Министерство науки и высшего образования Российской Федерации Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имен В.Ф. Уткина»

Кафедра «САПР ВС»

Отчет о лабораторной работе №4

«Программирование на языке PHP»

по дисциплине:

«Веб-программирование»

Выполнили:

Студенты группы 146

Бригада № 2

Соколов Н.А.

Малин И.С.

Проверили:

доц. Сапрыкин А.Н

**Лабораторная работа №1:**

**Тема:** Создание Web-страниц на стороне Web-сервера Apache средствами языка PHP. Использование математических операторов и функций языка PHP.

**Цель работы:** Закрепление теоретических знаний по работе с математическими операторами и функциями языка PHP. Приобретение практических навыков настройки Web-сервера Apache для тестирования Web-страниц.

**Практическая часть**

**Задание по вариантам.**

Создать Web-страницу, предлагающую пользователю угадать задуманное число в заданном диапозоне (от 1 до 1000). Web-страница должна содержать форму с текстовой областью, в которую пользователь должен вводить число, пытаясь угадать правильный ответ. Это число случайным образом рассчитывается по засекреченной формуле, заданной по вариантам. Значение аргумента хранится в программе, либо при каждом новом запуске определяется как случайное на заданном отрезке. Полученное по секретной формуле значение функции необходимо округлить до целого числа и сравнить с числом, введенным пользователем. Если пользователь угадал загаданное число, необходимо вывести соответствующее сообщение, в противном случаевывести правильное число и предложить попытку заново. На Web-странице для вычисления загаданного числа необходимо использовать стандартные операторы присваивания, математические, логические операторы, а также условные и циклические операторы языка PHP.

Засекреченная формула:

*,*

значение аргумента x принадлежит [1,10]

В исходном HTML-коде использовать комментарии каждого тела.

Результат представлен в тексте HTML документа ниже:

<!-- Тип документа -->

<!DOCTYPE html>

<!-- Начало html кода -->

<html>

<!-- Начало тега head -->

<head>

<!-- Название вкладки -->

<title>Угадай число</title>

</head>

<!-- Начало основного тела документа -->

<body>

<!-- Форма для ввода числа -->

<form method="post">

<!-- Подсказка для пользователя -->

<label for="guess">Введите число от 1 до 1000:</label><br>

<!-- Поле ввода текста -->

<input type="text" id="guess" name="guess">

<!-- Кнопка для отправки ответа -->

<input type="submit" value="Проверить">

</form>

<?php

// Задание диапазона загадываемого числа

$min = 1;

$max = 1000;

// Генерация случайного числа с использованием секретной формулы

$x = rand(1, 10);

$secretNumber = sqrt(7 \* $x + cos($x));

$secretNumber = round($secretNumber); // Округление до целого числа

// Проверка, было ли отправлено число пользователем

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

// Получение введенного числа

$userGuess = $\_POST['guess'];

// Проверка числа на соответствие загаданному

if ($userGuess == $secretNumber) {

echo "Поздравляем! Вы угадали число $secretNumber!";

} else {

echo "К сожалению, это не верно. Правильное число: $secretNumber. Попробуйте еще раз.";

}

}

?>

<!-- Конец основного тела документа -->

</body>

<!-- Конец html кода -->

</html>

Результат выполнения кода представлен на рисунке 1.

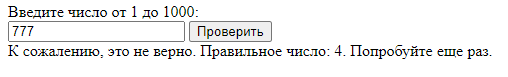


Рисунок 1 – результат выполнения кода для задания 1

**Контрольные вопросы.**

**Вопрос 1.** Дайте определение понятию веб-сервер.

Ответ 1:

Веб-сервером называют совокупность программного обеспечения и сервера, на котором работает данное программное обеспечение (ПО). С точки зрения ПО веб-сервер – это специальная программа, принимающая HTTP-запросы и возвращающая ответ браузеру в виде HTML-разметки.

