CIM 225 - Syllabus - Saddleback College
Introduction to PHP/MySQL

Class No: 20120

The FINAL DATE of this class is Dec. 16, 2005
No assignments or exams will be accepted after that date!

INSTRUCTOR

Steve Perry
Office Hours: N/A
Phone: N/A (Email me!)

Email: steveperrymail@yahoo.com

I check my email account at least once per day, Monday-Friday, usually in
the daytime. I generally do not check for emails on Saturday, Sunday, and Holidays.

Emergency email: sperry@palomar.edu

COURSE DESCRIPTION

Covers the PHP scripting language and the MySQL database to create
dynamic Web pages. Topics covered include: PHP scripting fundamentals;
creating, accessing, and manipulating data with the MySQL database
within a PHP program; creating HTML forms; and writing secure PHP programs.

COURSE OBJECTIVES

Upon completion of this course, the student will be able to:

I. Introduction to PHP
II. Using variables
III. HTML forms and PHP
IV. Using numbers
V. Using Strings
VI. Control Structures
VII. Using Arrays
VIII. Creating Functions
IX. Files and Directories
X. Using the MySQL Database

STUDENT RESOURCES

Required Text:
PREREQUISITES

There is an inferred prerequisite for this class... You must know some native HTML.

Please read the link below to determine whether or not you have the proper HTML background to take this class:

HTML Prerequisites

PARTICIPATION

To be considered actively participating in this class, you MUST logon at least once per week. A week is measured from Monday-Sunday. Any student who misses two full weeks may be dropped from the class.

It is still your responsibility to make sure that you withdraw from the class, by the official withdraw date found on the college Web site, if you want to receive a grade of "W".

COMMUNICATION

Sending your questions via email is the way to get help with the course material. At a minimum, I check and respond to my emails at least once per day Monday-Friday.

Please do not hesitate to contact me via email. Some students feel they are imposing on me by asking questions, but that's what I'm here for!

In fact, it's better to email more frequently with a question or two, than to wait until you are completely frustrated and send me 10 questions in a panic.

I teach from 5-8 different classes a semester at multiple colleges. When you send me an email, please identify yourself by first and last name, what class you are in, and what college you are attending.
Most times I will wait for you to contact me with a question before you hear from me directly. However, if I notice you are getting behind in the class, or struggling in some way, I may contact you first to see if there is anything I can do to help.

On rare occasions, I may suggest a phone call appointment if we both agree it is warranted.

Periodically I may send out an email to the entire class or send you an individual email. I MUST have a valid email id from you at all times to insure proper communication. If you change your email id, notify me via email.

If I send you an email and it "bounces" (gets returned as undeliverable) I will temporarily remove your access to the class web site until I establish email communication with you again.

If you are emailing me about a question/problem with something you're working on, be sure to send me your entire program (or SQL statement) as an attachment to the email. It's very hard to help you debug your code just by your description alone.

Emails can (and do) get lost sometimes. I usually respond within 24 hours (weekends and holidays excepted) so if you do not get a reply from me within 48 hours - email again! At that point, feel free to send a copy of the email to my emergency email listed above.

NOTE: I have a SPAM filter. It works pretty good. Sometimes too good. Sometimes it will filter out a legitimate email from you! If you suspect this is happening (or just want to prevent it from happening), then put the following somewhere in the Subject line of your email (all caps): ITSSAFE

Be as specific as you can be with your questions. Please don't send an email that says something like, "I'm having some trouble with this assignment, can you look at it and tell me what I'm doing wrong?" Try to get some of the code to work and then ask me about the part you are struggling with. I expect students to make an attempt at solving at least part of the problem before I will help them.

Sometimes when I answer your questions with factual information, the tone of my reply can sound cold and unfeeling. That's not my intention! I answer a lot of email, and if I'm really busy, I may give you "just the facts".
Please don't interpret that as a form of abruptness that means I want you to go away! :-D Ask as many questions as you need to :-) 

WRITTEN COMMUNICATION POLICY

In the past I've gotten a number of emails that are very difficult for me to read because they are so poorly constructed.

