Introduction to programming (PHP)

Part 0 - Program

We will offer every day of the reading, activities / exercises and resources. This is described in the corresponding parts.

This first part describes what we expect of you today.

1 - Be curious!

See section *"Playback"*.

2 - Do activities in order and do them

And do not hesitate to go further **if and only if** you need it time and as you go well.

You will gain knowledge over your advancement activities. Use them to solve problems from a maximum of exercise routines that we propose.

3 - Widen your social networks and be curious (repeat)!

Remember to share your findings with others via **Slack**!

Part 1 - Reading

Understand what constitutes programming! (3 min video)

Short video introduction to online courses. Always helpful. But, the rest of the videos is expensive.

Https://www.video2brain.com/fr/tuto/en-quoi-consiste-la-programmation

Part 2 - Activities

PHP: the basics

PHP is a powerful tool, it is used for the development of sites worldwide. With Php a site becomes dynamic, ie that the pages can be generated by the user, the language, language etc.

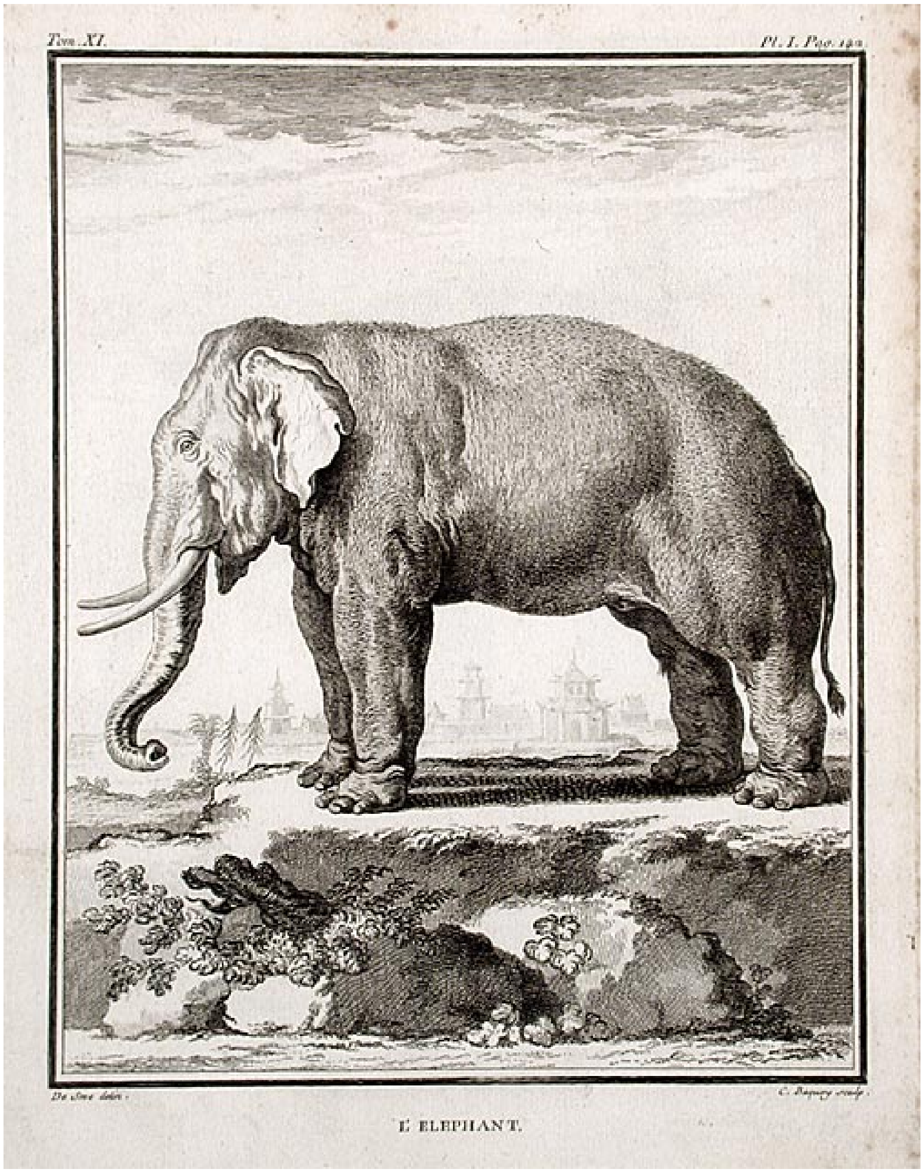

Figure 1: The Elephant

After the first two parts of the course: *Part 1 - The basics of PHP* and *Part 2 - Transmit data page to page*, you will realize. Minimum 2 exercises in the categories *"PHP Introduction"*, *"Loops"*, *"Tables and blocks"*.