**Вопрос 2.** Перечислите наиболее популярные веб-серверы и их отличия.

Ответ 2:

Существует большое количество ПО для выполнения функции веб-сервера. Наиболее популярными среди них являются:

* Apache
* Microsoft – IIS
* Nginx

Apache представляет собой ПО с открытым исходным кодом для организации веб-сервера. С его помощью обеспечивается работа около 46 % всех сайтов. Apache HTTP-сервер является кроссплатформенным ПО, поддерживает операционные системы Linux, BSD, Mac OS, Microsoft Windows. Основными достоинствами Apache считаются надёжность и гибкость конфигурации.

Nginx – это легковесный и мощный web-сервер c открытым исходным кодом, который способен держать до 10 тысяч одновременных соединений, расходуя при этом минимальное количество ресурсов. В данный момент Nginx обслуживает около 2 30,8 % всех сайтов. К основной особенности Nginx можно отнести то, что при совмещенной работе двух веб-серверов (Apache + Nginx) производительность Nginx в два раза выше, чем у Apache, а также заметно ниже потребление оперативной памяти.

Microsoft – IIS – это веб-сервер, разработанный компанией Microsoft для своих операционных систем. Продукт полностью проприетарный и идет в комплекте с Windows. IIS поддерживает технологию .NET.

**Вопрос 3.** Как добавить PHP-код на веб-страницу?

Ответ 3:

Программа или скрипт на PHP, как правило, находится в файле расширением .php. Хотя также допустимо вставлять код php и в файлы с расширениями .html. Документ PHP может содержать как разметку html, так и код на языке php. Для перехода от разметки html к коду php используются теги <?php и ?> между которыми идет код php. Данные теги служат указанием интерпретатору PHP, что их содержимое надо интерпретировать как код php, а не разметку html.

**Вопрос 4.** Какие символы может включать в себя название переменной?

Ответ 4:

При наименовании переменных нам надо учитывать следующие правила:

* + - имена переменных должны начинаться с алфавитного символа или с подчеркивания;
* имена переменных могут содержать только символы: a–z, A–Z, 0–9 и знак подчеркивания;
  + - имена переменных не должны включать в себя пробелы.

**Вопрос 5.** Как осуществляется проверка инициализации переменной?

Ответ 5:

Функция isset() позволяет определить, инициализирована переменная или нет. Если переменная определена, то isset() возвращает значение true. Если переменная не определена, то isset() возвращает false. Также, если переменная имеет значение null, функция isset()возвращает false.

**Вопрос 6.** Опишите способы задания констант.

Ответ 6:

Для определения константы применяется оператор const, при этом в названии константы знак доллара $ (в отличие от переменных) не используется. Для обозначения констант принято использовать заглавные символы.

<?php const PI = 3.14; echo PI; ?>

Также для определения константы может применяться функция define(), которая имеет следующую форму:

define(string $name, string $value);

Параметр $name передает название константы, а параметр $value – ее значение. Значение константы может представлять тип int,float, string, bool, null или массивы.

<?php define("NUMBER", 55); echo NUMBER; // 55 ?>

**Вопрос 7.** Для чего предназначены «магические» константы в PHP?

Ответ 7:

Кроме создаваемых пользователем констант в PHP имеется еще

несколько так называемых «магических» констант, которые есть в

языке по умолчанию:

* \_\_FILE\_\_ – хранит полный путь и имя текущего файла;
* \_\_LINE\_\_ – хранит текущий номер строки, которую
* обрабатывает интерпретатор;
* \_\_DIR\_\_ – хранит каталог текущего файла;
* \_\_FUNCTION\_\_ – название обрабатываемой функции;
* \_\_CLASS\_\_ – название текущего класса;
* \_\_METHOD\_\_ – название обрабатываемого метода.

**Вопрос 8.** Назовите базовые типы данных.

Ответ 8:

В PHP есть десять базовых типов данных:

* bool – логический тип;
* int – целые числа;
* float – дробные числа;
* string – строки;
* array – массивы;
* object – объекты;
* callable – функции;
* mixed – любой тип;
* resource – ресурсы;
* null – отсутствие значения.