Back in high school, I was never a particularly good English student. Now, many years later, I'm beginning to understand why it was important to learn what they were trying to teach me! :-D

This written communication policy doesn't demand perfection or formal writing. It does have some (very basic) rules you must follow or I won't read your message!

Rules:

1. Write in complete paragraphs! Cover only one topic per paragraph!

It's very hard to read emails when the ideas are all run together. Don't give me a "stream of consciousness". Here's a particularly bad example (somewhat modified to protect the guilty :-D )

"Thanks for checking the status of my assignment. I probably should have got back to you sooner about finishing the class but thought you knew I was going to the class website and was trying to figure it out without bothering you. Which brings me back to my original problem. I was trying too over wright one of the program I down loaded (Assignment 2) and it did not over wright. I know this because had a test for the category on the original program and left it off the new program but it kept testing. How does one over wright on the class website? The reason for this was my check for the select code always came up false even with the correct code entered. I was trying to figure out if my program was able to reach the interface or the functions. There is no error about not finding the functions or an error at all. The printout on the web page is perfect but the results are screwy."
It would take me a long time to parse an email like this and respond to the individual points. I won’t do it!

2. Use proper capitalization and spelling. Do not write in "text messaging!"

Example:

hey steve how r u? im doin gr8 w/asgn 6 wut is the next 1?

I don't expect perfect spelling in your emails (Lord knows, I'd be a big hypocrite to expect that!), but your emails shouldn't be riddled with spelling errors either.

Give me a fighting chance! I'm trying to understand what you are saying ;-)

3. Give me the relevant history to support what you are asking about.

I have about 200-250 students during a semester and if you refer to a previous post by you (or another student) I'm not likely to remember it. Just copy it into the email you are sending me (or leave in the Email history when replying.)

Read your emails after writing them to make sure they make sense. Be sure that you are giving me enough information so I can help you effectively.

4. Try not to go on long tangents.

I answer a lot of email, so if yours is the 20th one I'm answering this hour, I tend to just look for the relevant details so I can answer your questions.

I don't want to take all the humanity out of our communications, just know that I probably don't have the time to read anything too involved about your personal life ;-)

Your emails should still be informal and conversational. It's ok to make jokes and be funny! We all need a little humor to make it through the day! :-)

5. Remember to give me your full name and the class you are in.

Consequences:
If I find that your email (or communication) is in violation of these rules, I will answer them with "Please rework this email following the written communication guidelines discussed in the Syllabus".

I'm not planning to be "anal" about this! :-D I just want you to make it easier for me to help you :-) 

REQUIRED READING

You are required to read all of the text pages that I have posted in each Learning Object, such as the Instructor Comments and the Assignments.
You need to read any of the required reading specified for your books.

I may post information on these pages that is not found in the book, but may be required for you to complete some of the assignments. In fact, I often do a fair amount of "teaching" in the assignment descriptions themselves. The assignments are where I tend to offer information based on my many years of experience working in Information Technology.

NOTE: The comments I provide for each chapter are not intended to replicate everything you can read in the books. The book is the primary "dispenser of information". As an Instructor, I supplement this by answering your questions, providing additional information, helping with class exercises/assignments, trying to steer you away from unnecessary problems, and emphasizing key points I want you to remember.

ASSIGNMENTS

There are a number of assignments that will require you to solve problems using PHP/MySQL programs or answer questions. Most of these assignments are worth 10 points each, but some are worth less.

The Final Exam is a written exam and is worth 10 points.

I require that you read the material shown when you click on the "Best Programming Practices" link found on each Assignment page. A portion of your grade for each assignment will be based on how well you adhere to these practices.

Students may work with their own Web servers if they wish, but their final submitted PHP/MySQL programs MUST compile and work on the practice server I provide. I will test all program assignments on the practice server only!

There is a "Grading Criteria" link in the Assignments section of the
Learning object (if applicable) which describe the criteria I will use to grade your programming assignments. Read it carefully!

Assignments will generally be graded within 7 days of your submission.

The Assignments must be submitted in the order they are due and the class Web site will enforce this by not enabling the link for an Assignment until the previous Assignment has been submitted.

Each assignment has a target due date. If you are going to submit an assignment more than a week past its due date, then you need to contact me beforehand or you may receive zero points for that assignment at my discretion.

