**Week 9, PHP and MySQL. -**   
  
Things to do this week:

Reading Quiz 8.

* Read the lecture notes - PHP and MySQL.
* Read chapter 4 in the text.
* Complete Reading Quiz 8.

Lab 8

* Download and unzip the starting files - PHP and MySQL.
* Read the description of Lab 8.
* Solve each of the problems as described in the lab.
* Participate in the forum - Lab 8 Questions as necessary.
* Submit lab 8.

**Lab 8 Instructions -**

The objective of this lab is to introduce you to the creation of simple data driven applications that combine MySQL and PHP. In the course of building the applications you will be introduced to the basic syntax for using PDO. This information is in chapter 4 of your text.

1. Download the starting files for lab 8 and "install" the Product Viewer and Product Manager applications from chapter 4 on your machine. Both applications require the installation of the my\_guitar\_shop databases from chapter 3. Run both applications and familiarize yourself with how each works.

2.Examine the code in the Product View application. Pay particular attention to the use of the PDO class to work with the database.

3. Improve the Product Viewer application by writing the following functions. Write the functions in a separate file. Test each function as you write it. Use try / catch block to deal with errors:

* getConnection - no parameters. Returns an open connection to the database
* getCategory - takes a database connection and a category id as parameters. Returns an array that represents one category.
* getCategories - takes a database connection as a parameter and returns a 2d array that represents all of the categories.
* getProductsByCategory - takes a database connection and a category id as parameters. Returns a a 2d array that represents all of the products in one category.

4. Improve the Product Viewer application by "applying" the MVC design pattern.

* Identify all of the data in the model. Use the functions you just wrote to gather that data in the index page.
* Identify the 2 pages that make up the view. Separate the view(s) from the index page that contains the controller logic.
* Re-write the index page so that it only contains the controller functionality. It should call your functions to generate the model and display (one of) the view.

5. Repeat this process CONCEPTUALLY to improve the Product Manager application. In small groups

* identify how you would divide the application code into functions
* identify how you would implement the MVC design pattern in this application

In class section students should:

* Submit a text file or word processing document in moodle that contains your source code for the improved Product Viewer application as well as your suggestions for improving the Product Manager application.