INF115 Compulsory Exercise 3

PHP Environment

In this exercise we will connect to and query a database using PHP. In order to setup a PHP development environment we recommend using XAMPP (https://www.apachefriends.org/index.html), you can find a guide explaining how to setup this environment on the mitt-uib INF115 webpage under:

INF115 > Pages > Connecting with XAMPP

Dataset

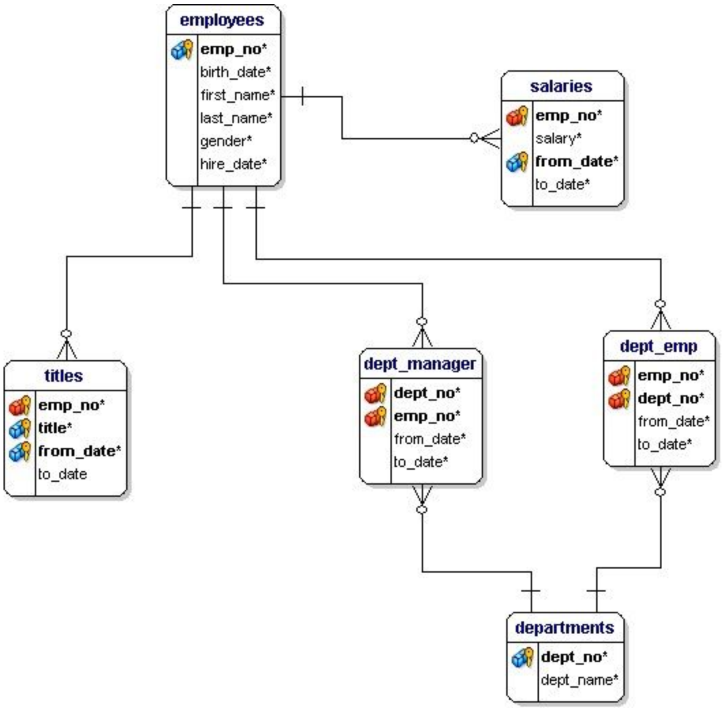
For the questions 4-10 below use 'employees' dataset, available at the following webpage:

https://github.com/datacharmer/test_db

To setup the database:

1. Create a new empty database called "employees" (http://localhost/phpmyadmin)
2. Download and unzip the repository
3. Open a terminal
4. Navigate to the directory (for me this is CD:O\Downloads\test_db-master\)
5. Type: **mysql -h localhost -u root employees < employees.sql**

The database schema is as follows:



Submission guidelines

Write a separate PHP/HTML script for each of the questions below. Put each of the scripts into a folder and submit the compressed (zipped) folder to mitt uib. We will use local version of the employees database to test your submissions so be careful not to change the names of entities and attributes within the database, and please do not set passwords in your script (points will be awarded for formatting and good naming conventions).

The deadline for Submissions is: **12:00 on 02nd May 2017.**

Tasks - Use HTML/PHP to complete the following tasks (for tasks 4 to 10 use the employee database specified above):

PHP and HTML forms (24%)

1. Write a PHP script to output "Good luck with the last compulsory!" in boldface.

2. Produce a HTML form that asks for “name”, “year of birth” and “present age” as input. The form should have a submit and a reset button. You should also implement a method to test the values that were entered into the HTML form. The script should check that all of the fields were completed, and that the values for “year of birth” and “present age” do not contradict each other. If there are any mistakes in the submission, a form with instructions (hints) on how to correct the submission should be returned.

3. Make a HTML form that asks a user for their preferred language. The form should have two check boxes corresponding to the options “English” and “Norwegian”. Depending on the option selected the script should return the HTML form specified in task 2 in either English or Norwegian (you do not need to reimplement all of the code from task 2, just the HTML form).

PHP for database querying (16%)

4. Write a PHP script to connect to the employee database and return a list of the employee's (first name and last name), sorted in alphabetical order by last name.

5. Make a script which returns all of the unique job titles from the employee database.

6. Use PHP to display a list of the tables in the employee database, with each table name on a separate line.

Further database queries (32%)

7. Produce a PHP script that displays a HTML table showing each of the attributes of each of the tables in the employee database. The database table name should be in the left hand column and the attribute name in the right hand column.

8. Using PHP query the employee database and produce a HTML table that shows how many employees there are in each department. Each row of the HTML table should contain the department name, the name of the manager and the employee count.

9. Write a script that queries the employees database and returns a list showing the the number of male and the number of female employees that were born in each decade.

HTML / database interface (24%):

10. Create a HTML form and a PHP script to provide information about the salaries in the employee database. The HTML form should allow the user to specify a particular year and to select if they wish to see the total salaries or the average salaries for the chosen year (note that the value of a salary can change in the middle of a year, and this should be dealt with appropriately). The requested information should be returned as a HTML page. Include checks to make sure that the user does not submit empty or invalid dates, and provide feedback to the user via the HTML form if there are problems.