**Вопрос 9.** Опишите определение оператора множественного выбора.

Ответ 9:

Конструкция switch..case является альтернативой

использованию конструкции if..elseif..else. Оператор switch

получает некоторое выражение и сравнивает его с набором значений.

switch (выражение){

case значение1: действия; break;

case значение2: действия; break;

//...

case значениеN: действия; break;

default: действия; break;

}

**Вопрос 10.** Перечислите виды циклов и их синтаксис.

Ответ 10:

В PHP имеются следующие виды циклов:

* for ;
* while ;
* do..while ;
* foreach .

Цикл for имеет следующее формальное определение:

for ([инициализация счетчика]; [условие];

[изменение счетчика])

{

// действия

}

Цикл while проверяет истинность некоторого условия и, если

условие истинно, то выполняется блок выражений цикла.

$counter = 1;

while($counter<10)

{

echo $counter \* $counter . "<br />";

$counter++;

}

Цикл do..while похож на цикл while, только теперь

выполняется блок цикла, и только потом выполняется проверка

условия. То есть даже, если условие ложно, то блок цикла выполнится

как минимум один раз.

$counter = 1;

do

{

echo $counter \* $counter . "<br />";

$counter++;

}

while($counter<10)

**Заключение:** В ходе выполнения лабораторной работы было получено закрепление теоретических знаний по работе с математическими операторами и функциями языка PHP. Приобретение практических навыков настройки Web-сервера Apache для тестирования Web-страниц.

**Часть 2:**

**Тема:** Использование языка написания сценариев PHP для работы со строковыми функциями.

**Цель работы:** Закрепление теоретических знаний и приобретение практических навыков работы со строками языка PHP.

**Практическая часть**

**Задание по вариантам.**

Создать Web-страницу для обработки введенной информации

пользователя с помощью строковых функций языка PHP. На странице должно быть представлено текстовое поле, в которое пользователь должен ввести любую строку, например ФИО. Программа должна реализовать работу с введенной строкой пользователя. Найти сколько в заданной определенных букв, на какой они позиции, определить длину строки, заменить в исходной строке одни подстроки на другие, вывести инвертированную строку, разбить строку на подстроки.

В исходном HTML-коде использовать комментарии каждого тега.

Результат представлен в тексте HTML документа ниже:

<!-- Тип документа -->

<!DOCTYPE html>

<!-- Начало html кода -->

<html>

<!-- Начало тега head -->

<head>

<!-- Название вкладки -->

<title>Обработка строк</title>

</head>

<!-- Начало основного тела документа -->

<body>

<!-- Форма для ввода строки -->

<form method="post">

<!-- Подсказка для пользователя -->

<label for="userInput">Введите строку:</label><br>

<!-- Текстовое поле для ввода текста -->

<input type="text" id="userInput" name="userInput">

<!-- Кнопка отправки запроса -->

<input type="submit" value="Обработать">

</form>

<?php

// Проверка, была ли отправлена строка пользователем

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

// Получение введенной строки

$userString = $\_POST['userInput'];

// Вывод информации о исходной строке

echo "<p>Исходная строка: $userString</p>";

// Конкатенация строк

echo "<p>Длина строки: " . strlen($userString) . "</p>";

// Найти сколько и на какой позиции определенных букв

$searchLetters = ['а', 'б', 'в', 'г','э','ю', 'я'];

foreach ($searchLetters as $letter) {

// Подсчитать количество каждой из букв в строке

$count = substr\_count(mb\_strtolower($userString, 'utf-8'), $letter);

echo "<p>Буква '$letter' встречается $count раз(а)</p>";

// Найти позиции буквы в строке

$positions = [];

$pos = mb\_strpos(mb\_strtolower($userString, 'utf-8'), $letter);

while ($pos !== false) {

$positions[] = $pos + 1;

$pos = mb\_strpos(mb\_strtolower($userString, 'utf-8'), $letter, $pos + 1);

}

echo "<p>Позиции буквы '$letter': " . implode(', ', $positions) . "</p>";

}

// Заменить подстроки в строке

$userString = str\_replace('и', 'e', $userString);

echo "<p>Строка после замены 'и' на 'e': $userString</p>";

// Вывести инвертированную строку

$length = mb\_strlen($userString);

$invertedString = '';

// Из-за особенностей языка при работе с кириллицей,

// реализуем инверсию как взятие на каждой итерации цикла подстроки длиной 1

while ($length > 0) {

$length--;

$invertedString .= mb\_substr($userString, $length, 1, 'UTF-8');

}

echo "<p>Инвертированная строка: $invertedString</p>";

// Разбить строку на подстроки (аналог split в JavaScript)

$substrings = explode(' ', $userString);

echo "<p>Подстроки после разбиения по пробелу: " . implode(', ', $substrings) . "</p>";

}

?>

<!-- Конец тега body -->

</body>

<!-- Конец тега html -->

</html>

Результат выполнения скрипта и итоговый вид html страницы представлен на рисунке 2.

