CSCI UA.0060 Spring 2024

Assignment 7 – Connect Website to SQL Database

Deliverables

Upload your PyCharm project to GitHub.

Overall Requirements

In this assignment, you will update the website skeleton provided in your GitHub repository, so that the content on the Category page will be driven from data in a database.

Specific Requirements

1. Decide on the specialty of your bookstore, choose a name and find/create a logo.
2. Create a database and implement the following schema:

book(**bookId**, *categoryId*, title, author, price, image, readNow)

category(**categoryId**, categoryName)

1. Populate the database with at least 4 categories and 16 books (four in each category). You can choose the books and categories, but they should relate to your bookstore’s specialty.
2. Clone the bookstore repository into PyCharm. The project currently contains:
   1. A templates folder containing four HTML pages including:
      1. Index.html – the welcome page
      2. Category.html – the page that displays books for a category
      3. Base.html - used for the header and footer for the other pages
      4. Error.html – that displays some error information
   2. A static folder containing:
      1. A css folder containing one file with rules for both pages
      2. An images folder with subfolders for books, categories and misc. The misc folder has various icons (which you are welcome to change if you wish). The other folders have a dummy image to allow them to be cloned.
   3. An app.py file that contains the overall structure for your Python program
3. You must update the Python program and HTML files to make the site functional. The book and category information must be retrieved from the database. Which books are displayed on the category page is determined by the user’s category selection.
4. Depending upon your interest, knowledge, skill and time, you are welcome to modify the site to look how you want it to. You can change colors, fonts, text and layout. The basic requirements for this assignment are that four different books are displayed on the category page depending upon the user’s selection of at least four different categories and that those books are retrieved from the database that you have created.
5. Screenshots of the two Treble Clef bookstore pages are included in the GitHub repository, so you can use them as a frame of reference. You do not have to exactly replicate these pages.
6. Other specific functionality that must be reflected on your site are:
   1. The header must have a logo for your bookstore
   2. The header must have a dropdown selection that allows the user to select a category of books. This determines the books that appear on the category page.
   3. The Index page must include some welcome text that describes the value proposition of your bookstore
   4. The index page must have clickable text and images for each of the categories. This determines the books that appear on the category page.
   5. The category page must show details and images for at least four books.
   6. The category page must have a “navigation bar” that shows all the categories. The selected category should be styled differently than the other categories. The “non-selected” categories must be clickable links that change the books on the category page.
   7. Only the books relating to the selected category should be retrieved from the database and displayed on the category page.

1. When you have completed the assignment, export the database as you did in assignment 5 and copy the resultant SQL file into the bookstore project folder. Once done, submit the assignment to GitHub.

Grading Rubric

See Brightspace for Grading Rubric