Lab 7 – Programming Arrays, Objects

Purpose

- Programming using Arrays
- Programming using Objects and Array of Objects
- Upload your website to a Web server

Due Date

- This lab must be handed in:
  Friday November 06, 2020 – before midnight

Assessment

- This Lab is worth 2% of your total course mark.

Assigned Readings

- **Lecture Slides** posted on Brightspace
  - Module 3 -> Part 3
- The following chapters of **Fundamentals of Web Development** will be useful in completing this exercise:
  - Chapter 9
  - Chapter 10

Lab Supplies

To complete this lab you will require the following lab supplies:

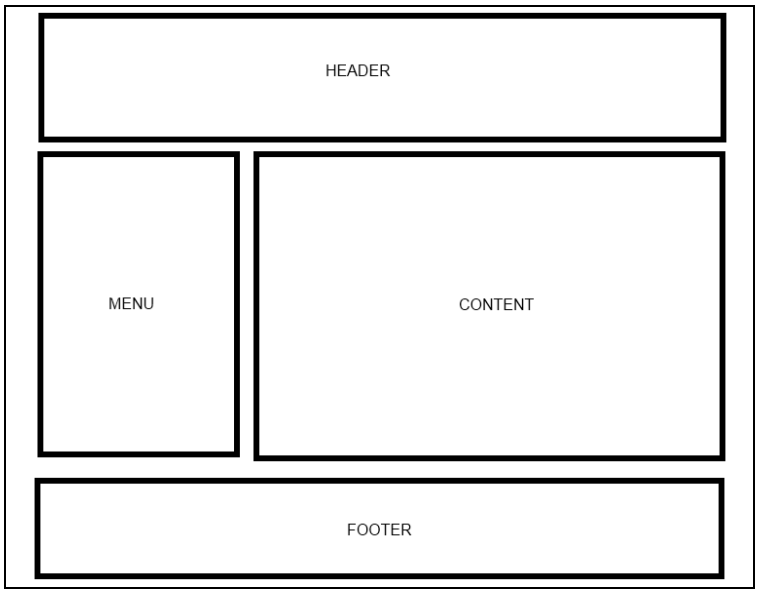
- Lecture Slides (**Module 3 -> Part 3**) posted on Brightspace
- Textbook: **Fundamentals of Web Development** by Randy Connolly and Ricardo Hoar
- EasyPHP, or other WAMP server
- Eclipse, Notepad++ (or other text editor, or IDE)

Summary of Tasks

1. Develop the logic to solve and display the output for Arrays.php.
2. Develop the logic to solve and display the output for ArrayofObjects.php.
3. Upload your website to the webserver
4. View your webpage using a web browser
5. Submit Lab Link on Brightspace
6. Submit source code of all PHP files on Brightspace

Task 1

Implement the following Design Pattern to create a 'Common Look and Feel' to be used on every page of your website.



Your web site will include the following PHP scripts:

- Header.php
- Footer.php
- Menu.php
- Arrays.php
- ArrayOfObjects.php

Header.php

Header.php must contain a script to display a Common Header that will appear on every page. The header must display Program Name and Course Name

Footer.php

Footer.php must contain a script to display a Common Footer that will appear on every page. The footer must contain Student Number, First Name, Last Name, and Email Address

Menu.php

Menu.php must contain a script to display a Common Menu to be shown on every page. The menu must contain links to Arrays.php and ArrayofObjects.php

Arrays.php

Create a PHP script that will perform the following tasks.

1. Create a multidimensional array called **'$November'**.
2. Implement the array as follows:

$November

Friday	1st	2nd	3rd	4th	
	6	13	20	27	
Saturday	1st	2nd	3rd	4th	
	7	14	21	28	
Sunday	1st	2nd	3rd	4th	5th
	1	8	15	22	29
Monday	1st	2nd	3rd	4th	5th
	2	9	16	23	30
Tuesday	1st	2nd	3rd	4th	
	3	10	17	24	
Wednesday	1st	2nd	3rd	4th	
	4	11	18	25	
Thursday	1st	2nd	3rd	4th	
	5	12	19	26	

3. Place a header (h2) on the page with the following word 'Output-1'. Use appropriate built-in function to display the contents of the array as ***key->value*** pair.

   Sample output is as follows:

Array ([Friday] => Array ([1st] => 6 [2nd] => 13 [3rd] => 20 [4th] => 27)
[Saturday] => Array ([1st] => 7 [2nd] => 14 [3rd] => 21 [4th] => 28)
……..
……..)

4. Place a header (h2) on the page with the following word 'Output-2'.
   Apply appropriate **loop** statement to display the following text where you must use the ***values****(e.g., 3,10,17, 24, ….)* and ***keys****(e.g., 1st, 2nd, …., Friday, Saturday, ….)* of the **$November** array:

6 is the 1st Friday in November
13 is the 2nd Friday in November
20 is the 3rd Friday in November
27 is the 4th Friday in November.

7 is the 1st Saturday in November
14 is the 2nd Saturday in November.
21 is the 3rd Saturday in November.
28 is the 4th Saturday in November
.
……
……
26 is the 4th Thursday in November.

5. Place a header (h2) on the page with the following word 'Output-3'.
   Use appropriate built-in **array** function to display the contents of the array in reverse order.
6. Place a header (h2) on the page with the following word 'Output-4'.
   Add '***31***' as the '**5th**' element of '**Tuesday**' in ***$ November*** array and display the contents of the array.
7. Include common Header, Menu and Footer to the page.

