

PHP Practical Evaluation

Time allowed : 4h00

Marking

At the end of the allotted time, you must deliver all of your files, and a score out of 20 will be awarded according to the following scale :

- **3 points for Exercise 1**
- **5 points for Exercise 2**
- **8 points for Exercise 3**
- **4 points for the following criteria:**
 - Indentation of code
 - Comments
 - Relevance of variable names
 - Readability

Bonus: bonus points can be added if you have time to complete the optional features. To be done only if you have the time!

Advice

If the task seems exhausting, don't t worry ! Each step is detailed so that you can progress little by little.

- Read all the instructions at the beginning, so that you where you are going.
- Take the time to code, commenting on your code as soon as it is needed, why not by copying the instruction into comment.
The person marking your evaluation must be able to understand what you did !
- Keep the functions simple, which do one thing only, to help you better find your way.
- Concentrate on PHP rather than CSS !
- The more indented and readable your code is, the easier it will be !

Exercise 1 : « Introduce yourself ! »

Create a table in PHP containing the following information:

- First name
- Last name
- Address
- Postal code
- City
- Email
- Telephone
- Date of birth in English format (YYYY-MM-DD)

Using a loop, display the contents of this array (keys + values) in an HTML list.

The date is to be displayed in French format (DD / MM / YYYY).

Bonus :

Manage date of birth display using the DateTime class

Exercise 2: "We're going on a trip"

Create a function to convert an amount in euros to an amount in US dollars.

This function will take two parameters:

- Amount (of type int or float)
- Currency of exit (only EUR or USD).

If the second parameter is "USD", the result of the function will be, for example:

1 euro = 1.085965 US dollars

The necessary checks must be made to validate the parameters.

Exercise 3: "How about watching a movie? »

You work for a cinema and need to create a movie database. Your database will be called "exercise_3". You will then need to create a script that will add and display movies. Follow the steps. Follow the steps.

Step 1 :

This table, named "movies" will consist of the following fields :

- title (varchar): the name of the movie
- actors (varchar): the names of actors
- director (varchar): the name of the director
- roducer (varchar): the name of the producer
- year_of_prod (year): the year of production
- language (varchar): the language of the film
- category (enum): the category of the film
- storyline (text): the synopsis of the movie
- storyline (text): the synopsis of the movie

Do not forget to create an ID for each movie and auto-increment.

Step 2 :

Create a form to add a movie and make the necessary checks.

Prerequisites :

- The fields "title, name of the director, actors, producer and synopsis" will have at least 5 characters.
- The fields: year of production, language, category, must be a drop-down menu
- The link for the trailer must be a valid URL
- In case of input errors, error messages will be displayed in red

Each movie will be added to the created database.. A "success" message will confirm the addition of the film.

Step 3 :

Create a page which lists the movies in the database in an HTML table. This table will contain, per film, only the name of the film, the director, and the year of production..

One column of this table will contain a "see more info" link to see the detailed information sheet of a film.

Step 4 :

Create a page that dynamically displays a movie detail. If the film does not exist, an error will be displayed.