PHP / SQL practical assessment

Time allowed: 4h00

Marking

At the end of the allocated time, you will have to deliver all of your files, and a mark out of 20 will be awarded according to the following scale:

- **3 points** for exercise 1
- **5 points** for exercise 2
- **4 points** for exercise 3
- **6 points** for exercise 4
- **2 points** on the following criteria:
  - Indentation of the code
  - Comments
  - Relevance of the name of the variables
  - Readability

Bonus : bonus points may be added if the optional functions have time to be carried out. Do it only if you have the time!

Tips

If the task seems overwhelming, don't worry! Each step is detailed to move forward little by little.

- Read the entire statement from the start, to know where you are going.
- Take the time to code, commenting on your code as soon as it is necessary, why not by copying the instructions in comment.
  The proofreader must be able to understand what you have done!
- Keep functions simple, which do only one thing, the better to navigate.
- Focus primarily on PHP rather than CSS!
- The more your code will be indented and readable, the easier the rest will be!

Each exercise is independent, be sure to separate them into different files or folders.

Exercise 1: “Let's join!

Assume the following SQL tables:

Table "artists”	Table "songs”	Table "categories”
• id • firstname • lastname • gender	• id • title • release_date • #categ_id • #artist_id	• id • title

In a separate file, write the SQL query to display a song (id = 7) and the artist name using a join.

In the same file, write the SQL query allowing to display all the songs (as well as the name of their category) of an artist (id = 2) using a join.

Note: Write only the SQL query, not PHP.

Exercise 2: "The trainee"

A trainee, for which you are responsible, must retrieve a list of users from a database but also give the possibility of adding them using a form.

The main script consists of two parts:

- The list of users and their information
- The form for adding a new user

The trainee begins in PHP programming. It does not indent or comment on its code. As an internship supervisor, you must help him correct and make his code work. Good indentation of the code is also necessary.

Implementation steps:

1. Download the files at this address:
   Evaluation Docs

2. Deploy the files *index.php* and *connect.php* on your local server and import the SQL script *users-install.sql*. This will create a new database containing a table *“users”.*

3. Help your trainee to debug his code, correct his errors and make it work.

Exercise 3: “Finding accommodation!

In this exercise, you will create a form.

The form will allow you to add a new housing item in the database and will contain the following fields:

- Title
- Surface
- Type (select : apartment or house)
- Date of creation
- Price
- Sold (checkbox)

The form will be processed in ajax.
You can use ajax in the classic way or using jQuery.

A success or error message is displayed depending on the result.
The more specific you are in error messages, the better.

---

To check if your code works, you can import the file 'exercise3.sql' into a database.

Exercise 4: "A banking service ..."

You work for a bank branch. As part of the site redesign, you must create a BankAccount () class which will have the following private properties:

- an account number (string: 8 characters)
- an amount (int)
- a withdraw limit (int)
- type (string: personal or pro)

Make the getters / setters allowing to validate the data type above as well as the constructor allowing to instantiate the class.

Add two methods:

- Withdraw: Withdraw is possible on the account at any time.
  The withdraw operation checks if the ceiling (withdraw limit) is not exceeded.
- Deposit: The deposit is possible at any time.
  The deposit is the addition of an amount to the account.

Both methods should update the amount on the bank account.

In a new file, instantiate the class so that you can display 3 different bank accounts.
For all 3 accounts, try to withdraw money until it is no longer possible using a loop and the previously created functions (withdraw).