

Web Development II (CSC10217)

Session 2, 2017

Assignment 2	Functional PHP to MySQL Website/Database.
Unit Objectives	1 - 4
Due Date	Week 12 (Midnight, Friday 29 th , September 2017)
Weight	45% of overall unit assessment

1. Task Description

Our hypothetical DVD rental shop has now decided to extend their business by creating an on-line booking site. We will be developing several modules for the shop, as the overall system is quite complex. Your assignment requires you to write a set of fully functional PHP Web pages suitable for the shop. Rather than give you a set way to build the pages, you can choose an appropriate design of your own to fit in with your original design.

When you have completed the assignment, you are required to;

1. upload your site to your student directory (on the school's server infotech.scu.edu.au)
2. upload a zip file of your site (and the design documentation) to the MySCU assignments folder. Be sure to include as a comment on the upload the fully qualified URL to the index.php page on the infotech server.

Please note that all material used on your website has to comply with Southern Cross University's "Web Hosting and Publishing Policy" located at:

http://study.scu.edu.au/casc/policy.php?policy_id=55

PART A: DESIGN

(i) Now that you are more familiar with PHP and MySQL and having received feedback on the design from your tutor (from assignment 1), you should update your design documents before starting on the coding for this assignment. Remember as well to update your testing procedure as additional testing for the database will be required. If you have not used the Database provided for the DVD shop, you will need to explain why and provide the design of the database you have used. Remember you **must** use the electronic Assignment Cover Sheet (located in the assignments submission folder of MySCU) as the title page of your design document.

(ii) The database provided for your assignment 2 holds the following information.

Member information, this table will store the information gathered on your assignment 1 JavaScript form or the provided PHP self-targeting form:

- a. Surname.
- b. Other names.
- c. Preferred contact method (land-line telephone, email or mobile)
- d. Day-time telephone number (must contain format (XX) XXXXXXXXX where X is a digit only). **Note:** this may be empty.
- e. Mobile telephone number (must contain format XXXX XXX XXX where X is a digit only) **Note:** this may be empty.

- f. E-mail address (Fully verified using supplied function validEmail – see Notes for the join.php page). **Note:** this maybe empty.
- g. Whether the user wants to receive the e-shops magazine.
- h. Postal address: 3 separate fields (Number / Street – Suburb / City and Postcode). **Note:** this maybe empty if the user chose not to receive the magazine.
- i. Username selected by user (6 characters min - 10 characters max.). **Note:** this should be unique!
- j. Password selected by user (10 characters max. containing at minimum one uppercase letter, one lowercase letter one number and one special character with no whitespace allowed).
- k. Occupation.
- l. Join date.

Stock (Movie) information these tables store the following information.

