# Exercises: PHP Syntax, Basic Web

Problems for exercises and homework for the [“Software Technologies” course @ SoftUni](https://softuni.bg/courses/software-technologies).

You can submit your solutions here <https://judge.softuni.bg/Contests/236/PHP-First-Steps-Exercises>.

# Part I: Simple operations with PHP

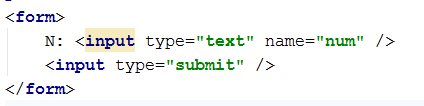
## Multiply a Number by 2

You are given a number num. Write a PHP script that **multiplies** the **number by 2** and prints the result. The input comes as a parameter named num.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 2 | 4 |  | 3 | 6 |

You will be given a HTML form that submits the input data:

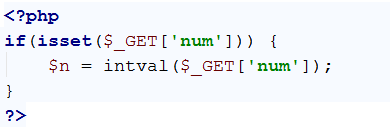


The name of the input parameter will be exactly as shown above – num.

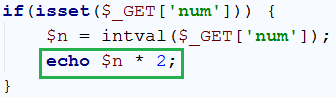
Print the output in the HTML document, just after the HTML form.

### Hints

* In case the form was submitted and the input parameter num exists, take its value as integer using the function intval(string).



* Then, just print the results: echo $n \* 2.



* Test whether your code workd as expected:







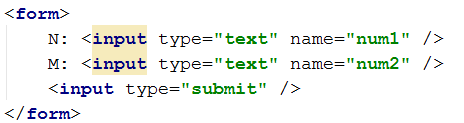
## Multiply Two Numbers

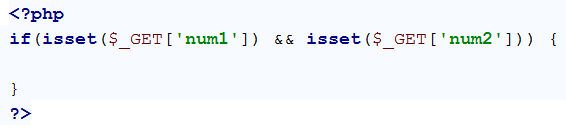
You are given a number num1 and a number num2. Write a PHP script that multiplies num1 \* num2 and prints the result. The input comes as parameters named num1 and num2. Print the output in the HTML page.

### Examples

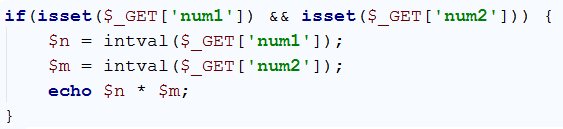
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameters names** | **Input** | **Output** |  | **Input** | **Output** |
| num1 | 2 | 6 | 13 | 169 |
| num2 | 3 | 13 |

### Hints

* This time the form which will be given to use will have 2 input elements, with names num1 and num2
* We must check both elements, if they have values before we perform any action



* When we have checked both elements we get them both and extract their values into variables and we perform the specified action:



* The result is as follows:





## Multiply / Divide Numbers

You are given a number num1 and a number num2. Write a PHP script that:

* Multiplies num1 \* num2 if num2 is greater than or equal to num1.
* Divides num1 / num2 if num1 is greater than num2.

The input comes as parameters named num1 and num2. Print the output in the HTML page.

### Examples

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Parameters names** | **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| num1 | 2 | 6 |  | 13 | 169 | 3 | 1.5 |
| num2 | 3 |  | 13 | 2 |

## Product of 3 Numbers

You are given a number num1, num2 and num3. Write a PHP script that finds if num1 \* num2 \* num3 (the product) is **negative** or **positive**. Try to do this **WITHOUT** multiplying the 3 numbers.

The input comes as parameters named num1, num2 and num3.

### Examples

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Parameters name** | **Input** | **Output** |  | **Input** | **Output** |  | **Input** | **Output** |
| num1 | 2 | Negative |  | 5 | Positive | -3 | Positive |
| num2 | 3 |  | 4 | -4 |
| num3 | -1 |  | 3 | 5 |

### Hints

* Count the **negative numbers**. If they are odd number, the result will be negative, otherwise 🡪 positive.
* Special case: one of the numbers is 0 🡪 the product is positive.

