CS 377: Database Systems

Project #4

Monday, April 25, 2016 at 11:59 pm

SUBMISSION:
The project should be submitted electronically via Blackboard before 11:59 pm. Your submission should:

- Include a README file with the comment that certifies that the work is your own work.

  ```
  /* THIS CODE IS MY OWN WORK.
  IT WAS WRITTEN WITHOUT CONSULTING CODE WRITTEN BY OTHER STUDENTS.
  _Your_Name_Here_ */
  ```

- Specifies in the README file if you are using one of your 2 late assignment exceptions
- Contain one file for each problem which follows the naming convention of camel case for your first name and last name and the problem number (i.e., <FirstnameLname>Prob<Number>.php). For example, the instructor's submission for problem number 1 would be: JoyceHoProb1.php

INSTRUCTIONS:
You have been asked to design a PHP prototype for the Online Bookstore that was introduced in Project #1. A database administrator has already set up a MySQL database according to the specifications from Project #1 and #2 and filled the database with tuples. Your online bookstore needs to connect to the cs377spring16 server and access the bookstore database using the cs377 username and password provided on Piazza (same as IMDB). For this project, you will not be graded on aesthetics but purely on functionality (whether or not it does what the requirements specify). But, if you happen to design a professional looking website, we may use your HTML code as a template for future CS377 PHP projects! Your PHP scripts will be evaluated on a separate server with the same database setup (but the bookstore instance will be different), so make sure you are not hard-coding anything into the webpages.

USEFUL NOTES:

- Consider changing your password (to something secure and not '123456' or 'password') rather than stick with the default assigned password. To change your password, after ssh type the following command to get the prompts to enter your old password and then a new one twice to set your new password:

  ```
  passwd
  ```

- Consider writing a stand-alone PHP program that you can execute from the command line that retrieves all the information necessary. This way you can have more information when the program has bugs (in the webpage version, you will get a blank page with no useful messages).
- You are free to work on your local machine, but you will need to install your own local Apache + PHP server, which may not be as straightforward as the MySQL installation. And you will want to make sure that it still works on the cs377spring16 server because testing will be done on a similar setup.
- To download your code from cs377spring16.mathcs.emory.edu, you should use secure copy to transfer it to your local machine using the following command:

```
scp <userid>@cs377spring16.mathcs.emory.edu:~/public_html/<filename> .
```

The command tells the secure copy program to copy the specified file located in your public_html directory on cs377spring16 server to the current location on your local machine (denoted by the period). UNIX and Unix-like operating systems (e.g., MacOS X) have scp automatically built into their system. For Windows users, you will need some client that supports scp, such as WinSCP or Putty SCP (PSCP) to be able to securely copy from the cs377spring16 server. The command can also be used to transfer files from your local machine onto the server as well. To transfer a particular file into your public_html directory, you can use the following command:

```
scp  <filename> <userid>@cs377spring16.mathcs.emory.edu:~/public_html/
```

WEBPAGES TO DESIGN:
Your PHP prototype will contain 3 pages that will serve different functions.

1. Online Bookstore Catalog (30 points)

   Design a webpage that displays all the books that are available in the bookstore. The book information should contain ISBN, the title, the year, the publisher name, the author(s) of the book, and the price of the book. You can choose how you want to display the information (e.g., table, list, etc.), as long as the page renders properly on a web browser and there is an obvious delineation between books. Basically, it should not be one continuous line of text without any properly format, it should be clear what the book title is, what year it was published, who published it, who wrote it, and how much it cost.

2. Add Book to Shopping Cart (35 points)

   Design a webpage (a form) that allows customers to add one book at a time to a shopping cart. The form should allow the user to input their email, the shopping cart number, and the ISBN of the book they would like to purchase. Your webpage should check the user input to make sure the following:

   - All the input fields have been set (i.e., customer has typed something) and if not print a message to the customer to type something in *that* field
   - Check that the user exists in the database and if not, print a message that the email is incorrect - we are assuming that we already have all the customers in our system and are not able to handle any additional customers
   - Check that the bookstore actually carries the book (ISBN number exists in the database) and if not, print a message saying that the bookstore does not carry that book.

   Once the user input has been sanitized, the webpage should add the book to the user's shopping cart in the database. Note that if the shopping cart number doesn't exist, you will need to create the new cart number before adding the book.

3. View the Shopping Cart (35 points)

   Design a webpage that allows the customer to view the contents of their shopping cart at any time. This page should allow the user to input their email and their shopping cart number and retrieves all the items in their shopping cart. The webpage should check the user input to make sure that the shopping cart they are trying to view exists in the table and print a message that the email or the shopping cart is incorrect. If successful, the shopping cart webpage should contain the number of items in the cart and the total cost of the items in the cart, and for each item in the cart display the ISBN, book title, and the price of each book in the shopping cart. Note that it should also handle the case when there are no items in the cart (cart exists but no items placed yet).