

**CIS 180 L - Intro to PHP - Fall 2006
Homework Assignment #5**

Due: THURSDAY, November 16th, beginning of class

Purpose: practice with PHP and Oracle

Note: as long as you meet the specifications below, you may add additional embellishments as you wish.

How to turn this in: See instructions in #HERE PLEASE below - you will e-mail me the URL's for your completed pages.

1. Consider **try_oracle1.php** . Use it as a basis to create a modified version **try_oracle2.php**, which meets the following specifications:

- * For now, be sure to put this on **sorrel**, in your **public_html** directory; there's currently a problem with **OCILogon** on redwood.
- * To the initial form (with the username and password inputs), add three radio buttons, one for each of **hiredate** and **salary** and **commission**, along with text asking the user to select the column whose contents he/she wants to see.
- * When the form is submitted, it then queries only for the column specified by the selected radio button -- it only displays the **hiredate** or the **salary** or the **commission** column of the **empl** table.
- * IMPORTANT TIDBIT: While SQL is not usually case-sensitive, it does seem to be the case that OCIResult requires the table column name to be uppercase.

But - if you've received a form value in lowercase, how can you convert it to uppercase?

Well, you could make the form input value uppercase. But I think it is more fun to get an uppercase version of a string \$myString using the function **strtoupper** : **strtoupper(\$myString)**

(that is, **strtoupper("hi") == "HI"**
strtoupper("Bye") == "BYE"
...and so on.)

2. Of course, once you access a database, you might do more with that data than just print it to the screen.

For a little experience in this direction, create **try_oracle3.php**, which meets these specifications:

- * This should also be on **sorrel**, in your **public_html** directory.
- * To the initial form (with the username and password inputs), include two radio buttons, one for **salary** and one for **commission**, along with text asking the user to select the column whose **sum** he/she wants to see.
- * When the form is submitted, it queries only for the column specified by the selected radio button -- but instead of displaying that column, it adds up all of the values in that column, and prints to the screen the sum of that column (in a readable, clear message).

3. When your pages are ready, send me an e-mail to **sharon.tuttle@humboldt.edu** before the deadline with **CIS 180 PHP HW 5** in the Subject: line containing the URL's for your **php** pages on **sorrel** .