

# You Should Use an Alias for Those Sub Query Tables!

Posted At : March 15, 2006 4:04 PM | Posted By : Mark Kruger

Related Categories: SQL tips, Coldfusion & Databases

Here's a rule of thumb for you. When you are writing a query that includes a sub query always use an alias for the table and qualify the columns - or use the full column name if you are so inclined. If you don't you may very well end up with unexpected results. Consider this query:

```
<cfquery name="getUsers" datasource="#myDsn#">
  SELECT    Username, fullname
  FROM      users
  WHERE     username IN
            (SELECT username
             FROM    groupsUsers
             WHERE   groupId = 5)
</cfquery>
```

Let's just say that the table "groupsUsers" is a cross reference (or bridge) table containing 2 columns, userid and groupid. With that piece of information let's have a quick show of cyber-hands - how many of you think this query will work (and by *work* I mean actually return values instead of errors)? Not too many of you right? That shows how sophisticated muse readers actually are. It definitely looks like a stinker because there is no "username" column in the "groupsUsers" table - right? So it should error out with a "column not found" type of error. Ah... not so fast.

## It Works!

Actually the query does work - though not as intended. You see, because the column name "username" is in the "users" table, when the DB server processes the query it says to itself... "username, username.... let's see, do I have a username column somewhere?" And because you have not specified the table name or an alias it says, "Ah ha! ...here's a username in the users table.". This has the effect of selecting the username from the users table for every row in the users table. To put it another way, the query above is actually doing this:

```
<cfquery name="getUsers" datasource="#myDsn#">
  SELECT    Username, fullname
  FROM      users
  WHERE     username = username
</cfquery>
```

The result? Every row in the users table will be returned.

## The fix

The fix is to specify the table using an alias or full table qualifier. That's good practice in all but the simplest applications. If you had done that in the example above...

```
<cfquery name="getUsers" datasource="#myDsn#">
  SELECT    u.Username, u.fullname
  FROM      users u
  WHERE     u.username IN
            (SELECT gu.username
             FROM    groupsUsers gu)
</cfquery>
```

```
WHERE    gu.groupId = 5)
</cfquery>
```

... It would have immediately thrown an error and you would have rewritten it as:

```
<cfquery name="getUsers" datasource="#myDsn#">
  SELECT    u.Username, u.fullname
  FROM      users u
  WHERE     u.userid IN
            (SELECT gu.userid
             FROM    groupsUsers gu
             WHERE   gu.groupId = 5)
</cfquery>
```

This is yet another example where following some "best practices" will help you avoid mistakes. It would be pretty easy to let the query above slip by because it does actually return records. Hopefully QA and process testing would ferret out such a mistake, but you would probably end up in a straight jacket trying to find and fix the query. Happy coding.