Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский

Томский политехнический Университет»



Инженерная школа ядерных технологий

Направление 01.04.02 «Прикладная математика и информатика»

**ОТЧЕТ**

по лабораторной работе

Машина Тьюринга

по дисциплине:

**История и методология прикладной математики и информатики**

|  |  |  |  |
| --- | --- | --- | --- |
| **Исполнитель:** |  | Е. В. Петрович | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| студент группы 0ВМ92 |  | Дата сдачи: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |  |  |
|  |  |  |  |
| **Руководитель:** |  | Ю. Б. Буркатовская | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| доцент, |  | Дата проверки: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| кандидат физико-математических наук |  |  |  |
|  |  |  |  |

Томск – 2020

Оглавление

[Задание варианта 8 2](#_Toc42206358)

[**Цель работы** 3](#_Toc42206359)

[**Основная часть** 3](#_Toc42206360)

[**Приложение 1** 8](#_Toc42206361)

[index.php 8](#_Toc42206362)

[lab\_6.php 9](#_Toc42206363)

[style.css 13](#_Toc42206364)

[**Приложение 2** 14](#_Toc42206365)

[Ссылка на проект 14](#_Toc42206366)

**Задание**

Задание 1

Удвоить строку (например, 00102 превратить в 0010200102).

Задание 2

Проверить корректность записи арифметического выражения, например:

1. (x\*(x+x)+x)\*(x+x)+x - корректно
2. (x\*(x++x)+x)\*(x+x+x - некорректно

**Цель работы**

Изучить работу алгоритмов на машине Тьюринга.

**Основная часть**

PHP – один из популярных скриптовых языков для выполнения на web-сервере. При помощи программ, написанных на PHP, сервер генерирует HTML код страницы и отправляет его браузеру пользователя. Для хранения данных, необходимых для генерирования HTML кода используются базы данных. Наиболее распространенной связкой в настоящее время является web-сервер Apache, интерпретатор PHP, база данных MySQL. Данный набор программных продуктов был развернут на компьютере, подключенном к локальной сети, и использован для выполнения данной работы. Для демонстрации работы сайта доступ к нему был выполнен с двух компьютеров локальной сети.

Код лабораторной работы представлен в Приложении 1. Ссылка на репозиторий github представлена в Приложении 2. Работа программы в браузере представлена на Рис. 1-4.

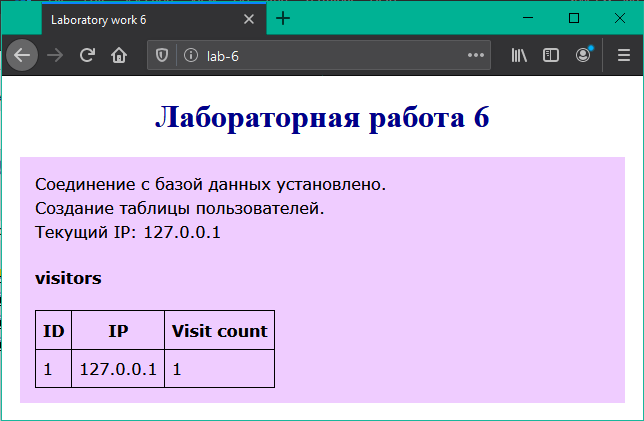


Рис. 1. Первое посещение сайта с сервера.

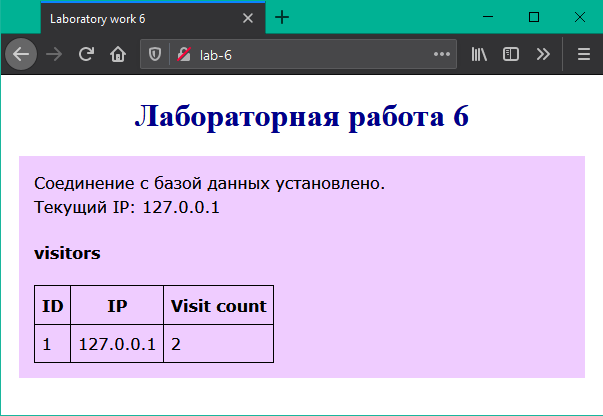


Рис. 2. Второе посещение сайта с сервера.