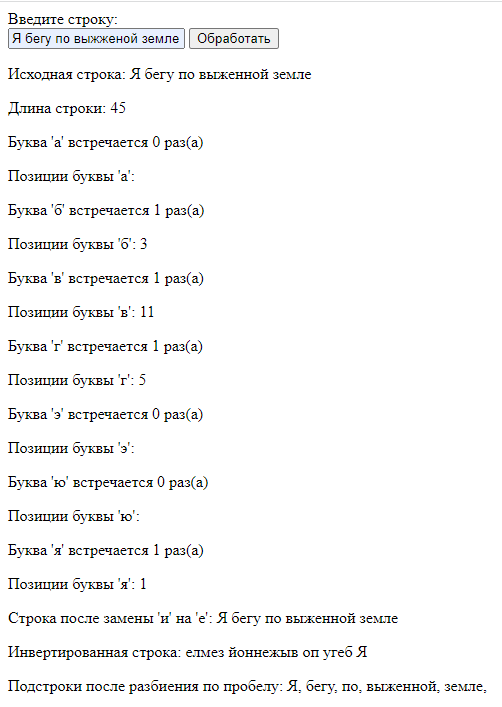


Рисунок 2 – результат скрипта задания 2 в браузере

**Заключение:** В ходе выполнения лабораторной работы мы закрепили теоретических знания и приобрели практические навыкы работы со строками языка PHP.

**Часть 3:**

**Тема:** Использование языка написания сценариев PHP для работы с многомерными и ассоциативными массивами. Использование функций сортировки массивов.

**Цель:** Закрепление теоретических знаний и приобретение практических навыков работы с массивами средствами языка PHP.

**Практическая часть**

**Задание по вариантам.**

Создать Web-сайт "Справочник городов", осуществляющий поиск информации о заданном городе страны "Турция". На сайте должно быть представлено текстовое поле, в которое пользователь должен ввести первую букву названия города. Программа должна реализовать поиск требуемого города и предоставить пользователю всю необходимую информацию об этом городе. Для этого необходимо создать ассоциированный массив, содержащий названия всех городов заданной страны (50 городов).На сайте необходимо реализовать работу с многомерными массивами для хранения информации о каждом городе. Помимо этого, в программе необходимо вывести содержание всего массива городов, отсортированного в определенном порядке. Результат представлен в тексте HTML документа ниже:

<!DOCTYPE html>

<html>

<head>

<title>Справочник городов</title>

</head>

<body>

<!-- Форма для ввода первой буквы города -->

<form method="post">

<label for="firstLetter">Введите первую букву города (u для вывода городов по возрастанию, l - по убыванию):</label><br>

