# Homework: PHP Flow Control

This document defines the homework assignments from the [“PHP Basics“ Course @ Software University](https://softuni.bg/trainings/coursesinstances/details/5). Please submit as homework a single zip / rar / 7z archive holding the solutions (source code) of all below described problems.

## Square Root Sum

Write a PHP script **SquareRootSum.php** that displays a **table** in your browser with **2 columns**. The first column should contain a **number** (even numbers from 0 to 100) and the second column should contain **the square root of that number**, rounded to the **second digit after the decimal point**. The **last row** of the table should contain the **sum** of all values in the **Square** column. *Styling the page is optional.* Partial output comes below:

|  |
| --- |
| **Output** |
|  |

## Rich People’s Problems

You are a very rich billionaire with an unhidden passion for cars. You like certain car manufacturers but you don’t really care about anything else, and that’s why you need your own randomizing algorithm that helps you decide how many and what color cars you should buy. Write a PHP script **CarRandomizer.php** that receives **a string of cars** from an **input HTML form**, separated by a comma and space (“**,** “). It then prints **each car**, a random **color** and a random **quantity** in a **table** like the one shown below. Use colors by your choice. Use as quantity a random number in range [1...5]. Styling the page is *optional.* Examples:

|  |  |
| --- | --- |
| **Input** | **Output** |
| “Mitsubishi, Maseratti, Maybach” | C:\Users\Mitko\AppData\Local\Microsoft\Windows\INetCache\Content.Word\02_output1.png |

## Show Annual Expenses

Write a PHP script **AnnualExpenses.php** that receives **n** years from an **input HTML form** and creates a **table** (like the one shown below) with **random expenses** by **months** and the corresponding **years** (n years back). For example, if N is 10, create a table that shows the expenses for each month for the last 10 years. Add a "**Total**"column at the end, showing the total expenses for the same year. The random expenses in the table should be in the range [0…999]. Styling the page is *optional.* Examples:

|  |  |
| --- | --- |
| **Input** |  |
| **Output** | C:\Users\Mitko\AppData\Local\Microsoft\Windows\INetCache\Content.Word\03_output1.png |

## Find Primes in Range

Write a PHP script **PrimesInRange.php** that receives **two numbers** – **start** and **end** –from an **input field** and displays all numbers in that range as a **comma-separated list**. **Prime** numbers should be **bolded**. Styling the page is *optional.* Examples:

|  |  |
| --- | --- |
| **Input / Output** |  |
| **Input / Output** |  |

## Sum of Digits

Write a PHP script **SumOfDigits.php** which receives a **comma-separated list** **of integers** from an **input form** and creates a **two-column table**. The first column should contain **each of the values** from the input. The second column should contain the **sum of the digits of each value**. If the value is not an integer number, print "**I cannot sum that**". Styling the pageis *optional.* Example:

|  |  |
| --- | --- |
| **Input / Output** |  |

## Modify String

Write a PHP script **StringModifier.php** which receives a **string** from an **input form** and **modifies it** according to the selected option (radio button). You should support the following operations: **palindrome check**, **reverse string**, **split** to extract leters only, **hash** the string with the default PHP hashing algorithm, **shuffle** the string characters randomly. The result should be displayed right under the input field.Styling the page is *optional.* Think about which of the modification can be achieved with already built-in functions in PHP. Where necessary, write your own algorithms to modify the given string. Hint**:** Use the **crypt()** function for the "Hash String" modification. Examples:

|  |  |
| --- | --- |
| **Input / Output** |  |
| **Input / Output** | **C:\Users\Mitko\AppData\Local\Microsoft\Windows\INetCache\Content.Word\6-output-reverse.png** |
| **Input / Output** |  |
| **Input / Output** | **C:\Users\Mitko\AppData\Local\Microsoft\Windows\INetCache\Content.Word\6-output-hash.png** |
| **Input / Output** | **C:\Users\Mitko\AppData\Local\Microsoft\Windows\INetCache\Content.Word\6-output-shuffle.png** |

## \*\* Student Sorting

Write a PHP program **StudentSorting.php** that receives data about several students from an **input form** (**first name**, **last name**, **email** and **grade**) and prints it as an HTML table. The user should be able to dynamically **add**/**remove** entries via the **+**/**-** buttons. The data can be sorted by 4 criteria: **First name**, **Last name**, **Email** and **Exam score**. The sorting can be done in **ascending**/**descending** order. The result should be printed as a table. The average exam score should be printed on the last row. (See the example below.) Styling the page is *optional*. Semantic HTML is required. (Hint: Use objects to store the data.)

|  |  |
| --- | --- |
| **Form** | |
|  | |
| **Result** | |
|  | |