After *Part 3 - Store information in a database* do the exercises of the category *"PHP / MySQL"*.

(For the years see Part 3 - Exercises *that is in this document).*

https://openclassrooms.com/courses/concevez-votre-site-web-avec-php-et-mysql

PHP / MySQL

The real interest of a type of **backend** language like PHP is to manage data (store, reproduce, process, etc..).

For example, for a contact form for a website it would be nice to store some information previously returned share.

In order to store and retrieve data from a database is used Data.

https://openclassrooms.com/courses/concevez-votre-site-web-avec-php-et-mysql/presentation-des-bases-de-donnees-2

Part 3 - Exercises

What to do after the *"Part 2 - Transmit data from page to page"*

To start, use a display page that retrieves data revenue per user and data processing page.

Category: Introduction PHP

Exercise 1 - Basket calculation

Calculate the total price of a basket. The basket have 3 products different. For each product, leave the option to enter the `price` (excluding tax), the `VAT rate` (percentage), and `quantity`.

Exercise 2 - Compare prices

Compare prices 4 products and display the name of the cheaper product and the name of the most expensive product. Give the opportunity to return the product `name` and `price`.

Exercise 3 - The bank account

You must view the status of your bank account. If it is `negative`, you have displayed **“Deficit”**, if it’s to `0`, you have displayed **“In balance”** and if `positive` you have displayed **“Credited”**. Let the ability to enter the account status.

Exercise 4 - Number of days in a month

Display the number of days in a month. Months are selected from a `select box`. We consider in this exercise that the month of February always has 28 days.

Exercise 5 - Validate a date

Confirm the anniversary date back by a user. A date valid must be of the form `"mm/dd/yyyy"` ie `"15/9/2019"`.

Exercise 6 - Alternative parking

In a street where the alternate parking practice, from 1 to 15 month, you park on the side of houses with an `odd number`, and the rest of the month, it is the other side station. Let the opportunity to choose the `date` and `house number` before which you parked, then view if you are well parked or not.

Category: Loops

Exercise 1 - Display the first `n` digits

The user enters two numbers corresponding to the `beginning` and at the `end` a range of numbers.

You must show all the numbers in this range respecting rules below:

- Numbers must be displayed in list form
- The even numbers are blue
- Odd numbers are orange
- The numbers that are perfect squares are bold

Note: The range of numbers must be positive numbers (Greater than `0`)

Exercise 2 - Maximum numbers

10 Display text fields. The user will fill them with numbers Of his choice. You must display the greatest number.

Exercise 3 - suites Generation

On a range of numbers from `0` to `100`, view the suites below in different blocks:

- a) A series of numbers with *"The not growing"*:

  `1, 2, 4, 7, 11, 16, ...`

- b) A so-called many more *"lame"*:

  `1, 2, 4, 5, 7, 8, 10, 11, ...`

- c) The Fibonacci sequence:

  `0, 1, 1, 2, 3, 5, 8, 13, 21, ...`

Exercise 4 - Numerology

Convert your name and first name in a number of the principle of numerology, each letter has a weight. e.g.:

`A = 1, B = 2, C = 3, etc..`

Show the weight of your first name and weight of your name. For example if your name is Bob, this gives:

`B = 2, O = 15, B = 2 so the weight is 2 + 15 + 2 = 19`

Exercise 5 - Game of the range

Simulate the game of the range. This game is to try to discover any number between `1` and `100` inclusive, drawn by computer (primitive `rand(min, max)` returns an `integer` between `min` and `max`). The user is entitled to a maximum of eight trials. With each test you must display a code message **"number given too small"** or **"number given too big"**. In conclusion, is **"bravo, you found in *[number]* test(s)"** or **"sorry, the number was *[value]*"**.

Category: Tables

Exercise 1 - Sum

Write an algorithm that calculates and displays the `sum` of the integers a board.