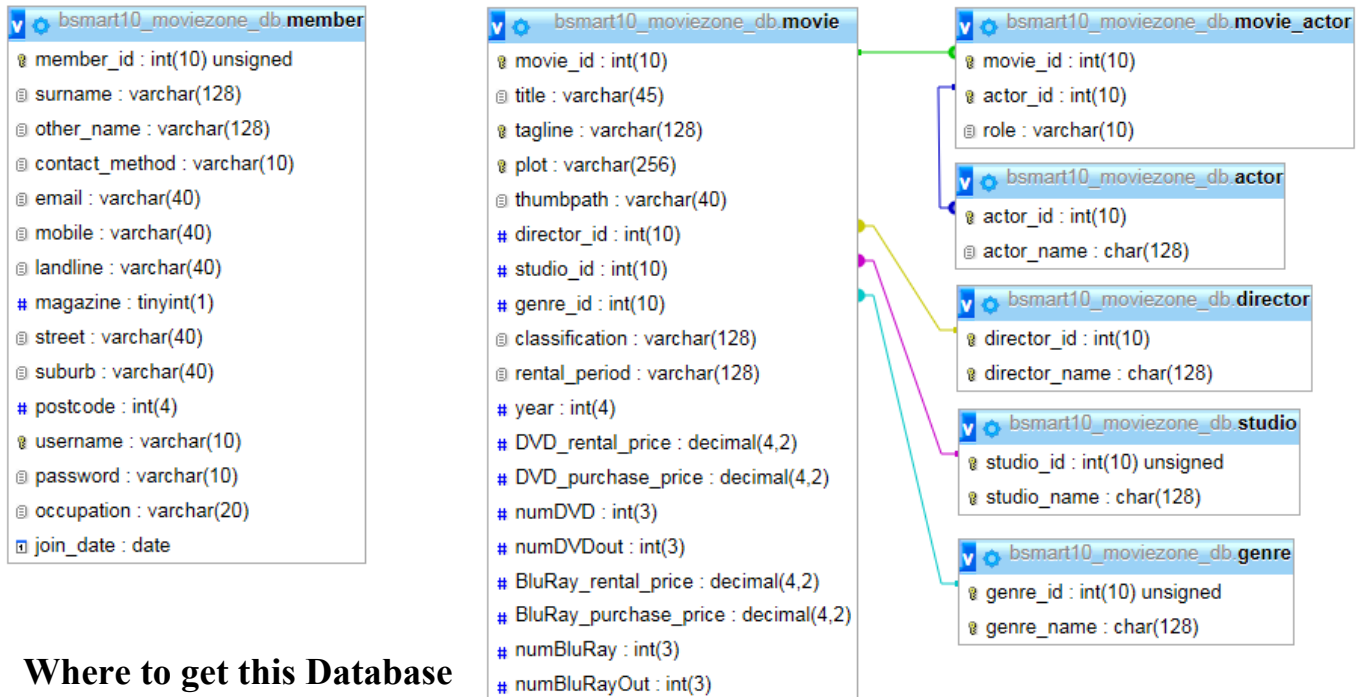
- a. Title (Not unique – how many King Kong movies are there?)
- b. Tag Line (unique)
- c. Plot (unique - up to 256 characters)
- d. Image name (appended with php timestamp)
- e. Director
- f. Studio
- g. Genre (only 1 as that is how done in a shop)
- h. Classification (G, PG, M, MA, R or X)
- i. Stars (3 max – at **least 2** required)
- j. Co-Stars (3 max – or none)
- k. Year of theatrical release
- l. Rental period (Overnight, 3 Day or Weekly)
- m. DVD rental price (individual for each Movie)
- n. BlueRay rental price (individual for each Movie)
- o. DVD purchase price (individual for each Movie)
- p. BlueRay purchase price (individual for each Movie)
- q. Total number of DVDs
- r. Number of DVDs on loan
- s. Total number of BlueRays
- t. Number of BlueRays on loan.

Database notes from Vinh:

You may use your own database design instead of the provided database (although this is a lot more work). However, if you do:

- I suggest all tables have an auto-increment id number (this will help you keep them unique).
- Your tables should be normalised, that is no redundant data should be contained within any table. For example – **many** movies can have **one** director; **one** actor can star in **many** movies etc.
- I suggest the database be loaded with 50 (with a minimum of 9 new release movies) movies for proper testing to be accomplished.
- Stars can be Co-Stars (and vice-versa) therefore it is probably best if they are combined in a single table
- Check out UNIQUE KEY in MySQL – this will cause an error that can be trapped if an attempt is made to add the same data to a column in a table.

moviezone_db - Database Design (thanks to Carolyn Seaton for the database):



Where to get this Database

The moviezone database complete with the entries for 75 movies is provided for you to install in your user account (both MySQL and the images for your `public_html` directory). Available from the MySCU website in Assignments -> Assistance for completing your assignment 2 and in Topic 07 – MySQL Database is a video (.mp4) demonstrating installation of the database using phpMyAdmin and all files required to install the database. The installation of the database is also one of the activities in Topic 07. All images associated with the database are also provided (borrowed for educational purposes from IMDb).

Sample query usage and notes on Views

bsmart10_moviezone_db.movie_actor_view
actor_name : char(128)
movie_id : int(10)

Two views have been created to simplify the use of the database. Using views with the SELECT statement is exactly like using tables.