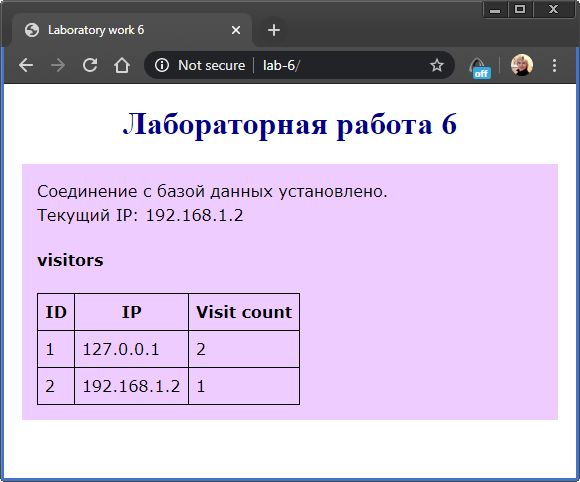


Рис. 3. Первое посещение сайта с компьютера из локальной сети.

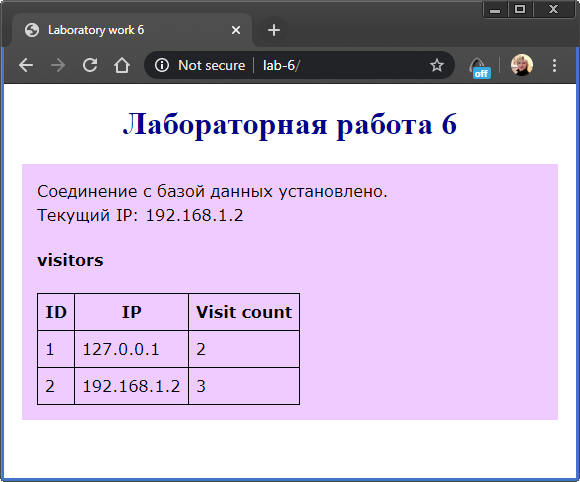


Рис. 4. Третье посещение сайта с компьютера из локальной сети.

**Заключение**

В работе написаны программы на языке PHP, продемонстрирована работа с базой данных MySQL. Представлены скриншоты работы программы, исходные коды.

**Приложение 1**

index.php

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Laboratory work 6</title>

<?php require('php\lab-6-8.php')?>

</head>

<body>

<h1>Лабораторная работа 6</h1>

<?php runLabScript(); ?>

</body>

</html>

lab\_6.php

<?php

$cssPath = 'http://lab-6/css/style.css';

echo "<link rel='stylesheet' href=$cssPath type='text/css'>";

$bd = null;

$host = "127.0.0.1";

$db\_name = "lab6";

$username = "root";

$password = "root";

$tableName = "visitors";

function runLabScript()

{

global $db;

echo '<div>';

if (!connectToDatabase($msg)) {

echo 'Невозможно соединиться с базой данных!<br>Ошибка: ' . $msg;

return;

};

echo 'Соединение с базой данных установлено.<br>';

if (!checkDbTable()) {

echo 'Создание таблицы пользователей.<br>';

if (!createDbTable($msg)) {

echo 'Невозможно создать таблицу в базе данных!<br>' . $msg;

$db->close();

return;

}

}

if (!getUserIp($ip)) {

echo 'Невозможно получить IP пользователя!<br>';

$db->close();

return;

}

echo 'Текущий IP: ' . $ip . '<br>';

if (!recordVisit($ip, $msg)) {

echo 'Невозможно записать данные в таблицу!<br>' . $msg;