ArrayOfObjects.php

Implement the following PHP scripts:

Employee.php:
Define a class ***Employee*** which contains protected properties: *employeeId, firstName, lastName.* Create a ***constructor*** method that takes in *employeeId, firstName, lastName.* You need to write the getters and setters of the corresponding properties of the class.

The prototype of the Employee class is below:

```
class Employee{
      protected $employeeId;
      protected $firstName;
      protected $lastName;

      function __construct($employeeId, $firstName, $lastName){
      }
      public function getEmployeeId(){
      }
      public function setEmployeeId($employeeId){
      }
      public function getFirstName(){
      }
      public function setFirstName($firstName){
      }
      public function getLastName(){
      }
      public function setLastName($lastName){
      }
   }
```

Supervisor.php:
Define a derived class ***Supervisor*** that inherits from the ***Employee*** class and contains a private property: *employees*. Create a ***constructor*** method that takes in *employeeId, firstName, lastName, employees.* You may need to call the parent ***constructor*** for *employeeId, firstName, lastName.* You need to write the getters and setters of the corresponding property of the class. The prototype of the Supervisor class is below:

```
class Supervisor extends Employee{
  private $employees;
  function __construct($employeeId, $firstName, $lastName, $employees){
  }
  public function getEmployees(){
  }
  public function setEmployees($employees){
  }
}
```

ArrayOfObjects.php:

Include Employee.php and Supervisor.php.

Instantiate (Create) six objects (employee1, employee2, employee3, employee4, employee5, employee6) of the ***Employee*** class.

Create an array of objects named ***employees1*** which will consist of employee1, employee2, and employee3. Create another array of objects named ***employees2*** which will consist of employee4, and employee5, and employee6.

Instantiate two objects (**supervisor1**, **supervisor2**) of the ***Supervisor*** class such that the first three employees (employee1, employee2, employee3) will be supervised by **supervisor1** and rest of the employees (employee4, employee5, employee6) will be supervised by **supervisor2**.

Display the employee ID, first and last name of the employees supervised by each of the Supervisors. You also need to display the name of the Supervisor of each employee.

Note: In ***ArrayOfObjects.php,*** you must invoke ***getEmployees()*** function of the ***Supervisor*** class to retrieve the employees of the corresponding Supervisor and then invoke the required functions (i.e., getters) of the Employee class to display the properties of each employee.

Sample output for ***ArrayOfObjects.php*** is as follows:

```
Employee Id: 1, Name: Chris Rogers, Supervisor: Adam Phillip
Employee Id: 2, Name: Matt Prior, Supervisor: Adam Phillip
Employee Id: 3, Name: Cindy Burnskill, Supervisor: Adam Phillip
Employee Id: 4, Name: Elizabeth Ford, Supervisor: Nicolas Jones
Employee Id: 5, Name: Doug May, Supervisor: Nicolas Jones
Employee Id: 6, Name: John Hopkins, Supervisor: Nicolas Jones
```

Include common Header, Menu and Footer to the **ArrayOfObjects.php** web page.

Task 2

Create Lab 7 submission folder '**Lab7**' and copy **Arrays.php**, **Employee.php.php**, **Supervisor.php**, and **ArrayOfObjects.php** and any other required files **(**e.g**. css file)** into this folder**.**

Task 3

Upload your websites for Lab 7 into a Web Hosting Server by uploading '**Lab7**' folder inside the 'public_html' directory of 'SiteGround' Web Hosting domain using DashBoard.

The 'File Upload' instruction is posted on Brightspace (**Course Contents -> Module 1 -> Part 2 -> SiteGround_FileUpload_Instruction.docx**).

Task 4

View your websites for Lab 7 using a web browser. Open a web browser and navigate to the following web address:

http://your_web-hosting_domain_name/Lab7/<filename>

For example, the sample URL for Lab 7 on an arbitrary web hosting domain is:

rejaulc.sgedu.site/Lab7/Arrays.php

where '*rejaulc.sgedu.site*' is the name of an arbitrary web hosting domain, '*Lab7*' is the submission folder for Lab 7, '*Arrays.php*' is the homepage of Lab 7.

Task 5

Once you have confirmed that your webpage is available online, you are ready to hand in your lab.
Create a compressed file (**Lab7.zip**) which will contain the following PHP files:

- Header.php
- Footer.php
- Menu.php
- Arrays.php
- ArrayOfObjects.php, Employee.php, Supervisor.php

N.B. Please keep in mind that ONLY .**zip** file is accepted as the format of the compressed file.

Create a word document (**Lab7.doc**) in which write the following Information:

- Student Number
- First Name
- Last Name
- The URL, or hyperlink of the home page (Arrays.php) of Lab 7

To hand in your lab, go to Brightspace and navigate to *Course Content* → *Labs* and click on 'Lab 7 – Programming Arrays, Objects' link.

Upload the word document (**Lab7.doc**) and the compressed file **(Lab7.zip**) on Brightspace. Finally, click the 'Submit' button to send the lab to your professor.

IMPORTANT NOTE: If the URL, or hyperlink, does not direct the professor to the lab you will receive a ZERO for the lab assignment.

IMPORTANT NOTE:
If you do not upload either Lab7.zip or Lab7.doc on Brightspace, you will receive a ZERO for the lab assignment.