



## PHP / SQL

# Intermediate practical assessment

11th November 2020

**Time allowed : 4h00**

## Marking

At the end of the allotted time, you must deliver all of your files in a Github repository, and a score out of 20 will be given based on to the following scale:

**Exercise 1 : 2 points**

**Exercise 2 : 3 points**

**Exercise 3 : 5 points**

**Exercise 4 : 8 points**

**Other : 2 points** ◦ Indentation and readability ◦ Code Comments

*NB it is not permitted to copy and use code obtained from previous student of this training - doing so is considered cheating and will be sanctioned*

## Exercise 1: Setting up the project

Create a new project on git. (To submit your work at the end of the evaluation, you will have to upload your sources to a GitHub repository).

!! Set your repository as private.

## Exercise 2: Introduce yourself

Create a PHP array containing the following information about yourself :

- First name
- Last name
- Address
- Postal code
- City
- Email
- Telephone
- Birthdate (YYYY-MM-DD)

Using a loop, display the content of this array (keys and values)

## Exercise 3: Speed Unit Converter

Create a function to convert miles per hour into kilometers per hour.

This function has 2 arguments:

- The speed (int type or float type)
- The unit of speed (KMPH or MPH).

If the second parameter is "MPH", you have to convert to MPH (for example): 1 kmph = 0,621371 miles per hour

You have to perform the necessary verifications in order to validate the arguments.

## Exercise 4: Rent-a-Car

### Part 1

Create a database and name it "car\_renting". Inside it, you must create a table that you will call "cars" with the following fields:

Table : cars Fields :

- id\_car (int)
- name (varchar)
- brand name (varchar)
- price (double)
- photo (varchar)
- type (enum: sport or break)
- description (varchar)
- year\_of\_prod (year)

### Part 2 :

Create a form to add cars in the table 'cars'.

Prerequisites:

- The field name, brand\_name, price, type and year\_of\_production are required.
- The field 'year\_of\_prod, type' will be a drop-down menu (mandatory).
- The price must contain numbers only.
- Error messages will be displayed in red

Each car will be added to the created database. A success message will be displayed.

### Part 3 :

Create a page which for displaying the cars. We should be able to find all the cars with their respective information. You have to display the cars in a HTML table. If the description is more than 30 characters long, separate the text by adding "...". In this table, you will add a column which will contain a "more" link.

### Part 4 :

Create a page which will dynamically display the detailed content of a car. If the car doesn't exist, an error message will be displayed.