TOP Movies

The main goal of this application is to store the information and punctuation of the movies you have watched. For each film, you have to save the following information: name of the film, ISAN number, year and your punctuation. There can’t be two movies with the same ISAN number, but the name can be repeated.

The page will be divided in two parts:

* At the top of the page we have a list with the movies’ information and the punctuation.
* At the bottom of the page you must show a form with: a textbox for the movie name, another textbox for the ISAN, another textbox for the movie year, and a combo box for the punctuation (0 to 5). Also, a button should send the form.

Each time the form is submitted:

1. If the name and the ISAN are empty, you have to show a warning message.
2. If the introduced ISAN is not in the list and the ISAN is 8 digits long a new register is stored if all the fields have values. Otherwise, you have to show a warning message.
3. If the ISAN is empty (and the name is not empty), we have to show a list of movies that contain that name. Letter case and accentuation marks don’t have to be taken into account: if the user types *supermán* the list must show:
   1. “Superman” from 1948.
   2. “Superman” from 1978.
   3. “Batman vs. Superman” from 2016.
   4. “Superman & Lois” from 2021.

We assume that we have all those movies in our list already.

1. If the ISAN number already exists in the list and all other fields are not empty, the name and the punctuation of the movie have to be updated with the information introduced in the form.
2. If the user gives an existing ISAN number and leaves the movie name empty, the movie register has to be deleted from the list.

Besides, the form has to keep the data introduced in the form fields in the previous request.

Once this part is finished, you will have to add an entry page (index.php) with a form where the user can enter his/her name. When clicking on the *send* button, it must redirect to the main page (main.php). In the header of the main page, you must show the title “*username* movies:”.

NOTE: the information is displayed on the screen, it is not permanently stored in a database or file.

# Improvements

Once the minimum requirements have been fulfilled, it is possible to expand the functionality in the following aspects:

* Improve the application interface.
* Store the entered data permanently (xml file, json, database...)

# Evaluation

| Criteria | Punctuation |
| --- | --- |
| Generate PHP page structure: input form. | 0.5 |
| Show existing movies in the list. | 0.5 |
| Enter the list data inside a hidden field in the form.  Use an associative array. | 1 |
| Check the data sent by the form, showing a warning when necessary (case 1). | 0.5 |
| Enter the data of a new movie in the list (case 2). | 1.75 |
| Search functionality (case 3). | 1.5 |
| Modify the name of an existing movie (case 4). | 1 |
| Delete a movie from the list (case 5). | 0.75 |
| The entered data remains in the text boxes after the submit button is clicked. | 0.5 |
| Add index.php entry page to customize the list. | 0.5 |
| Avoid code injection in the input data  (i.e. <body onload="alert('Hello')">) | 0.5 |
| Code commented. | 0.5 |
| Make commits when new features are added. | 0.5 |