ALGONQUIN COLLEGE | Centre for Continuing & Online Learning

# CST8285 – Assignment 6: Let's PHP ! [worth 10% of the final mark]

This assignment relates to the following Course Learning Requirements:

CLR 1: Write program code using structured programming, and top-down coding techniques
CLR 2: Produce tested code that executes correctly and consistently through the use of valid and invalid data.
CLR 3: Prepare program documentation using prescribed program specifications
CLR 4: Debug program problems using manual methods and computerized tools in an appropriate manner
CLR5: Modify an existing program according to program specifications

## Objective of this Assignment:

The purpose of this assignment is to put in practice the theory acquired so far in PHP programming language.

## Pre-Assignment Instructions:

### Big picture

Upon completing this assignment, you should be able to:

- Create PHP files
- Declare PHP variables and functions
- Understand PHP data types
- Determine PHP variable types
- Upload the .php files to your home directory
- Accept and process data from the browser using the GET (form) and POST (form) methods

**Assignment Tasks:**

## Challenge 1 - Howdy World using a GET method

Create a file called **welcome.php** that accepts **two parameters in the URL**: **firstName** and **lastName**. Have the script write a "Howdy" message using the name provided by the user. Error checking must be done in the following ways:

- If the first name is set and not blank, but the last name is not set or is blank, display a welcome message using only their firstName.
  - Firstname NoLastName; i.e. Howdy Firstname
- If the last name is set and not blank, but the first name is not set or is blank, display a welcome message with blank firstName lastName.
  - i.e. Howdy LastName
- If neither name is set, or if they are both blank, display a “Howdy no names” message
  - i.e. Howdy no names
- **You need to make the parameters to be EXACTLY firstName and lastName, case sensitive and NO OTHER variations like first, last, firstname, lastname … anything other than firstName and lastName will end up with a zero mark for this section. I will only test firstName and lastName and if it does not work with them, it is considered a full failure !!!**

## Challenge 2 - Currency Converter using a POST method

Create a file called **currency.php**. In this task, you will be creating a currency converter. All programming must be done in PHP (no JavaScript). The requirements are:

- The currency input value must be accepted from a text box, in an **HTML form.**
- The user must be given the option to:
  - Set the currency type of the input value (converting from)
    - Types are: Canadian, American, Euro, British Pound and Chinese Yuan
    - You need to use the appropriate flags as icons beside each currency type
    - Hint: Use radio buttons
  - Set the currency type of the desired output value (converting into)
    - Types are: Canadian, American, Euro, British Pound and Chinese Yuan
    - You need to use the appropriate flags as icons beside each currency type
    - Hint: Use radio buttons
- Error checking must be done to ensure that the user has entered a number. If they did not, they should be notified of the error
- Not using flags will results in a significant deduction

- Check for the case when converting from/to with the same currency that the amount does not change
- If the user entered a number, the number should now be converted to the requested currency unit. The conversion logic should be programmed in a function. The function should take three parameters: the amount to be converted, the input currency and the currency to be converted to. Once complete, the function should return the converted amount.
- **Finally, display the original amount and the converted amount to the user, with appropriate flags and currency symbol.**
- For conversion rate, please visit www.xe.com or even a Google search can suffice.

Once you are ready, you need to validate the html documents again and make sure no errors are left in them.

Submit it through Brightspace's Module 9 Assignment 6

## What to submit

Submit your project through Brightspace and in a **zip file** containing all the files you have developed so far but make sure you include:
<welcome.php>
<currency.php>

I will only look for welcome.php and currency.php. If you name your files any other name or the case sensitivity is different, I will not look for them and unfortunately, a zero will be granted. Do not forget to include them, along with other required files in a zip file.
Any other related file/s, flags icons but the marking will be based on these two challenges listed above only!

P.S. Part of the training received through this course is to learn how to fulfill requirements properly. That's why I am becoming more strict, not that I want to be mean but simply to help develop your skills in completing what the customer wants/needs exactly.