



PHP Pratical Evaluation
Time allowed: 4h00

PHP / SQL Practical evaluation

Time allowed: 4h00

Marking

At the end of the allotted time, you will have to deliver all of your files, and a score out of 20 will be allocated 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 code
 - Comments
 - Pertinence of the names of variables
 - Readability

Bonus : bonus point can be added if the mandatory features have been done in the time allotted. To do only if you have time !

Advice

If the task seems exhausting, do not worry! Each step is laid out in detail for you to progress little by little.

- Read all the instruction before beginning, to know where you are going.
- Take the time to code, commenting on your code as soon as it's needed, why not by copying the instruction in a comment. The marker must understand what you have done!
- Keep to using simple functions which do only one thing, to better keep track of your work.
- Concentrate on PHP rather than CSS!
- The more indented and readable your code is, the easier it will be!

Each exercise is independent, make sure to separate them in different files or folders.

Exercise 1: "Let's join! "

Assume the following SQL tables:

Table "users"	Table "articles"
<ul style="list-style-type: none">• id• firstname• lastname• email• role	<ul style="list-style-type: none">• id• title• content• picture• date_publish• id_user

In a separate file, write the SQL query to display an article (id = 10) and its author using a join.

Note : Only write the SQL query, not PHP.

Exercise 2: "The trainee"

A trainee, for whom you are responsible, must retrieve a list of users from a database but also give the possibility to add a list using a form.

The main script consists of two parts:

- The list of users and their information
- The form to add a new user

The trainee is just beginning PHP programming. He does not indent or comment on his code. As a tutor, you need to help him correct and run his code. Good indentation of the code is also needed.

Implementation steps:

1. Download the files from this address: http://wf3.axessweb.io/evaluations/php-sql/evaluation_docs.zip
2. Deploy the index.php and connect.php files on your local server and do not forget to import the SQL users-install.sql script. This will create a new database containing a "users" table.
3. Help your trainee to debug his code, correct his mistakes and make it work.

Exercise 3: "Let's tidy up! "

In this exercise you will need to use PHP to handle a request to process an Ajax form.

The form will add a new vehicle to the database and will contain the following fields:

- Make
- Model
- Year
- Colour

The form will be processed in PHP. A success or error message will be displayed depending on the entry.

Note: you do not need to use AJAX or SQL to do this, only PHP.

Exercise 4: "A cat that has class ..."

You work for the SPA (Society for the Protection of Animals). As part of the redesign of the site, you must create a class `Cat ()` that will have the following private properties:

- First name (string of 3 to 20 characters)
- Age (int)
- Color (string of 3 to 10 characters)
- Sex (string: male or female)
- Race (string of 3 to 20 characters)

Make getters/setters to validate the data types above and the constructor to instantiate the class.

Add a `getInfo()` method to return all of the properties as an array.

In a new file, instantiate the class so that you can display 3 different cats and display the result using the previously created `getInfo()` method.