Unit 2: Application of OOP (I): AJAX

Practice 5: Search with Auto Suggestion in the Movie Library Application

This practice continues Lab 3 by adding the auto suggestion feature. Four files need to be modified or created.

- *application/app_view.class.php:* this file needs to be modified to load an external JavaScript file, allow the search box to respond to the **keyup** event, and add a div block below the search box to display auto suggestion items.
- config.php: The constant **BASE_URL** defined in the file needs to be modified to match the path of the current application.
- *www/js/ajax_autosuggestion.js:* this file handles client-side scripting including the AJAX requests.
- *search_movie.php:* this file handles server-side tasks including searching the database for movies and sending all matched movies in JSON objects.

Get Ready

1. Download data file *autosuggestion.zip* from Canvas and extract the **autosuggestion** folder into the **htdocs/I211/Unit2** folder.
2. Review the code in PhpStorm and preview the application in a browser.

Part 1: Modify the *app_view.class.php* file

3. Open *application/app_view.class.php* file in PhpStorm. Scroll down to the end of the file. Right above the closing </body> tag around Line 63, insert the following line of code to load an external file called *ajax_autosuggest.js*. Please note the *ajax_autosuggestion.js* file is located in the **www/js** subfoder.

```
<script src="www/js/ajax_autosuggestion.js"></script>
```

5. Right below the form and still inside the div element, create a div block and set its id to **suggestionDiv**.

```
<div id="suggestionDiv"></div>
```

Part 2: Modify the BASE_URL in *config.php* file

6. Open *application/config.php* file in PhpStorm. Make sure the value of **BASE_URL** is set to http://localhost/I211/Unit2/autosuggestion.

Part 3: Modify the *ajax_autosuggestion.js* file to enable AJAX

6. Open *www/js/ajax_autosuggestion.js* in PhpStorm.
7. Review the existing code so you understand the logic. You only need to complete the **suggest** function.
8. Complete the **suggest** function. This function sends an asynchronous request using "GET" method to *search_movie.php* file. A query string variable called **q** is sent along with the request.

```
function suggest(query) {
    //if the search term is empty, clear the suggestion box.
    if (query === "") {
        suggestionBoxObj.innerHTML = "";
        return;
    }


    //create an XHR object
    let xhr = new XMLHttpRequest();

    //open an asynchronous AJAX request
    xhr.open("GET", "search_movie.php?q=" + query, true);

    //handle server's responses
    xhr.onload = function () {
        //retrieve server's response and parse it to a json object
        var titles = JSON.parse(xhr.responseText);

        // display suggested titles in a div block
        displayTitles(titles);
    }

    //send the AJAX request
    xhr.send(null);
 }
```

Part 4: Modify the *search_movie.php* file

9. Open *search_movie.php* in PhpStorm. The file currently contains code that searches movies. Now you will add code that automatically suggests movies as the user types search terms in the search box.
10. Scroll down to the bottom of the file. At the very end, create a new **if** statement:

```
//auto suggest movies
if (filter_has_var(INPUT_GET, "q")) { //process ajax request

}
```

11. Inside the **if** statement, type the following line of code to retrieve the search terms from the query string variable named **q**:

```
$query_terms = filter_input(INPUT_GET, 'q', FILTER_SANITIZE_STRING);
```

12. Now, search the database for movies and store the matched movies in a variable called **movies**.

```
$movies = $movie_manager->search_movie($query_terms);
```

13. Retrieve all movie titles and store them in an array.

```
$titles = array();
if ($movies) {
     foreach ($movies as $movie) {
          $titles[] = $movie->getTitle();
     }
}
```

14. At last, encode and echo the string representation of a JSON object.

```
echo json_encode($titles);
```

15. Save the file. Now the application is complete. Run the *index.php* file. Type some search terms in the search box. If everything works well, you should see a list of suggested movie titles as you type. You may use the **up** or **down** arrow key to navigate the list. Press the **Enter** key to select an item in the list to be searched.