1. *movie_detail_view* - display list of movies with all details, sorted by movie title:

```
SELECT * FROM movie_detail_view ;
```

```
SELECT * FROM movie_detail_view WHERE genre = 'Sci-fi';
SELECT * FROM movie_detail_view WHERE title = 'Furious Seven';
SELECT * FROM movie_detail_view WHERE classification = 'G';
```

2. *movie_actor_view* -display movie_id's that star or co-star a particular actor:

```
SELECT movie_id FROM movie_actor_view WHERE actor_name = "Vin Diesel";
```

3. Combining the above to get the details of movies with a particular actor:

```
SELECT * FROM movie_detail_view
WHERE movie_id IN (
SELECT movie_id FROM movie_actor_view WHERE
actor_name = 'Vin Diesel');
```

bsmart10_moviezone_db.movie_detail_view
movie_id : int(10)
@ title : varchar(45)
@ tagline : varchar(128)
@ plot : varchar(256)
@ thumbpath : varchar(40)
@ star1 : longtext
@ star2 : longtext
@ star3 : longtext
@ costar1 : longtext
@ costar2 : longtext
@ costar3 : longtext
@ director : char(128)
@ studio : char(128)
@ genre : char(128)
@ classification : varchar(128)
@ rental_period : varchar(128)
year : int(4)
DVD_rental_price : decimal(4,2)
DVD_purchase_price : decimal(4,2)
numDVD : int(3)
numDVDout : int(3)
BluRay_rental_price : decimal(4,2)
BluRay_purchase_price : decimal(4,2)
numBluRay : int(3)
numBluRayOut : int(3)

Where to get additional data or complete data if you design your own database.

To find new movies to add to your database (or to load it if you created your own), I suggest you visit the Internet Movie Database (IMDb - <http://www.imdb.com>) for your movie needs – Additionally as this is an educational exercise (a hypothetical DVD database – that will be deleted at the end of semester) DVD pictures can be “borrowed” from this site for the purposes of this assignment. As homage to IMDb (the site you “borrowed” images from) **include a thank you on your homepage (index.php) linking to the IMDb**

A model DVD shop has been set-up as an on-line example of what you need to achieve. The site is located at: <http://infotech.scu.edu.au/emporium> this site contains all of the data provided to you in the moviezone_db.

PART B: RESEARCH

Add a second article to your TechZone page. The article will be titled “*What CMS Does Your Business Need*”. This will be a **fully cited, using Harvard referencing style**, essay researched and authored by you, of approximately 1200 words where more is acceptable – less is not – articles of more than 1600 words or less than 1200 will be penalised. The article will describe why you (the expert) would choose a particular Content Management System (CMS) for a small to medium business. It should:

- a. Compare and contrast several CMS systems (including costing).
- b. Contain a bibliography & a **word count** (not counting the bibliography).

Notes: You may use diagrams, pictures, graphs as desired to support your article. The bulk of the article is to be authored by **YOU**, an essay that is mostly quotes from other sources will be penalised. Plagiarism will **NOT** be tolerated. See Section 6 of this assignment for further information.

Where should you get information on CMS? A good place to start would be Wikipedia (http://en.wikipedia.org/wiki/Content_management_system), however, do not quote/reference Wikipedia as it is not an authoritative source. However, Wikipedia does contain a great list of CMS systems (open source and proprietary) available.

PART C: PHP

We will be developing several of the **modules** for your hypothetical DVD shop. These are listed below:

Homepage Modification (index.php)

Your original index.html page will become a PHP page. In assignment 1 it was foreshadowed that your Homepage would advertise a random new release movie from the database. Complete this after the MovieZone page – as the formatting and SQL are similar to that needed for the MovieZone.

Notes from Vinh:

- (1) Once you figure out the MovieZone page this should be real easy.
- (2) To randomise your choice of movie to display you will have to know how many movies match your criteria (new release by say number of rows returned in the SQL query) and place that as the maximum number in the following PHP function.

```
int rand ( int $min , int $max )
```

moviezone.php

This is a display page for all movies in the database. When a user first enters the page they will be presented with a view of all the new release movies in the database (ordered by title).

You need to provide the user different views of the movies in your database. Provide the following views:

1. Show all movies,
2. New Releases,
3. Search by Actor,
4. Search by Genre,
5. Search by Director and
6. Search by Classification.

Also provide a links to your admin.php and booking.php on this page. In the case of dropdowns for searching actor and director the list should be ordered ascending. Additionally, the result of any query should display the movies in alpha order ascending. This is exactly how the emporium model is designed, you can view it at: <http://infotech.scu.edu.au/emporium>

Notes from Vinh:

(1) Don't forget that copies of movies (on Shelf) available for booking (both Blue Ray and DVD see example) is a result of total copies subtracting the number out on loan.

