# Exercises: HTTP and HTML

You can check your solutions here: <https://judge.softuni.bg/Contests/1264/HTTP-and-HTML-Basic-Exercise>.

## Problem 1. Hello, Person!

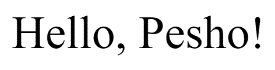
Write a program which receives a **name** from a **form** and prints in the html the **following** **greeting** (after you click on the **[Submit]** button).

* Hello, {name}!

You should read the name from the following form:

|  |
| --- |
| <**form**>  Name: <**input type="text" name="person"** />  <**input type="submit"** /> </**form**> |

### Examples

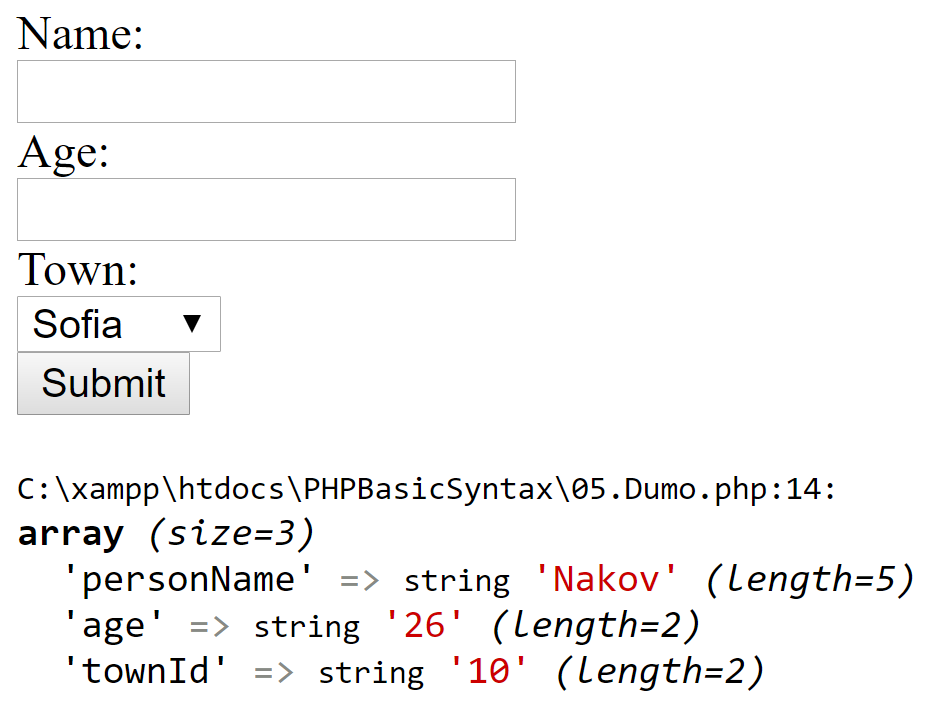
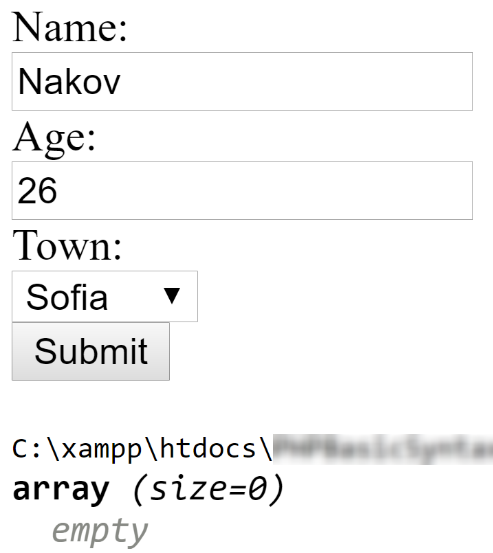


## Problem 2. Dump an HTTP GET Request

Write a PHP script to dump the **content** of the **HTTP GET** request parameters using var\_dump(…). Dump the content of the following form:

|  |
| --- |
| <**form**>  <**div**>Name:</**div**>  <**input type="text" name="personName"**/>  <**div**>Age:</**div**>  <**input type="number" name="age"**/>  <**div**>Town:</**div**>  <**select name="townId"**>  <**option value="10"**>Sofia</**option**>  <**option value="20"**>Varna</**option**>  <**option value="30"**>Plodvid</**option**>  </**select**>  <**div**><**input type="submit"**/></**div**> </**form**> |

