Assessment Description

You are to produce a single PHP file named functions.php which contains 10 user defined PHP functions as detailed below. Code must be commented to indicate the use of each function and to show your understanding of the code. You must use the provided name for each function. You may include additional functions to improve code structure if you wish.

**Function 1**

Name: check\_ac\_email  
Description: Checks an email address ensure the address is correctly formatted and ends with .ac.uk  
Parameter: The e-mail address to be tested  
Return: true if the e-mail address is a valid.ac.uk address otherwise false

**Function 2**

Name: list\_dirs  
Description: Lists all directories in a given path.  
Parameter: The path to the directory to be checked to be checked  
Return: An array containing all the folders in the given path or false in no folders found. The current directory (.) and the parent directory (..) should not be returned in the array.

**Function 3**

Name: count\_files\_of\_type  
Description: Counts all the files in a given path with a particular file extension. If no file extension is given the default value of txt is used  
Parameter 1: The path to the directory to be searched  
Parameter 2: The file extension to be searched for. Note file extensions should be provided without trailing dots eg jpeg not .jpeg  
Return: The number of files found in the path with the required file extension.

**Function 4**

Name: rand\_array  
Description: Returns an array of random number between two limits  
Parameter 1: The number of random numbers to return  
Parameter 2: The lower limit of the random numbers to pick  
Parameter 3: The upper limit of the random numbers to pick  
Return: The array of random numbers

**Function 5**

Name: num\_match\_array  
Description: Fids the number of entries in two arrays that match  
Parameter 1: An array of values  
Parameter 2: Another array of values  
Return: The number of entries that appear in both the arrays

**Function 6**

Name: rand\_password  
Description: Return a random password of a given length. The password must contain at least one capital letter and at least one symbol  
Parameter: The length of the password  
Return: The generated password

**Function 7**

Name: clean\_text  
Description: Removes all spaces and capital letters from a string  
Parameter: The input string containing spaces and capital letters  
Return: A string containing no spaces or capital letters

**Function 8**

Name: times\_table  
Description: Creates a multiplication table from 1 to the given number. The result should be echoed in an HTML table with classes assigned to odd and even columns odd and even rows and a class for the table element.  
Parameter: The upper limit of the table (integer)  
Return: void

 NOTE: If a parameter of 3 is passed to the above function the output should appear as follows

<table class=”mult\_table”>  
<tr class=”odd\_row”>  
       <td class=”odd\_col”>1</td>  
       <td class=”even\_col”>2</td>  
       <td class=”odd\_col”>3</td>  
</tr>  
  
<tr class=”even\_row”>  
       <td class=”odd\_col”>2</td>  
       <td class=”even\_col”>4</td>  
       <td class=”odd\_col”>6</td>  
</tr>  
  
<tr class=”odd\_row”>  
       <td class=”odd\_col”>3</td>  
       <td class=”even\_col”>6</td>  
       <td class=”odd\_col”>9</td>  
</tr>  
</table>

**Function 9**

Name: rgb\_to\_cmyk  
Description: Converts a hexadecimal colour in #xxxxxx format to an array containing CMYK values  
Parameter: The RGB colour in #xxxxxx format  
Return: An array containing four floating point values for Cyan, Magenta, Yellow and Black (Key) respectively.

NOTE: Conversion details can be found at [http://www.rapidtables.com/convert/color/rgb-to-cmyk.htm (Links to an external site.)Links to an external site.](http://www.rapidtables.com/convert/color/rgb-to-cmyk.htm)

**Function 10**

Name: anagram  
Description: Checks if one string is an anagram of another  
Parameter 1: String to check  
Parameter 2: Second string to check  
Return: true if the two strings are anagrams and false if they are not.

Intended learning outcomes for the assessment

This element of the assignment will covers the skills associated with analysing the elements of a cloud based web site and the design of elements within the site. You will apply key coding design concepts and undertake planning and implementation tasks.

Other skills and competencies

This element of the assignment will introduce best practice in PHP coding and develop practical coding experience.

Marking scheme, criteria or rubric

|  |  |
| --- | --- |
|  | Each function is allocated 10% of the marks for this element. Of this 10% the marks should be divided as follows |
| Functionality 40% | Does each function perform as expected? Are the correct return values generated as required? |
| PHP code 20% | Is the PHP code well formed? Is a suitable algorithm used for each function without the use of unnecessary code |
| Comments 40% | Have the functions been commented in a meaningful way? Does each function have a header comment listing a short description of the function, each parameter and the range of possible return values? Does the function contain additional code to indicate the steps implemented in the function. |

### Links to Supporting Materials

The initial files for this element are available in the week 1 section of this site and can be downloaded [here](https://canvas.anglia.ac.uk/courses/563/files/118870/download?verifier=rqmhLMugij9kj8sb1WH4jSOzx1P9KpYV7ti8TTj3).

### Assessment Literacy

The assessment is fine graded as indicated in the marking scheme above. The assessment is similar to and builds on the assignment for the first year Web Development module and students are encouraged to revisit the feedback they received for the Web Development module.