<input type="text" id="firstLetter" name="firstLetter">

<input type="submit" value="Найти">

</form>

<?php

$citiesTurkey = array(

'Агачёрен' => array(

'Население' => 3047,

'Высота центра' => 1169,

),

'Агры' => array(

'Население' => 97839,

'Высота центра' => 1632,

),

'Адана' => array(

'Население' => 13844,

'Высота центра' => 23,

),

'Адапазары' => array(

'Население' => 377683,

'Высота центра' => 31,

),

'Аджипаям' => array(

'Население' => 12588,

'Высота центра' => 953,

),

'Аланья' => array(

'Население' => 364180,

'Высота центра' => 27,

),

'Ахлат' => array(

'Население' => 19951,

'Высота центра' => 1650,

),

'Анкара' => array(

'Население' => 5445026,

'Высота центра' => 938,

),

'Аксарай' => array(

'Население' => 129949,

'Высота центра' => 980,

),

'Бабаэски' => array(

'Население' => 28200,

'Высота центра' => 55,

),

'Байбурт' => array(

'Население' => 32782,

'Высота центра' => 1550,

),

'Балыкесир' => array(

'Население' => 32782,

'Высота центра' => 1550,

),

'Бартын' => array(

'Население' => 56557,

'Высота центра' => 25,

),

'Бахче' => array(

'Население' => 12841,

'Высота центра' => 650,

),

'Башкале' => array(

'Население' => 14601,

'Высота центра' => 21,

),

'Бурса' => array(

'Население' => 2994521,

'Высота центра' => 100,

),

'Бюньян' => array(

'Население' => 12505,

'Высота центра' => 1337,

),

'Вакфыкебир' => array(

'Население' => 14157,

'Высота центра' => 68,

),

'Газиантеп' => array(

'Население' => 1680723,

'Высота центра' => 850,

),

'Гекчебей' => array(

'Население' => 7273,

'Высота центра' => 61,

),

'Гелиболу' => array(

'Население' => 27160,

'Высота центра' => 0,

),

'Герменджик' => array(

'Население' => 12635,

'Высота центра' => 65,

),

'Гиресун' => array(

'Население' => 135920,

'Высота центра' => 10,

),

'Гюлагач' => array(

'Население' => 4565,

'Высота центра' => 1162,

),

'Гюлюч' => array(

'Население' => 7249,

'Высота центра' => 97,

),

'Гюрюн' => array(

'Население' => 18009,

'Высота центра' => 1334,

),

'Даламан' => array(

'Население' => 37364,

'Высота центра' => 8,

),

'Дальян' => array(

'Население' => 4848,

'Высота центра' => 3,

),

'Даренде' => array(

'Население' => 8645,

'Высота центра' => 1006,

),

'Демре' => array(

'Население' => 27691,

'Высота центра' => 10,

),

'Денизли' => array(

'Население' => 544424,

'Высота центра' => 354,

),

'Дикили' => array(

'Население' => 15530,

'Высота центра' => 201,

),

'Дюздже' => array(

'Население' => 240633,

'Высота центра' => 160,

),

// 35

'Зара' => array(

'Население' => 11588,

'Высота центра' => 1350,

),

'Измир' => array(

'Население' => 2948160,

'Высота центра' => 25,

),

'Измит' => array(

'Население' => 302960,

'Высота центра' => 100,

),

'Имамоглу' => array(

'Население' => 21331,

'Высота центра' => 73,

),

'Искендерун' => array(

'Население' => 184833,

'Высота центра' => 8,

),

// 40

'Йозгат' => array(

'Население' => 432560,

'Высота центра' => 1335,

),

'Кадынханы' => array(

'Население' => 33036,

'Высота центра' => 1127,

),

'Кандыра' => array(

'Население' => 13511,

'Высота центра' => 18,

),

'Карабюк' => array(

'Население' => 108167,

'Высота центра' => 354,

),

'Караман' => array(

'Население' => 152450,

'Высота центра' => 1039,

),

// 45

'Каш' => array(

'Население' => 62866,

'Высота центра' => 100,

),

'Кастамону' => array(

'Население' => 82289,

'Высота центра' => 904,

),

'Кемер' => array(

'Население' => 49383,

'Высота центра' => 4,

),

'Коркутели' => array(

'Население' => 56285,

'Высота центра' => 1020,

),

'Ладик' => array(

'Население' => 8626,

'Высота центра' => 913,

),

// 50

'Мармарис' => array(

'Население' => 95851,

'Высота центра' => 7,

),

'Аванос' => array(

'Население' => 13210,

'Высота центра' => 935,

)

);

function printInfo($associativeArray) {

foreach ($associativeArray as $key => $value) {

echo "<p style='padding: 0px 0px 0px 10px;margin: 0 0 0 0;'> $key = $value </p>";