## Numbers from 1 to N

You are given a number num. Write a PHP script that loops through all of the numbers from 1 to num and prints them. The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 5 | 1  2  3  4  5 |  | 2 | 1  2 |

## Numbers from N to 1

You are given a number num. Write a PHP script that loops through all of the numbers from num to 1 and prints them. The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 5 | 5  4  3  2  1 |  | 2 | 2  1 |

## Even Numbers from 1 to N

You are given a number num. Write a PHP script that loops through all of the numbers from 1 to num and prints only the even ones. The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 5 | 2  4 |  | 2 | 2 |

## Odd Numbers from N to 1

You are given a number num. Write a PHP script that loops through all of the numbers from num to 1 and prints only the odd ones. The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 5 | 5  3  1 |  | 2 | 1 |

## N Factorial

You are given a number num. Write a PHP script that prints **factorial** of num. Factorial is calculated by multiplying all numbers to the given number. Factorial (N) = 1 \* 2 \* 3… \* num. The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 5 | 120 |  | 3 | 6 |

## Not Divisor Numbers

You are given a number num. Write a PHP script that prints all the numbers from num to 1**,** which are not divisors of num**.** The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 10 | 9  8  7  6  4  3 |  | 12 | 11  10  9  8  7  5 |

## Fibonacci Sequence

You are given a number num. Write a PHP script that printsnumnumbers from The Fibonacci sequence, each on a new line, starting from 1. In the Fibonacci sequence, every number is formed from the sum of the previous 2. The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 5 | 1  1  2  3  5 |  | 10 | 1  1  2  3  5  8  13  21  34  55 |

## Tribonacci Sequence

In the **“Tribonacci” sequence**, every number is formed by the **sum of the previous 3**.

You are given a number num. Write a PHP script that printsnumnumbers from the Tribonacci sequence, each on a new line, starting from 1. The input comes as a parameter named num. The value num will always be positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 4 | 1  1  2  4 |  | 8 | 1  1  2  4  7  13  24  44 |

## Prime Numbers from N to 1

You are given a number num. Write a PHP script that printsonly the **prime** numbers from num to 1. A prime number is a number that can be divided only by 1 and itself. **1 is not a prime number.** The input comes as a parameter named num. The parameter num will hold a positive integer.

### Examples

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** |  | **Input** | **Output** |
| num | 10 | 7  5  3 |  | 20 | 19  17  13  11  7  5  3 |

# Part 2: Manipulating HTML with PHP

## HTML Buttons

You are given a number num. Write a PHP script that **generates HTML <button> elements.** The buttons must hold a number from 1 to num as their caption. The input comes as a parameter named num, holding a positive integer.

There is no indentation on the elements.

### Examples

|  |  |  |  |
| --- | --- | --- | --- |
| **Parameter name** | **Input** | **Output** | **Picture** |
| num | 3 | <button>1</button>  <button>2</button>  <button>3</button> |  |
| num | 5 | <button>1</button>  <button>2</button>  <button>3</button>  <button>4</button>  <button>5</button> |
| num | 1 | <button>1</button> |

## Sub-Lists

You are given a number num1 and a number num2. Write a PHP script that generates a list with **num1** elements, and each of those elements has a sub-list with **num2** elements. Each of the Lists has a caption “**List #**” where #is the current number from 1 to num1**.** And each of the elements of those lists has a caption “**Element #.#**” where the first # is the current list number and the second **#** is the current element from 1 to num2. The input comes as a parameters named num1 and num2, holding positive integers.

