



Assignment 4 (NF, minor subject)

Due: Friday 25.11.2016; 14:00h (1 Week)

Goals

After doing these exercises,

- You know how data can be imported to a database
- You can utilize PHP to store data in a MySQL database
- You will have repeated how to handle form data through PHP
- You will know a few more CSS styling tricks.

Task 1: Film List

Difficulty: Easy

Download the [material](#) for this assignment. The material includes a small database table that contains information about films (film.sql). Your task is to import the data into your database and make the file fim-list.php work.

Option 1: phpMyAdmin

a) Click the “home” icon



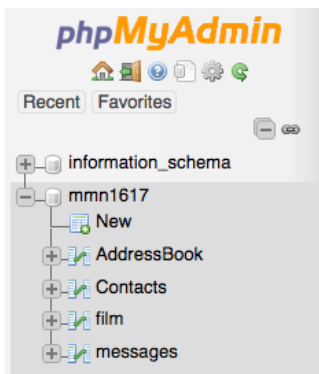
b) Click “import” on the right



c) Click “Choose file” and select the film.sql file

Browse your computer: film.sql

d) Click “Go” at the bottom. You should now see the film table in your overview:



Option 2: CIP Pool / Command Line

a) Open a terminal and type

```
mysql -u <your_user_name> -h db2.cip.ifi.lmu.de -p
```

b) Type: USE <your_database_name>;

c) Type: SOURCE /your/path/to/the/material/film.sql;

d) Type: DESCRIBE film;



Now you have the data and can access it via PHP.

- a) Open the file `film-list.php`
- b) Update the `$connectionInfo`, so that it contains your connection info. If you use XAMPP, all you have to update is the database name.
- c) Open the web-page in a browser. You should see something like this:

ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who must Battle a Teacher in The Canadian Rockies	2006
ACE GOLDFINGER	A Astounding Epistle of a Database Administrator And a Explorer who must Find a Car in Ancient China	2006
ADAPTATION HOLES	A Astounding Reflection of a Lumberjack And a Car who must Sink a Lumberjack in A Baloon Factory	2006
AFFAIR PREJUDICE	A Fanciful Documentary of a Frisbee And a Lumberjack who must Chase a Monkey in A Shark Tank	2006
AFRICAN EGG	A Fast-Paced Documentary of a Pastry Chef And a Dentist who must Pursue a Forensic Psychologist in The Gulf of Mexico	2006
AGENT TRUMAN	A Intrepid Panorama of a Robot And a Boy who must Escape a Sumo Wrestler in Ancient China	2006
AIRPLANE SIERRA	A Touching Saga of a Hunter And a Butler who must Discover a Butler in A Jet Boat	2006
AIRPORT POLLOCK	A Epic Tale of a Moose And a Girl who must Confront a Monkey in Ancient India	2006
ALABAMA DEVIL	A Thoughtful Panorama of a Database Administrator And a Mad Scientist who must Outgun a Mad Scientist in A Jet Boat	2006
ALADDIN CALENDAR	A Action-Packed Tale of a Man And a Lumberjack who must Reach a Feminist in Ancient China	2006
ALAMO VIDEOTAPE	A Boring Epistle of a Butler And a Cat who must Fight a Pastry Chef in A MySQL Convention	2006
ALASKA PHANTOM	A Fanciful Saga of a Hunter And a Pastry Chef who must Vanquish a Boy in Australia	2006
ALI FOREVER	A Action-Packed Drama of a Dentist And a Crocodile who must Battle a Feminist in The Canadian Rockies	2006

Now, go ahead and extend the `film-list.php` file to make the web page a little more useful.

- a) Adjust the query:
 - a. Only show the films with a length between 70 and 120 minutes. (WHERE length ...)
 - b. Sort the films by length (ORDER BY ...)
- b) Show the length of the movie in an additional column
- c) Style the output, be creative and make it look nice using CSS.

Include the updated `film-list.php` file in your submission.



Task 2: Address Book Web-page with PHP and MySQL

Difficulty: Medium

Imagine you want to implement to create a web page that serves as an address book. You choose PHP and MySQL to do that.

The result can look something like this:

First Name	Last Name	Street Address	City
Max	Mustermann	Superstraße 1	Superstadt

We provide a skeleton to solve this task in the material:

http://www.medien.ifi.lmu.de/lehre/ws1617/mmn/uebung/material/mmn_skeleton_04_NF.zip

All you need to do is to look for every TODO in the `contacts.php` file and complete the instructions given there. Here is a summary of the tasks:

- Update the `$connectionInfo` array that stores all information necessary to connect to your database (see tutorial)
- Use `$connectionInfo` to connect to the database.
- Provide an SQL statement to create the `AddressBook` table (see previous assignment)
- Create an HTML form that takes a first name, last name, street address and city (see screenshot above). It's nice to have a "placeholder" attribute in each of them to let the user know what goes into the form. All fields should be required.
- Check if the user submitted the form. If s/he did, insert the data into the database with the corresponding statement (see previous assignment)
- Construct a dynamic HTML table that shows all contacts stored in the database (see screenshot above)

Please contact us, in case you require additional information to solve this task.

Submit all your source code files, but do not include any passwords (`$connectionInfo`...)!



Submission

Please turn in your solution via UniWorX as a ZIP archive. You can form groups of up to four people.

We encourage you to sign up for our Slack Team! All you need is a CIP account and an email address that ends in “@cip.ifi.lmu.de”. Ask us if you don’t know how to get them.

If you have questions or comments before the submission, please contact one of the tutors. They are on Slack: [@tobi.seitz](#), [@peterjuras](#) and [@thomas-weber](#). Remember, that they also want to enjoy their weekends 😊

It also makes a lot of sense to ask questions directly in our [#mmn-ws1617](#) channel. Maybe fellow students can offer help or benefit from the answers, too!