}

}

function printCities($cities, $firstLetter) {

foreach ($cities as $key => $value) {

$keyFirstLetter = mb\_substr($key, 0, 1, 'UTF-8');

if (mb\_strtoupper($keyFirstLetter) === mb\_strtoupper($firstLetter) || ($firstLetter == 'U') || ($firstLetter == 'L')) {

echo "Город: $key, Информация: <br>";

echo printInfo($value);

echo "<br>";

}

}

}

$firstLetter = mb\_strtoupper($\_POST['firstLetter'] ?? ''); // Use the null coalescing operator to handle the case when $\_POST['firstLetter'] is not set

// Проверка, была ли отправлена первая буква города

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

if (($firstLetter != 'U') && ($firstLetter != 'L')) {

printCities($citiesTurkey, $firstLetter);

}

if (mb\_strtoupper($firstLetter) == 'U') {

// Sort the array in ascending order and then print cities

ksort($citiesTurkey, SORT\_STRING);

printCities($citiesTurkey, 'U');

}

if (mb\_strtoupper($firstLetter) == 'L') {

// Sort the array in descending order and then print cities

krsort($citiesTurkey, SORT\_STRING);

printCities($citiesTurkey, 'L');

}

}

?>

</body>

</html>

Результат отработки различных сценариев представлен на рисунках 3-5.

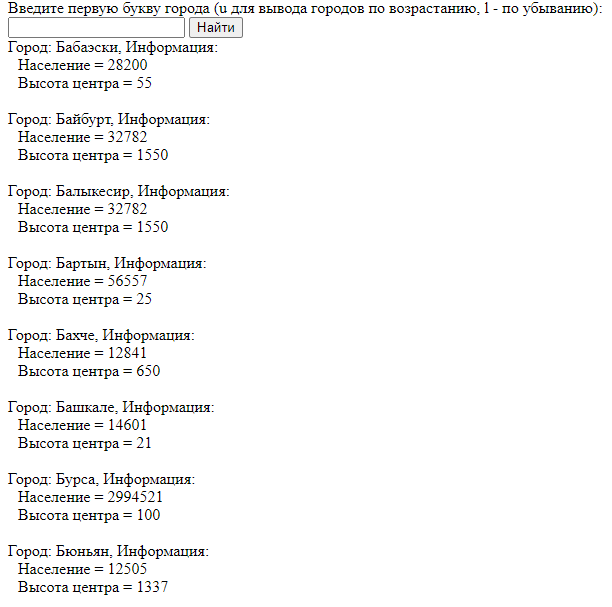


Рисунок 3 – Вывод турецких городов на букву «Б»