(2) Best place to start is list all movies – this will help you figure out your formatting (again – see example site <http://infotech.scu.edu.au/emporium>), then it is a simple matter of figuring out the SQL required to give you the different views.

admin.php

Stock and Membership control. The link to this page should be a menu item on the MovieZone.php page. This page will:

- 1) Require the staff member of your hypothetical DVD shop to enter their name (any name) and password. The password will be “webdev2”. For security purposes the admin page will use a session (I stored an authorised Boolean value – if the password matched the hardcoded value “webdev2”). Once the user is verified (by the password) they should be presented with a menu to allow them to complete the following tasks.
- 2) **Enter a Movie.** This is a more complex task than it sounds as:
 - a. You will need to have the ability to add additional data to the other tables containing the normalised movie data (for example directors).
 - b. All normalised data should be available in dropdowns. My suggestion is Stars, Co-Stars, Director, Studio, Genre, Classification and Rental Period should all use dropdowns (as well as text boxes to address the above – perhaps?).
 - c. Error checking is required, has all the data been entered properly as per your database? You may use PHP (see note on sessions below) or JavaScript for error checking.
 - d. You will need to error check that the move being entered doesn't already exist in the database (perhaps by tagline/plot the fact that these are unique will crash the program unless you deal with the problem!).
 - e. An upload of a picture is required for display purposes. Note: the computer e-shop examined in topic 9 contains this functionality (the code is also given to you – although you will have to figure out how to append the timestamp).
- 3) **Delete a Movie.** Remove it completely from the database and delete the uploaded picture. **Note 1** - deletion of an item in the Computer e-shop also accomplishes this task. **Note 2** - The deletion of a movie will not remove data from the director, actor or studio tables (but should remove the information from the movie and movie_actor tables).
- 4) **Edit a Movie.** Note only certain things can change, i.e. rental period, rental price, purchase price and number in stock. Many things will not change, for example Director, Stars, Co-Stars etc.
- 5) **Edit / Delete a Member.** You will need to list all members in the system (from the join.php) and provide a form allowing the editing of member data. The most common request by users as you can imagine is retrieving/changing the password/username of a member – as these are commonly forgotten. Additionally, you will need to provide a method of deleting the selected member.
- 6) **Exit the system.** This should destroy the session and return the user to the login page.

Notes from Vinh:

(1) When using a session in PHP there are major problems with using the browsers back button –the session forces a refresh on the server rather than using the client browsers previously cached page. This can be

rectified by using the `session_cache_limiter('private_no_expire');` function. This function needs to be declared prior to the `session_start()`; This can be accomplished for your system by checking for a `$_POST` or `$_GET`.....

- (2) To assist you in developing your admin system, I am allowing you full access to the system I developed as an answer for this assignment. It may not be perfect and if you find an error in it let me know (I will attempt to fix it). I expect that with many students utilising this resource that errors will be introduced into the database (again let me know and I will reload the database). The DVD Emporium shop is located at: <http://infotech.scu.edu.au/emporium>

booking.php

Once a user is added to the system (via the join form/php) they will be able to login to and book up to 5 movies for rental/purchase. Similar to the admin.php, the login, logout/checkout should be available on the as a menu item on the MovieZone.php page. This module of the website will NOT be completed at this time (no database functionality) but should have the same capabilities as demonstrated on the video "Booking Movies" located in the Assignments -> Assistance for completing your assignment 2 folder of the the MySCU site.

Notes from Vinh:

- (1) This will also use a session to accomplish usability – remember if all DVD's are out only BluRays can be rented/purchased and vice-versa, if none (DVD and BluRay) are available can't be booked or purchased at all (I achieved this functionality via the booking buttons).
- (2) Only 1 of each title may be booked/purchased
- (3) No actual changes are made to the database at this stage – no changes to availability to DVD or BluRays. – this will be the next module developed for the system (in the future). When the user logs out all data is lost.....
- (4) To assist you in developing your booking system, I am allowing you full access to the system I developed as an answer for this assignment. It may not be perfect and if you find an error in it let me know (I will attempt to fix it). The DVD Emporium shop is located at: <http://infotech.scu.edu.au/emporium> simply either join the system via the join.php page or use one of the existing logins (as per the database you have been given).