If you are having trouble with an assignment and we are working together to get through it, I don't enforce the late penalty. I let you take your time, as long as you are actively working on the assignment.

The bottom line is: communicate with me before the due dates and I'll be lenient about the schedule. It's more important that you learn the material, than meet the schedule.

There are some additional rules...

Remember that NO assignments or exams will be accepted after the FINAL DATE of this class (listed at the top of this page) unless I've given you an explicit extension.

You may work through the class faster than the schedule if you wish, however, you may NOT submit more than one assignment on any given day! This rule has two purposes. One, to keep you from rushing through the material and, two, to keep students from submitting most of their assignments during the last week of class!

DEBUGGING YOUR PROGRAMS

The best way to work on solving the problems is to get a little bit working at a time. Break the problem into steps and solve one step at a time. Then move on to the next step and get it working, and so forth.

If you are completely lost, it is usually because you are trying to get too much of the problem solved at one time.

If you've found you've gotten too far into a program and are confused, try commenting out large areas of the coding until your program does something correctly. Then start uncommenting small portions of code to try and isolate the problem.

Another technique is to display messages to yourself from within the program at intermediate points. Say that a formatted name is not appearing properly on a Web page, for example. You might display the unformatted name first to see that it was retrieved properly.

While I will help you debug your programs, the ultimate responsibility
is yours.

I will usually give you much more specific debugging help in the
beginning of the class, but as time goes on my comments will become
more general and I will expect you to go through the debugging
techniques described above.

After the first few assignments, you find that you are not "getting it"
that is a sign that this class is probably not for you and you should
consider withdrawing with a "W" before the cut-off date.

In some case, you may just need some more experience before attempting
the class again.

CHEATING

Any student found to be cheating on an exam or programming assignment
will receive zero points for that assignment.

It's perfectly acceptable for students to work with and help one
another, however, it is not acceptable for students to hand in the same
solutions.

Do NOT share code! If you want to help another student show them
an example of how you solve a particular problem, don't give them the
exact code or design that you will be submitting for your assignment.

Sometimes students think they can copy assignments and then "just
change a few things around" to disguise the fact. Don't try it. Almost
everyone who does slips up and leaves in some telltale similarity that
I pick up on.

If I detect that code has been shared or copied, ALL students involved
will receive zero points for that assignment, even if you have already
received a grade for that assignment. For example, if Student A submits an
assignment on March 1st and gets 8 points, then Student B submits the
assignment a week later and I notice it is a thinly disguised rework of
Student A’s assignment, I will go back and change Student A’s
grade to 0 points along with Student B’s grade.

If a student is caught cheating more than once, they will be expelled
from the class and receive an automatic "F" for the semester.

GRADING POLICY AND STANDARDS

90% Assignments, Quizzes, Surveys, and other Assessments
10% Final Exam

Points Letter Grade
TIME EXTENSIONS

Generally, I don't give out the official "Incomplete" grade but, if you are unable to complete the class because of unforeseen circumstances, I will give you an extension until Feb. 17, 2006 to complete the remaining assignments and submit a Grade Change card for you at that time to update your grade.

Please note that I usually take the Christmas and New Year's weeks and the first three weeks after the end of the Spring semester as vacation weeks and do not answer email during that time.

THIS OPTION IS NOT AVAILABLE TO ANYONE WHO IS MORE THAN ONE MONTH BEHIND THE TARGET_DATE SCHEDULE!

It is YOUR responsibility to let me know as soon as possible that some emergency has come up that will prevent you from completing the work by the end of the semester.

You do not have to tell me any of the particulars of your situation (e.g. medical problems, death in family, etc.) Simply make the request for the extension if you determine that it is necessary. If you are not more one month behind schedule at the end of the semester, I will grant the request.

I'm willing to help those who really need it, but too many "Extension" requests have come during the last week of the semester from students who have done NONE of the work! They suddenly realize, "Hey, I might not be able to finish in time!" Probably not. ;-)

MILITARY PERSONNEL

If you are in the military and your duties prevent you from completing the work on time (such as a deployment), send me an email from your military email address and I will give you whatever time you need to complete the class.