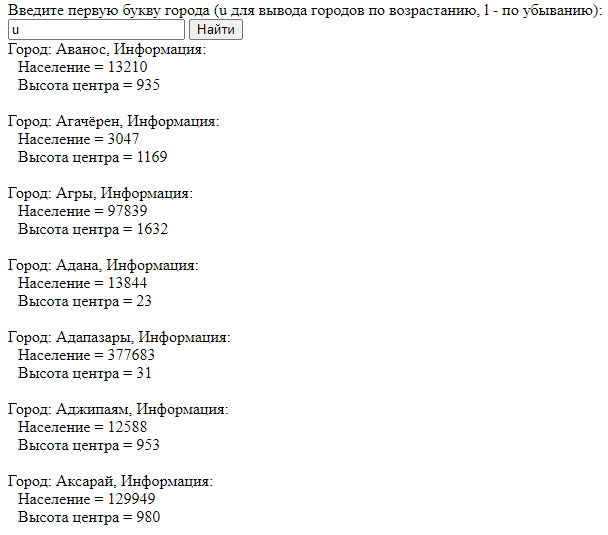


Рисунок 4 – Начало вывода турецких городов по возрастанию их названий

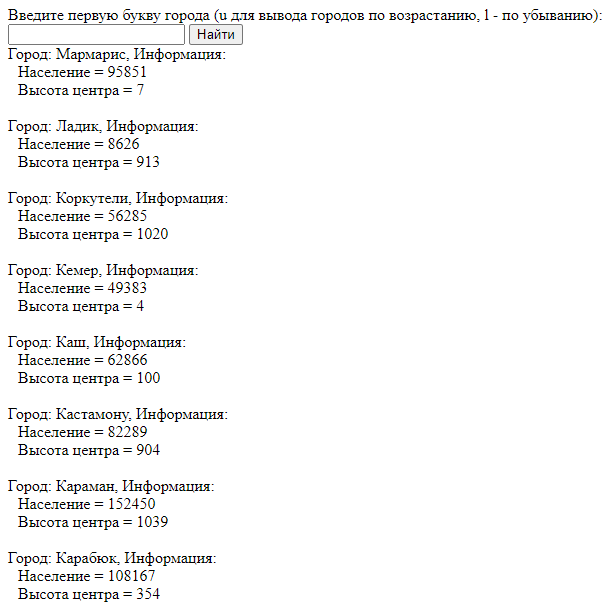


Рисунок 5 – Начало вывода турецких городов по убыванию их названий

**Заключение:** В ходе выполнения лабораторной работы были закреплены теоретические знание и приобретены практические навыки работы с массивами средствами языка PHP.

**Часть 4:**

**Тема:** Использование возможностей работы с файлами средствами языка написания сценариев PHP для обработки данных пользователя при заказе товаров в «Интернет-магазине». Использование стандартных операторов языка PHP.

**Цель:** Закрепление теоретических знаний и приобретение практических навыков работы с данными из файлов средствами языка PHP при обработке данных пользователя из форм.

**Практическая часть**

**Задание по вариантам.**

Создать веб-сайт "Интернет-магазин". На странице необходимо создать элемент дизайна формы для заказа товара в "Интернет-магазине". Реализовать средствами языка PHP обработку заказов и поиска товаров по каталогу. Для этого необходимо создать скрипт, который считывает информацию из формы для обработки. По результатам заказа на странице необходимо вывести содержимое корзины пользователя с рассчитанной суммой заказа. При этом на странице необходимо осуществить проверку введенных пользователем данных. При обработке заказа следует использовать математические функции, а также различные условные, логические и циклические операторы языка PHP. Например, при начислении скидки или оформлении доставки товара на дом. Помимо этого, на странице необходимо реализовать сохранение данных пользователя в текстовый файл. Организовать работу с открытием, просмотром и записью данных в файл. Также необходимо реализовать вывод содержимого файла на экран по определенному критерию заказчика. Задание:

**«Книжный магазин учебной литературы (заказ учебников по определенной специальности)»**

Результат выполнения данных заданий представлен в HTML документах ниже:

