CSE 340 Final Project

Commented [RB1]: Owner password is: 2243S4000W

NOTE: This project and all code generated to solve the problems contained herein are proprietary to Blaine Robertson. Neither this document, nor any code used to solve the problems described, are to be posted to any server or service where they can be accessed by others, other than the normal submission process for grading to occur.

This exam consists of three tasks. Task A is an update of the phpmotors database and placement of a new folder containing images within the phpmotors folder. Task A is required to accomplish the two coding tasks (B and C). As with Enhancements, the phpmotors folder, containing all code from the semester, including the final project, should be zipped, and submitted by the deadline. A video must be created, posted as "Unlisted" to YouTube, and the video URL be submitted as a comment with the zip file. The video should demonstrate the items listed in Objective 6 of the grading matrix.

Task A

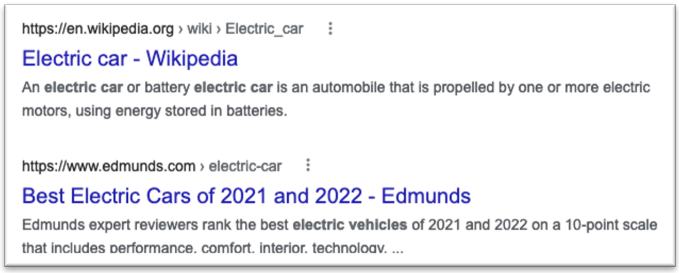
1. Download the final project zip file.
2. Unzip the file.
3. Inside will be a SQL file, used in step 5 of this task, and a folder, used in step 6 of this task.
4. In phpMyAdmin:
    a. Open the phpmotors database.
    b. Click the "Operations" tab.
    c. Find and click "Copy database to" and enter name like "phpmotorsBackup".
    d. Leave all other settings as they are.
    e. Click "Go".
    f. On the left, you should see the "phpmotors" database and the new one.
    g. The copied one is for a backup only.
5. Update the phpmotors database by importing the included sql file. This will replace the inventory and carclassification tables and add an additional images table.
6. Move the vehicles folder to the images folder at the root of the phpmotors site. This folder should replace the existing folder of the same name.
7. Make necessary adjustments to your existing code base so that the phpmotors site continues to work as it did prior to starting the final project.

Task B

Add search capability to the phpmotors site. The search should allow the user to enter partial strings, not be forced to find exact matches. Processing code must be of you own construction, not an implementation of a pre-built search tool.

1. The search tool must be implemented in the page header to be available in each view.
2. Base the search on fields in the inventory table. You may choose the fields to be used, but the chosen fields (at least two) must be made indexes within the table to optimize search capabilities.

3. The search results should display in a search result view and have the same general appearance as search results in Google. (See illustration)



4. You may want to display the vehicle thumbnail in the results as well.
5. Use custom functions to organize and prepare data for use in the view.
6. The view must have the same look and feel of all other website views.
7. Each vehicle displayed in the search results must contain a link to display the detail view as built in enhancement 8. The vehicle data in the inventory table has changed slightly, so update the detail view to accommodate these changes.
8. At the top of the results, a line should indicate the total number of records found through the search, like what Google search does. (See illustration).

Task C

1. Limit the results displayed in the search result view to 10 records at a time.
2. If a search returns more than 10 records, implement pagination.
3. The pagination should appear at the bottom of the view, below the search results, but above the footer. The behavior should be like Google. Specifically:
   a. A "Next" and "Previous" arrow / symbol / text should be displayed, and be a link, to move through the results, in groups of 10. The next arrow should only display if there are more records to display. The previous arrow should only display if there are previous records from the search to display.
   b. A number must display, and be a link, to the appropriate group of 10 results to be displayed, unless the group being viewed is that number. In the illustration below, the second group of results is being displayed, so the "2" is not a link and is a different color to indicate that is the case. (See illustration)

Grading Matrix - Each Objective is worth 25 points

Objective 1 (Minus 5 pts for each error)

a. All views meet the standards of the Frontend Checklist.

Objective 2 (Each task is worth 9 points, up to the maximum of 25 points)

PHP is used to process the search request and deliver results. Specifically,

a. The search process works and meets the requirements listed in Task B.
b. The results of the search request are processed, wrapped in appropriate HTML using a custom function, and sent back to the search results view in maximum groups of 10 records.
c. Pagination has been implemented and meets the requirements listed in Task C.

Objective 3 (Each task is worth 9 points, up to the maximum of 25 points)

a. A controller has been created to handle all search related activities.
b. A model has been created to handle all search related database interactions.
c. A search results view has been created to display results according to Tasks B and C.
d. Custom functions exist to handle non-model based needs.
e. Existing controllers and models may be used, as needed, to display vehicle details.

Objective 4 (Yes or No)

a. A search model handles all necessary interactions with the database successfully, to carry out the search interactions.

Objective 5 (Minus 5 pts for each error or omission)

a. All inputs are validated and sanitized appropriately throughout the application.
b. Errors are handled appropriately as done previously in the class.
c. If no results are found, an appropriate message is displayed in the result view.

Objective 6 (Minus 5 pts for each error or omission)

a. The zip file and video URL are submitted correctly and on time. If NO, no credit will be given for the entire project.
b. The video shows:
    a. The search access tool being present in multiple views.
    b. A search being conducted, and results being returned.
    c. The pagination working with a result set greater than 10 records.
    d. A search being done that returns no records and the appropriate message being displayed.
    e. A search, containing disallowed characters being conducted and error checking dealing with the issue.
    f. The search results view meeting the frontend checklist.