Exercise 2 - Maximum and minimum

Write an algorithm that displays the `largest` and `smallest` number in an array of integers.

Exercise 3 - Number of array elements

Write an algorithm that displays the `number of items` there are in a painting.

Exercise 4 - Largest gap

An algorithm that calculates the `largest gap` between two integers consecutive in a table.

Exercise 5 - Table ordered?

Write an algorithm that displays **"true"** if an array of integers is ordered (strictly) increasing the values, or **"false"** if not not.

Exercise 6 - Occurence count

Write an algorithm that displays for each digit in the number many times it appears in a number. Thus, for the number `10502851125`, the display will mention that the number `0` appears 2 times, `1` appears 3 times, `2` times appears 2, `5` appears 3 times and `8` appears one times (the display will not refer to the numbers do not appear).

"Blocks and sort"

Exercise 1 - The Caesar cipher

Since ancient times, politicians, soldiers, men business seeking to keep secret the important messages they should send. The emperor Caesar used a technique (a one says encryption) which carries now his name replace each letter of message with the letter that is k position later in the alphabet (Cyclically).

Example: If k is `2`, then the clear text `"CESAR"` becomes `"EGUCT"` when it is encrypted and the text `"heck"` becomes `"BWV"`.

Of course, it is necessary that the sender and receiver are agreed on the value of `k`.

You must allow the home user a text to be encrypted to a ciphertext to be decrypted and choose the fill `k` value.

Exercise 2 - Validity of date

Resume validation algorithm developed to date in the part *"Control Structures"* and make it modular. Which means you will have several methods in your code: * A function that verifies the validity of the day. She will receive setting the day and return a boolean (**"true"** if the day is valid, otherwise **"false"**); * A function that verifies the validity of the month (in number, 1 at 12). She will receive parameter in the month and return a boolean (**"true"** if the month is valid, otherwise **"false"**); * A function that verifies the validity of one year. She will receive setting the year and return a boolean (**"true"** if the year is valid, otherwise **"false"**); * A function that uses the previous three to check that Date is invalid. This function will receive as parameter the day, month and year, each in the form of an integer.

Will display **"true"** if the user-encoded date is valid, otherwise you will display **"false"**.

Warning ! We will not consider leap years in this exercise.

Exercise 3 - Alternating parking with license plate

Repeat the exercise *"Alternative parking"*. Now, in addition previous constraints to park a car must have an even number of `1` to `15` and an odd number the rest of the month. Add a field to enter the license plate number. We take into account that the new license plates. eg `AA 555 ZZ`

Category: PHP / MySQL

Create a vendor database using **phpMyAdmin**.

Exercise 1 - Identity

In the database of vendor create the table `user`. This table will have the fields: `id`, `password`.

Add users to the table through the interface `phpmyadmin`.

Create a form to allow the user to enter his username and password.

Create another page that checks if the couple username and password passes, returned by the user is valid. Display the message **"Welcome"** if the user is valid and **"Password incorrect password or username"** otherwise.

Exercise 2 - Vendor

In the database `vendor`, create a table named `vehicle`. This vehicle will table fields as: `id`, `name`, `color`, `brand`, `consumption`, `power`, `price`.

We will interact with these databases.

First, create the form to add a vehicle in the database.

Create a page that displays, in a table all the information vehicles.

Implement functionality to allow the removal of a vehicle.

[For the brave] Challenge - Directory reorganization

For the last year, we give you some instructions, no runs, this is for you to get by for a solution that works and build a clean code, readable, modular, reusable.

Write a program that rearranges the elements of a directory of your machine.

It will store the files by extensions in one folder. By example all `.mp3` will be stored in the *"mp3"* folder, all files. `.odt` will be stored in the *"odt"* folder.

Resources

- PHP documentation: http://php.net
- PDO Cheat Cheets: http://www.mustbebuilt.co.uk/2012/10/16/pdo-cheatsheet/

The scope of a variable

The variables have different scopes. Each variable exists in a context.

Http://php.net/manual/fr/language.variables.scope.php

In order to make the relationship between tables must be used system `primary key` and `foreign key`:

https://openclassrooms.com/courses/administrez-vos-bases-de-donnees-avec-mysql/cles-primaires-et-etrangeres