$db->close();

return;

}

if (!getTable($table, $msg)) {

echo 'Невозможно получить таблицу!<br>' . $msg;

$db->close();

return;

}

outputTable($table);

echo '</div>';

$table->close();

$db->close();

}

function connectToDatabase(&$msg)

{

global $db, $host, $username, $password, $db\_name;

$db = new mysqli($host, $username, $password);

if ($db->connect\_errno !== 0) {

$msg = $db->connect\_errno . ': ' . $db->connect\_error;

return false;

}

if ($db->select\_db($db\_name)) {

return true;

}

$query = "CREATE DATABASE $db\_name";

if (!$db->query($query)) {

$msg = $db->error;

return false;

}

if (!$db->select\_db($db\_name)) {

$msg = $db->error;

return false;

}

return true;

}

function checkDbTable()

{

global $db, $db\_name, $tableName;

$tableList = $db->query("SHOW TABLES FROM $db\_name");

while ($row = $tableList->fetch\_row()) {

if ($row[0] === $tableName) return true;

}

$tableList->close();

return false;

}

function createDbTable(&$msg)

{

global $db, $tableName;

$query = "CREATE TABLE $tableName (

id INT(6) UNSIGNED AUTO\_INCREMENT PRIMARY KEY,

ip VARCHAR(16) NOT NULL,

visit\_count INT(6) NOT NULL

)";

if ($db->query($query) === FALSE) {

$msg = $msg = $db->connect\_errno . ': ' . $db->connect\_error;

return false;

}

return true;

}

function getUserIp(&$ip\_address)

{

if (!empty($\_SERVER['HTTP\_CLIENT\_IP'])) {

$ip\_address = $\_SERVER['HTTP\_CLIENT\_IP'];

} elseif (!empty($\_SERVER['HTTP\_X\_FORWARDED\_FOR'])) {

$ip\_address = $\_SERVER['HTTP\_X\_FORWARDED\_FOR'];

} else {

$ip\_address = $\_SERVER['REMOTE\_ADDR'];

}

return !$ip\_address->empty;

}

function recordVisit($ip, &$msg)

{

global $db, $tableName;

$query = "SELECT \* FROM $tableName WHERE ip='$ip'";

$record = $db->query($query);

if ($db->error) {

$msg = $db->error;

return false;

}

if ($record->num\_rows === 0) {

// first visit from current ip

$query = "INSERT INTO $tableName (ip, visit\_count) VALUES ('$ip', 1)";

} else {

$row = $record->fetch\_row();

$visitCount = $row[2] + 1;

$query = "UPDATE $tableName SET visit\_count=$visitCount WHERE ip='$ip'";

}

if ($db->query($query) === FALSE) {

$msg = $db->error;

return false;

}

$record->close();

return true;

}

function getTable(&$table, &$msg)

{

global $db, $tableName;

$query = "SELECT \* FROM $tableName";

$table = $db->query($query);

if ($db->error) {

$msg = $db->error;

return false;

}

return true;

}

function outputTable($table)

{

global $tableName;

echo "<h4>$tableName</h4>";

echo '<table>';

echo "<tr><th>ID</th><th>IP</th><th>Visit count</th></tr>";

while ($row = $table->fetch\_row()) {

echo "<tr><td>";

echo $row[0];

echo "</td><td>";

echo $row[1];

echo "</td><td>";

echo $row[2];

echo "</td></tr>";

}

echo "</table>";

}

style.css

table {

border-collapse: collapse;

}

table,

th,

td {

border: 1px solid black;

padding: 5pt;

}

h1 {

color: darkblue;

text-align: center;

}

div {

background-color: #efccff;

line-height: 150%;

font-family: Verdana, Geneva, Tahoma, sans-serif;

font-size: 16px;

margin: 10px;

padding: 15px;

}

**Приложение 2**

Ссылка на проект

<https://github.com/petrovicheugene/WebProgramming.git>