### Examples



## Problem 3. Sum Two Numbers

Write a PHP script to sum two numbers. The script should display a HTML form to enter the numbers. If the numbers are passed as parameters, display their sum. You will receive the input from the following form:

|  |
| --- |
| <**form**>  <**div**>First Number:</**div**>  <**input type="number" name="num1"** />  <**div**>Second Number:</**div**>  <**input type="number" name="num2"** />  <**div**><**input type="submit"** /></**div**> </**form**> |

### Examples

### 

## Problem 4. Sort Lines

Write a PHP script that **sorts** the **text** lines from a <textarea>. Take the input from the following form:

|  |
| --- |
| <**form**>  <**textarea rows="10" name="lines"**>**<?=** $sortedLines  **?>**</**textarea**> <**br**>  <**input type="submit" value="Sort"**> </**form**> |

### Examples



## Problem 5. Extract Capital-Case Words

Write a **PHP** script to extract all **capital-case** words from an **array** of **strings**. All **non-letter** chars are considered **separators**. Use the following form:

|  |
| --- |
| <**form**>  <**textarea rows="10" name="text"**></**textarea**><**br**>  <**input type="submit" value="Extract"**> </**form**> |

### Examples

|  |  |
| --- | --- |
| **Input** | **Output** |
| We start by HTML, CSS, JavaScript, JSON and REST.  Later we touch some PHP, MySQL and SQL.  Later we play with C#, EF, SQL Server and ASP.NET MVC.  Finally, we touch some Java, Hibernate and Spring.MVC. | HTML, CSS, JSON, REST, PHP, SQL, C, EF, SQL, ASP, NET, MVC, MVC |

## Problem 6. Word Mapping

Write a PHP program **WordMapper.php** that takes a text from a textarea **with attribute name="input"** and prints each word and the number of times it occurs in the text. The search should be **case-insensitive**. The result should be printed as an **HTML table with border attribute set to 2**. If you get 0/100 in judge, check in details the zero test.

|  |  |
| --- | --- |
| **Input** | **Output** |
| The quick brows fox.!!! jumped over..// the lazy dog.! |  |
|  |

## Problem 7. Coloring Texts

Write a PHP program **TextColorer.php** that takes a text from a textfield **with attribute name=”input”** , colors each character according to its ASCII value and prints the result. If the ASCII value of a character is odd, the **character** should be printed in **blue**. If it’s even, it should be printed in **red**.For the colors **use the <span> tag.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| What a wonderful world! | W h a t a w o n d e r f u l w o r l d ! |
|  |

## Problem 8. Navigation Builder

Write a PHP program **Navigation.php** that takes data from several input fields and builds 3 navigation bars. The input fields are **categories with attribute name=”categories”**, **tags with attribute name=”tags”** and **months with attribute name=”tags”**. The first navigation bar should contain a list of the **categories**, the second navigation bar – a list of the **tags** and the third should contain the **months**. The entries in the input strings will be separated by a **comma** and **space** "**,** ". When you print your result don’t forget to **use** **<h2> tag for “Categories”, “Tags”, and “Months”.** Semantic HTML is required.

|  |  |  |
| --- | --- | --- |
| **Input** | | **Output** |
| categories | Knitting, Cycling, Swimming, Dancing |  |
| tags | person, free time, love, peace, protest |
| months | April, May, June, July |
|  | |

## Problem 9. HTML Table

Write a PHP script **InformationTable.php** which **generates an HTML table** by given information about a person (**name**, **phone number**, **age**, **address**). The result should be printed as an **HTML table with border attribute set to 2 and background color: ‘orange’** Examples:

**name with attribute name=”name”**

**phone number with attribute name=”phone”**

**age with attribute name=”age”**

**address with attribute name=”address”**

|  |  |
| --- | --- |
| **Input** | **Output** |
| Gosho  0882-321-423  24  Hadji Dimitar |  |
| Pesho  0884-888-888  67  Suhata Reka |  |

## Problem 10. Form Data

Write a PHP script **GetFormData.php** that retrieves the **input data from an HTML form**, and displays it as string. The input fields should hold **name**, **age** and **gender** (radio buttons). The returned string should be a message containing the information submitted by the form. 100% accuracy is *NOT* required. Semantic HTML is required.   
Example:

**name with attribute name=”name”**

**age with attribute name=”age”**

**gender with attribute name=”gender”**

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

## Problem 11. \*\* Awesome Calendar

Write a PHP script **AwesomeCalendar.php** which creates a calendar in HTML **for a given year**. Styling the calendar is *optional*. Semantic HTML is required. Hint: Embed HTML in your PHP code. Use *tables* or *divs* for structuring the calendar. Look for a way to find the day of the week.