<РЕЗУЛЬТАТ>

Результаты работы страницы при различных сценариях представлены на рисунках 5-

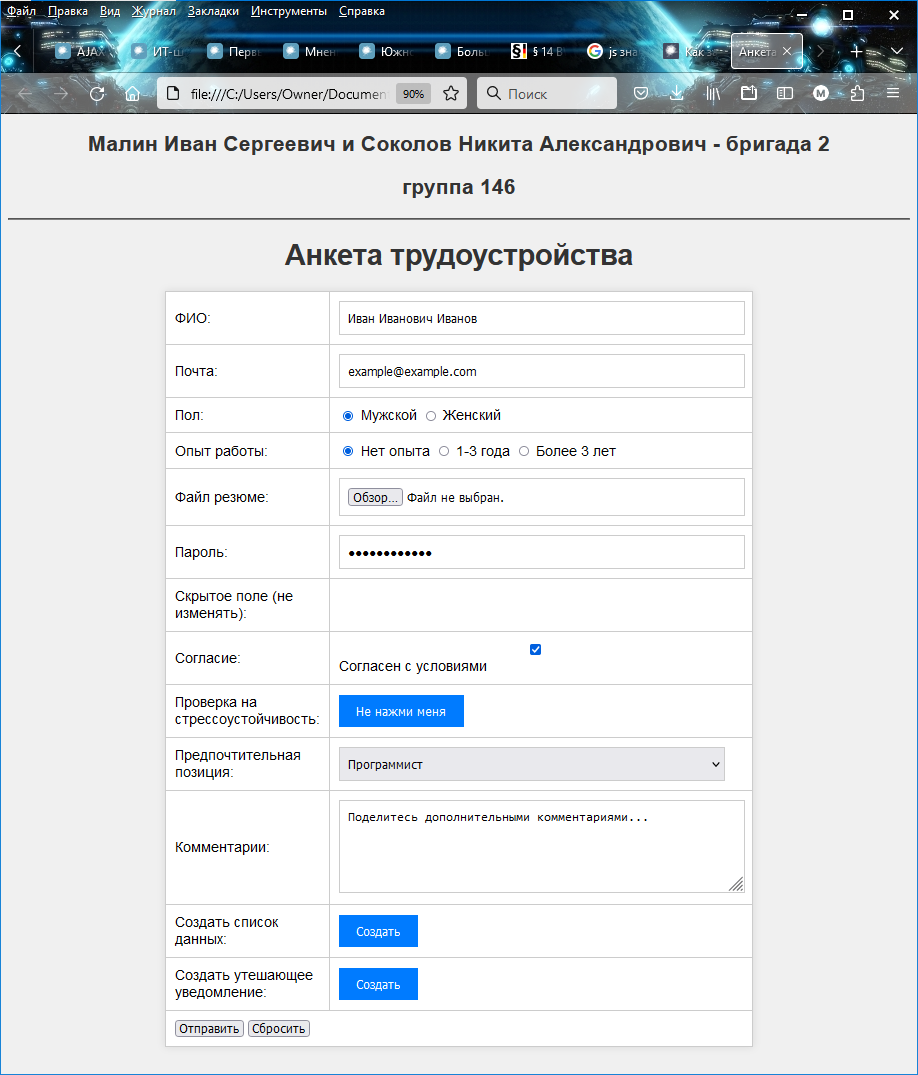


Рисунок 5 – анкета с новыми элементами взаимодействия

**Заключение:** В ходе выполнения лабораторной работы были закреплены теоретические знания и приобретены практические навыки работы с данными из файлов средствами языка PHP при обработке данных пользователя из форм.

**Часть 5:**

**Тема:** Создание базы данных в СУБД MySQL. Работа с базой данных средствами инструмента phpMyAdmin и языка PHP.

**Цель:** Закрепление теоретических знаний по созданию базы данных в СУБД MySQL с помощью программы phpMyAdmin. Приобретение практических навыков работы с данными средствами языка PHP

**Практическая часть**

**Задание по вариантам.**

Создать базу данных в СУБД MySQL, состоящую из трех таблиц с использованием инструмента phpMyAdmin. Реализовать подключение к базе данных средствами языка PHP. Для этого необходимо создать веб-интерфейс, с помощью которого организовать добавление, редактирование данных по различным критериям, изменение данных в базе данных, извлечение данных из таблицы и вывод данных на экран средствами языка PHP. Задание:

**«Аптека (учет лекарств в аптеке)»**

Результат выполнения данных заданий представлен в HTML документах ниже:

<РЕЗУЛЬТАТ>

Результаты работы страницы при различных сценариях представлены на рисунках 5-

**Заключение:** В ходе выполнения лабораторной работы были закреплены теоретические знания по созданию базы данных в СУБД MySQL с помощью программы phpMyAdmin. Приобретены практические навыки работы с данными средствами языка PHP

**Часть 6:**

**Тема:** работа с файловой системой.

Задание: Для разрабатываемых веб-сайтов создайте галерею фотографий. Требования:

1. Возможность загрузки фотографии на сервер (при загрузке должен проверяться тип файла).
2. Загруженная фотография помещается в новый каталог "uploads".
3. Вывод на экран всех загруженных фотографий.
4. При загрузке должна автоматически создаваться ее уменьшенная копия (не более 200 пикселей по наибольшей стороне).
5. Для хранения файлов используются два каталога - один для маленьких изображений, другой - для больших.
6. Возможность просмотра уменьшенных фотографий (все на одной странице).
7. Возможность увеличения фотографии при клике по ней (переход на отдельную страницу).

Результат выполнения данных заданий представлен в HTML документах ниже:

<РЕЗУЛЬТАТ>

Результаты работы страницы при различных сценариях представлены на рисунках 5-

**Заключение:** В ходе выполнения лабораторной работы были закреплены теоретические знания по работе с файлами в PHP.