Server-side Web Development

Units 01, 02 and 03

Practice 05 Solution

IES Jaume II El Just
Tavernes de la Valldigna
Departament d'Informàtica
Curs 2023-24

1 PRACTICE 05 SOLUTIONS

1.1 Exercise 01

Multiplication Table.

Create a variable with a numeric value (from 1 to 10) and then display an HTML table with the multiplication table of the given number.

8	x	1	=	8
8	x	2	=	16
8	x	3	=	24
8	x	4	=	32
8	x	5	=	40
8	x	6	=	48
8	x	7	=	56
8	x	8	=	64
8	x	9	=	72
8	x	10	=	80

Figure 1: Multiplication table

```php
<?php
$value=8;
?>
<table style="border:2px solid blue;">
    <?php
        for($i=1;$i<=10;$i++) {
            echo "<tr>";
            echo "<td>$value</td>";
            echo "<td>x</td>";
            echo "<td>$i</td>";
            echo "<td>=</td>";
            echo "<td>".($value*$i)."</td>";
            echo "</tr>";
        }
    ?>
```

```
16 </table>
```

1.2 Exercise 02

Classify elements from an array

Given an array with 10 numbers, display on the screen how many numbers are with one digit, two digits, three digits and more than three digits. An example: with this array

```
1 $array=array(10,4,44,233,1000,9,99,999,3,27);
```

The output should be

Numbers with one digit: 3
Numbers with two digits: 4
Numbers with three digit: 2
Numbers with > 3 digits: 1

Figure 2: Numbers by digits

```
1  <?php
2  $counter1=0;
3  $counter2=0;
4  $counter3=0;
5  $counter4=0;
6  $array=array(10,4,44,233,1000,9,99,999,3,27);
7  foreach($array as $value) {
8      if($value < 10) {
9          $counter1++;
10     }
11     elseif($value < 100) {
12         $counter2++;
13     }
14     elseif($value < 1000) {
15         $counter3++;
16     }
17     else {
18         $counter4++;
19     }
20 }
21 echo "<br/>Numbers with one digit: ".$counter1;
22 echo "<br/>Numbers with two digits: ".$counter2;
23 echo "<br/>Numbers with three digits: ".$counter3;
```

```php
echo "<br/>Numbers with > three digits: ".$counter4;
?>
```

1.3 Exercise 03

Function to extract first and last character of a string

Create a function that receives as a parameter a string. The function will return an array of 2 positions: in the first position we'll get the first character of the string, and in the second position we'll get the last character of the string. For example:

```php
$originalString='I am a student';
// let's suppose that the name of the function is f
$array=f($originalString);
echo "First: ".$array[0]." Last: ".$array[1];
```

The output should be

First: I Last: t

Figure 3: First and last characters

```php
<?php function ex3($cadena) {
    $firstC=substr($cadena,0,1);
    $lastC=substr($cadena,strlen($cadena)-1,1);
    $array=array($firstC,$lastC);
    return $array;
}
$a=ex3("I am an student");
echo "First character ".$a[0].", last character ".$a[1];
echo "<br/>";
?>
```

1.4 Exercise 04

Creating a function to sum an array

Create a function that receives an array as a parameter. The array must have numbers.The function must do the addition of all the numbers in the array and then return the result.

An example:

```php
$total=f(array(3,10,4)); // will return 3+10+4 --> 17
```

Check that the function works with a couple of examples of your own.

```php
<?php
function f($array) {
    $total=0;
    foreach($array as $value) {
        $total+=$value;
    }
    return $total;
}
$array=array(1,20,4);

echo "The total addition is ".f($array)."<br/>";
?>
```

1.5 Exercise 05

Creating a function to sum or multiply an array

Create a function that receives two parameters. The first is an array, and the second is a string. The array must have numbers, and the string can be "+" or "*".

If the string is "+", then the function will return the addition of all the numbers in the array. If the string is "*", then the function will return the multiplication of all the numbers in the array.

A couple of examples:

```php
$total=f(array(3,10,4),"+"); // will return 3+10+4 --> 17
$total=f(array(4,5,2,8),"*") // will return 4*5*2*8 --> 320
```

Check that the function works with different arrays of your own.

```php
<?php
function f($array) {
    $total=0;
    foreach($array as $value) {
        $total+=$value;
    }
    return $total;
}
$array=array(1,20,4);

echo "The total addition is ".f($array)."<br/>";
?>
```

1.6 Exercise 06

Validating a DNI

Create a function that receives a string as a parameter. The string should be a valid DNI without the final char. The function must return the char that fits the DNI, using this formula:

dni % 23 —> a result between 0 and 22 With the result we will get the letter from this table

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
T	R	W	A	G	M	Y	F	P	D	X	B	N	J	Z	S	Q	V	H	L	C	K	E

If the DNI is not valid, then the function will return an error message like “DNI not valid”.

Call the function using your own DNI (remember,without the final char because the function must calculate it)and check the result.

```php
<?php
function validateDni($dni) {
    $array=array("T","R","W","A","G","M","Y","F","P",
                 "D","X","B","N","J","Z","S","Q","V",
                 "H","L","C","K","E");
    if(!is_numeric($dni)) {
        return "DNI not numeric";
    }
    $dni=(int)$dni;
    if($dni > 99999999) {
        return "DNI too big";
    }
    $value=(int)$dni%23;
    return $array[$value];
}
$dni=12234521;
echo "Your letter is ".validateDni($dni);
```