

## Re: Making dynamic table sortable

**Source:** <http://coding.derkeiler.com/Archive/PHP/comp.lang.php/2005-01/2406.html>

---

**From:** Sean ([sdemerchREMOVE\\_at\\_REMOVEhotmail.com](mailto:sdemerchREMOVE_at_REMOVEhotmail.com))

**Date:** 01/30/05

Date: Sun, 30 Jan 2005 06:50:49 -0800

On Sun, 30 Jan 2005 10:01:01 +0000, Geoff Berrow  
<blthecat@ckdog.co.uk> reverently intoned upon the aether:

> *I noticed that Message-ID: <2dfov09mnelj2863punlsk45rm2ft7hkc9@4ax.com>  
> from Sean contained the following:*  
>  
> >> *I may be wrong but I thought that MySql would not allow this. Must do  
> >> some tests.*  
> >  
> >*I ran some quick tests (I am tired from landscaping the yard and  
> >hungry, so I do not trust my mind at the moment) and could not pull  
> >off multiple SQL queries in a mysql\_query call(), but that does not  
> >rule out the potential for UNIONS and other injection techniques.*  
>  
> *You are heading for SQL territory in which I am unfamiliar. Still I'm  
> glad I was right about one thing :-)*  
>  
> *It can be very confusing to self confessed newcomers to a language to as  
> a simple question and then be confronted with 20 lines of unfamiliar  
> code and I try to keep help as simple as possible.*

The flip side of which results is websites filled with SQL injections and XSS (cross site scripting) vulnerabilities all over the net. Sometimes doing something so it functions is not enough. Especially where credit card data or other sensitive information is handled. But it is also unwise when little or no backups are done.

Outsourcing hosting to a commercial host should get you daily incrementals and weekly full backups or something similar. But can you expect this on a free account from your ISP or a friends system? I would not.

I should admit I am biased and have some experience doing software engineering on fault tolerant operations systems so I have had checking every return value and doing something about any error state beaten into me.

> *But you are right about not trusting user input.*

I am fond of the "users are evil" statement. I know I am a nasty user, I expect things to work the way they should, not the way the manual I did not read says it works.

> *Rather than if/else or switch I like to*  
> *use arrays. It enables you to set up all choices on one line, but can*  
> *look a bit confusing to someone new to PHP*  
>  
> `$sort_order=array("Year"=>"Year","Type"=>"Type"...);`  
>  
> *then use*  
>  
> `$sql = "SELECT * FROM albums ORDER BY $sort_order [$_GET['order']]";`  
>  
> *Can you see any problems with that?*

If the GET value is invalid, then you get:

```
$sql = "SELECT * FROM albums ORDER BY ";
```

Which is likely to be invalid SQL.

Instead, you might try:

```
$sort_order = array("Year" => " ORDER BY YEAR", ...)
```

So that the SQL is embedded in the array so that an invalid input yields:

```
$sql = "SELECT * FROM albums";
```

And of course you lost the ability to catch an error if it occurs. I guess one could check \$sql for validity in your original code to ameliorate this.

```
if( $sql === "SELECT * FROM albums ORDER BY " ){  
    // Could be a hacking attempt, could be a request that  
    // was corrupted in transit. Either way, log it so we can  
    // diagnose the situation later.  
    log_message("Illegal GET value $_GET[order]");  
    $sql = "SELECT * FROM albums ORDER BY Album";  
}
```

But I think simply checking the input with a conditional before constructing the query is easier and allows you to more clearly handle error conditions. Shorter code is not necessarily clearer or easier to maintain. And checking the input first, rather than the query after the fact shows in the logic that it is the input (GET value) that is the issue rather than the query. You might just thank yourself for separating such logic six months down the road.

some thoughts,

Sean :o)

"In the End, we will remember not the words of our enemies,  
but the silence of our friends."

– Martin Luther King Jr. (1929–1968)

Photo Archive @ <http://www.tearnet.com/Sean>

Last Updated 29 Sept. 2004