Lab 07: Introduction to Server-Side Development with PHP

NOTE: Refer to *"Guidelines - Setting up a development server (XAMPP)"* to install PHP, MySQL and Apache, before doing your lab exercises. If you have problem to setup a development server, you may use an online PHP editor (e.g. https://www.runphponline.com/) to test your PHP codes.

1. Write a PHP program to print name. Create a file called "`printName.php`". Inside the file, write a new function named "`printName`" that:
   - takes 3 arguments: **first name** (in string), **last name** (in string) and a **number**
   - it should print out the name for a number of times in separate lines when you call the "`printName`" function, for example:

   ```
    //should print the first name and last name 10 times
   printName("John", "Smith", 10)
   ```

   Reference: https://www.w3schools.com/php/php_looping.asp

2. Create a file called "`printNameObj.php`". Rewrite your PHP codes for Question 1 in an object-oriented style:
   a. Define a new class named '`Person`' consisting of **two public properties** (`$firstName` and `$lastName`) and the following **two methods**:
      i. `function __construct($firstName, $lastName)` - A constructor function to create an object from a class;
      ii. `public function printName($num)` – A public function called '`printName`' with one argument (`$num`) to print out the `$firstName` and `$lastName` of a Person's object for a number of times in separate lines when you call this function.

   b. After defining the class, in your PHP codes, call the constructor function to create an object named '`$person1`' from `Person` class, with the first name 'John' and last name 'Smith'.

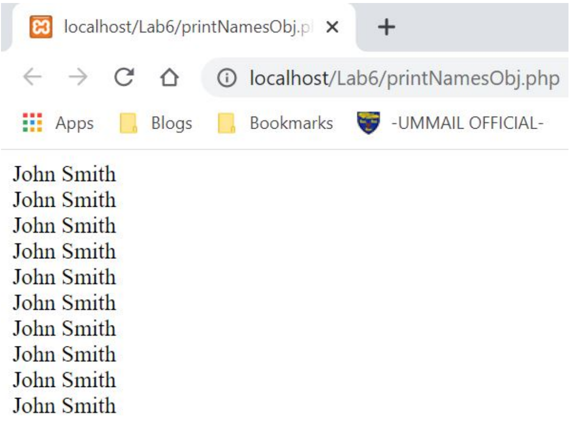
   c. Call the '`printName`' function to print the name of the new object, `$person1` for **10 times**.

   References:
   https://www.w3schools.com/php/php_oop_classes_objects.asp

https://www.w3schools.com/php/php_oop_constructor.asp

https://www.w3schools.com/php/php_oop_access_modifiers.asp

Sample output:


localhost/Lab6/printNamesObj.p
localhost/Lab6/printNamesObj.php
Apps
Blogs
Bookmarks
-UMMAIL OFFICIAL-

```
John Smith
John Smith
John Smith
John Smith
John Smith
John Smith
John Smith
John Smith
John Smith
John Smith
```

3. Write a PHP program to calculate average grade. Create a file called “grader.php”. Inside the file:
   - Define a variable called “`$scores`” to takes a single parameter: an array of test scores (all numbers)
   - Compute and return the average score in the array, rounded to one decimal place.
   - Write a new function named “`checkGrade`” that use program control statement to print the grade of the average score. The grading scheme is as follows:
     - 80 – 100: A
     - 60 – 79: B
     - 40 – 59: C
     - 20 – 39: D
     - 1 – 19: E
     - 0: F
   - Examples:

     Sample scores: 90, 98, 89, 100, 100, 86
     Sample output:
     Average score for [90 98 89 100 100 86] is 93.8
     Average grade is A

     Sample scores: 40, 65, 77, 82, 80, 54, 73, 63, 95, 49
     Sample output:
     Average score for [40 65 77 82 80 54 73 63 95 49] is 67.8
     Average grade is B

4. Write a PHP program to take two input numbers, reverse the numbers, compute the sum of the two reversed numbers and display the sum.

   Sample Input: 13, 14
   Sample Output: 72

   Sample Input: 305, 794
   Sample Output: 1000

   Sample Input: 130, 1
   Sample Output: 32

   Hints:

   - Write one function called "`reverseInteger`" to reverse the two numbers
   - Write one function called "`computeReverseSum`" to calculate the sum of the two reversed numbers.
   - The result will not be unique for every number for example 31 is a reversed form of several numbers of 13, 130, 1300 and etc. Therefore, all the leading zeros will be omitted.