a survey, closely based on the outline provided by user mx.2000. Maybe it is helpful to some of you.

The thing I wanted to accomplish was to randomize *sequences of groups* in a survey. This is a more refined variant of global randomization because you may have a situation like this:

```
demographics
itemgroup1
itemgroup2
itemgroup3
interlude
itemgroup4
itemgroup5
itemgroup6
closingcomments
```

where you separately want to randomize the order of itemgroup1...3 and itemgroup4...6, without changing the position of the other three groups.

Since I did not want to mess around with the code base too much I opted for a workaround which is not specifically elegant but does the trick until the randomization feature is officially implemented with LimeSurvey. The solution I went for was to indicate randomization by means of a suffix to the group name ("`_rand()`"). Hence, in the above example, one would need to give the groups in the sequences to be randomized the following names:

```
demographics
itemgroup1_rand()
itemgroup2_rand()
itemgroup3_rand()
interlude
itemgroup4_rand()
itemgroup5_rand()
itemgroup6_rand()
closingcomments
```

My code modification just searches the group names for a trailing "`_rand()`" and randomizes the order of all subsequent groups carrying this suffix. Groups without the suffix maintain their position within the survey.

The only file that has to be modified is `index.php` in the LimeSurvey root directory. The code has to be amended in two places.

1. Add a function `RandomizeSessionGroupList()` at line 2307 in `index.php`, which is right before the function `UpdateSessionGroupList()`, containing the following code:

```

function RandomizeSessionGroupList()
{
    $scratcharr = array();
    $randarr = array();

    if (isset($_SESSION['grouplist']))
    {
        foreach ($_SESSION['grouplist'] as $group)
        {
            if (strpos(rtrim($group[1]), '_rand()') == (strlen($group[1])-7))
            {
                $randarr[] = $group;
            }
            else
            {
                if (count($randarr)>0)
                {
                    shuffle($randarr);
                    $scratcharr = array_merge($scratcharr, $randarr);
                    $randarr = array();
                }

                $scratcharr[] = $group;
            }
        }

        if (count($randarr)>0)
        {
            shuffle($randarr);
            $scratcharr = array_merge($scratcharr, $randarr);
            $randarr = array();
        }

        $_SESSION['grouplist'] = $scratcharr;
    }
}

```

2. Add a call to this function to UpdateSessionGroupList() in line 2359 of index.php. The function now looks like this:

```

function UpdateSessionGroupList($language)
//1. SESSION VARIABLE: grouplist
//A list of groups in this survey, ordered by group name.
{
    global $surveyid;
    unset ($_SESSION['grouplist']);
    $query = "SELECT * FROM ".db_table_name('groups')." WHERE sid=$surveyid
        AND language='".$language.'" ORDER BY
        ".db_table_name('groups')."group_order";
    $result = db_execute_assoc($query) or safe_die ("Couldn't get group
        list<br />$query<br />".$connect->ErrorMsg()); //Checked
    while ($row = $result->FetchRow())
    {
        $_SESSION['grouplist'][]=array($row['gid'], $row['group_name'],
            $row['description']);
    }

    // The line below needs to be added in order to randomize group order
    RandomizeSessionGroupList();
}

```

As advised by a moderator in the aforementioned thread, one ought not use conditions on randomized groups.

Best wishes

Malte