join.php

Rather than have you recode this page in PHP I will be giving it to all students as part of your assignment 2. However, as it is so similar to the requirements (and design) of assignment 1 – I will not be providing it to students until all assignments 1 have been received. Again full user access to the model answer I developed for assignment 2 is available to you at: <http://infotech.scu.edu.au/emporium>

2. Marking criteria

A spreadsheet that will be used for the marking of your site is provided (co-located with this assignment specification) on MySCU to itemise exactly what tutors will be looking at in relation to marking your website. It contains a detailed breakdown of the marking criteria for this assignment.

I strongly suggest you peruse this spreadsheet.

3. Submission Format

When you have completed the assignment, you are required to submit your assignment (a zip file containing your complete site and the design document) to the MySCU assignment folder.

The zip file will be named using the following convention:

filename = **FirstInitialYourLastName_CSC10217_Ass2.zip**
(i.e. FJones_CSC10217_Ass2.zip)

Additionally, you are required to upload your site to your student directory (on the schools server infotech.scu.edu.au) and verify that all works well.

Note to **IBS** students: Please check with your local lecturer/tutor the submission requirements for your assignments. Typically, they will be different from on-shore students of Southern Cross University.

4. Advice

You should start this assignment as soon as possible – **Warning!** – don't leave the whole assignment to the last week. You can make a start on in **week 8** (once you understand the MySQL – even earlier for the TechZone article), this will give you the maximum time (just over a month) to complete it.

5. More information

Your updated design documents must be contained in **one single word document**. Drawings and diagrams should be word-processed or scanned, then placed into a word document. Remember you **must** use the electronic Assignment Cover Sheet (located in the assignments submission folder of MySCU) as the title page of your design document.

Files created in non-university standard programs cannot be accepted, as the marker may not be able to view them.

If you need more information about completing this assignment, then you should contact your tutor.

6. Original work

Your assignment must be your original work. Assignments form a major part of course work. Exchange of ideas can be considered educationally valuable; however, excessive collaboration will be regarded as plagiarism, which is a University offence. For example, the copying of significant parts of a document, even if subsequently modified, is plagiarism. Such academic dishonesty will be penalised in accordance with the University's rules and regulations. A simple way around this is to site other material sources (this action will avoid plagiarism but will reduce the level of mark that you receive – compared to your own original work).

You must not copy material from books, magazines, internet sources or other students' assignments. Of course, you may include direct quotes from any source, but these **must be small (e.g. one sentence or one paragraph) and must be properly cited and referenced**, using the **Harvard Referencing Style**.

The assessment process may require some students to attend an interview in order to explain aspects of their assignment.

Plagiarism is strictly prohibited. If plagiarism is found, students may receive zero marks on that piece of assessment item.

Southern Cross University policies for both plagiarism and referencing are located in the General Info Section of the MySCU site for this unit.

7. Retain duplicate copy

Before submitting the assignment, you are advised to retain electronic copies of original work. In the event of any uncertainty regarding the submission of assessment items, you may be requested to reproduce a final copy.

8. School Extension Policy

Students wanting an extension must make a request at least 24 hours before the assessment item is due and the request must be received in writing by the unit assessor or designated academic.

Extensions within 24 hours of submission or following the submission deadline will not be granted (unless supported by a doctor's certificate or where there are exceptional circumstances – this will be at unit assessor's discretion and will be considered on a case by case basis). Extensions will be for a maximum of 48 hours (longer extensions supported by a doctor's certificate or exceptional circumstances to be considered on a case by case basis).

A penalty of 5% of the total available grade will accrue for each 24-hour period that an assessment item is submitted late. Therefore, an assessment item worth 20 marks will have 1 marks deducted for every 24-hour period and at the end of 20 days will receive 0 marks.

Students who fail to submit following the guidelines in this Unit Information Guide will be deemed to have not submitted the assessment item and the above penalty will be applied until the specified submission guidelines are followed.

9. Marks and Feedback

All assessment materials submitted during the semester will normally be marked and returned within two weeks of the required date of submission (provided that the assessment materials have been submitted by the due date). Any delay in submission of this assignment (extension/late) could result in the marks for this assignment being unavailable prior to the exam.

End.