# Lab 5: Movie Database

**Overview**: Based on the lessons in PHP basics, validation, and creating a simple MySQL CRUDS application, you will create a simple database of movies. These could be movies you own, have watched or want to watch.

As opposed to previous labs where the instructor has gone through much of the functionality in class, this lab will require you to create everything on your own.

You will have several hours of lab time during scheduled class to work on this and have instructor assistance if required. You are still required to create and debug this on your own however. Upon request, the instructor can show you a demo of this (in-class only).

## Step 1:

Plan your project by sketching out what information you will store for each movie. This will determine what your database table will be and what form inputs you create for the interface.

## Step 2:

Create a table with all necessary columns with the correct datatypes. Most will be VARCHAR(100), a description column may be TEXT, and your Primary Key will be INT(10) and MUST be Auto-Incremented.

## Step 3:

Create a page called insert.php that will display a form to the user for inputting a new movie.

* Required info includes the following:
  + Movie name
  + Production company's website
  + Runtime
  + Description
* Each of those MUST be validated (not empty and correct format).
* You decide which other info is required and which is optional: Please have **at least 7 columns** of useful data. Basic validation is required on those optional fields that are filled out to not cause a problem with the database.
* If validation is passed, then all info for this new movie is entered into the database.

## Step 4:

Create a page called index.php that outputs all movie names and creates links from each to a movie.php page (and sends a query string to it).

## Step 5:

Create a page called movie.php that displays all the info for the particular movie.

* All info is nicely styled and certain info is only outputted if it exists.
* If have email addresses, they are linked using a mailto: so that you can easily click the link to send an email.
* Website URLs are linked so that you may click the link to visit the website (target a new page/tab so that you don’t leave your application please).

## Step 6:

Once you have functional input/output working and tested, please create any other pages for update, delete, and search.

## Marking: 10 marks total.

*Note: This project MUST be populated with real data for* ***at least 10 movies****. The instructor will NOT accept it unless it contains real data.*

1. Create, Update, Delete, and Search functionality: [4 marks]
   * 5 Necessary fields from step 3
   * From 7 others you need to add
     1. a checkbox or radio
     2. a select
     3. your choice for other 5
2. Functionality and formatting of your Index and Movie pages: [2 marks]

* Index shows movie names (alphabetical order) and links to a movie page.
* Movie page shows all data with labels.
* Email addresses and URLs are linked.

1. Overall design & usability: [2 mark]

* You may use our template, or design your own.
* Validation of all necessary fields. Email, URL (try FILTER\_VALIDATE\_URL) must be valid formats.
* All pages must be linked.
* All content is well-aligned, padded, and readable.
* All form elements well styled.
* User knows exactly what do for a adding a new movie (required fields marked; validation messages) and using all aspects of the application.

1. All admin pages (Create, Update, Delete) are secured by a single-user, sessions-based login and can easily logout. [2 mark]

Hints:

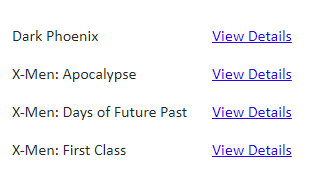
* Please make sure that you have your basic MySQL input/output working before you move on.
* Beware of Copy/Paste/FAIL if you simply move files over from your last database project.

## Submission

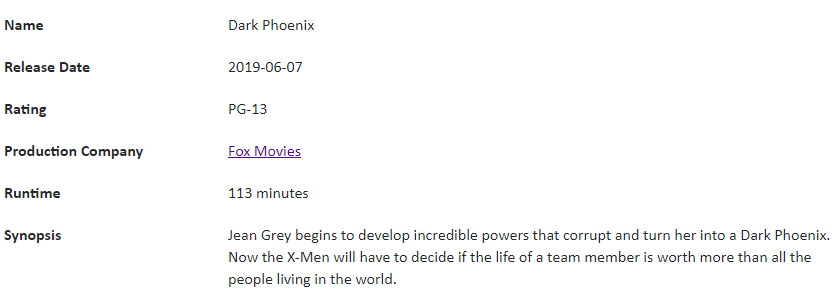
After successfully demonstrating it to your instructor, zip your folder/file(s) and submit it to your dropbox as Lab5.zip

# Screen Captures

### index.php



### movie.php



### insert.php



### Search.php