### Examples

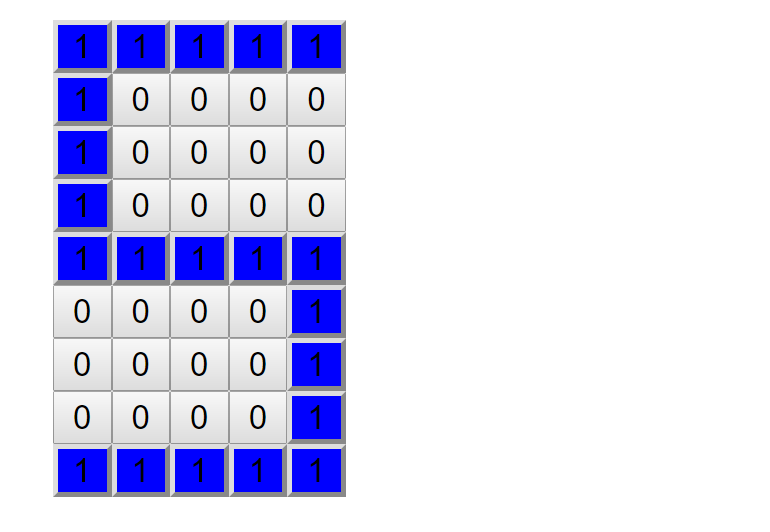
|  |  |  |  |
| --- | --- | --- | --- |
| **Parameters names** | **Input** | **Output** | **Picture** |
| num1 | 1 | <ul>  <li>List 1  <ul>  <li>  Element 1.1  </li>  <li>  Element 1.2  </li>  </ul>  </li>  </ul> |  |
| num2 | 2 |

There is **indentation** on the elements. Each nested element is **tabbed** **once right** from its parent.

## Draw an “S” with Buttons

Write a PHP script that draws 5 x 9 <button> elements with **0** and **1** in them. The **1**s should form a figure **“S”**. Make the **1**s’ background color – **blue**. There is no indentation on the elements.

### Example



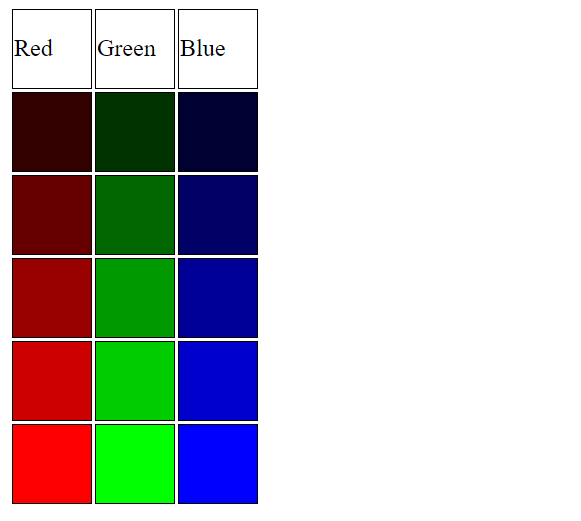
## RGB Table

Write a PHP script that draws a table with 3 columns, with captions – **Red**, **Green** and **Blue**, and 5 rows, each having as **background** **1** of **5** shades of those colors. Increase the color index by **51** for each row. For example, for the Red Column it will be:

* RGB(51, 0, 0)
* RGB(102, 0, 0)
* RGB(153, 0, 0)
* . . .

The caption columns should **NOT** be colored. There indentation on the elements. Make the table cells in the table row **tabbed 1 times to the right**.

### Picture:



## 50 Shades of Grey

Write a PHP script that generates 5 rows of 10 <div> **elements**, each having as background a different shade of grey – resulting in 50 shades of grey. Every row starts with a color index – a **multiple of 51**.

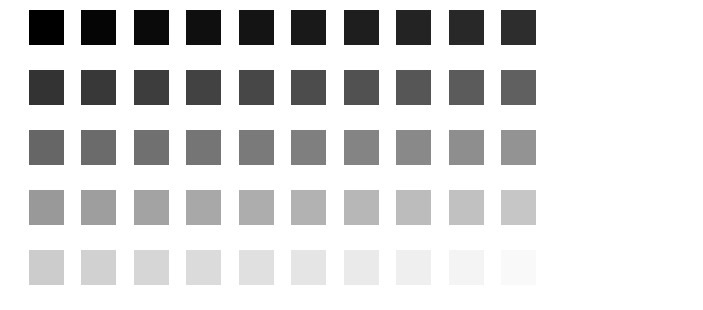
* First row – 0,
* Second row – 51,
* Third row – 102 …

Then you need to print **10 divs**, each increasing that index with **5.**

* First row – 0, 5, 10…
* Second row – 51, 56, 61…

The color index needs to be applied to **all** **parameters** of the **RGB function**. Example: rgb(51, 51, 51);

### Picture:



There is no indentation on the elements.