МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА №  43

ОТЧЁТ

ЗАЩИЩЁН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

Старший преподаватель                                                               Соловьева Н.А.

должность, уч. Степень, звание   подпись, дата                    инициалы, фамилия

ОТЧЁТ О ЛАБОРАТОРНОЙ РАБОТЕ №6

Серверные сценарии. Работа с базой данных.

по курсу: Web-технологии.

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. 4136                                                                        Бобрович Н. С.

                                                                         подпись, дата                      инициалы, фамилия

Санкт-Петербург 2024

**Содержание отчёта:**

[Содержание отчёта: 2](#_Toc11869)

[1. Цель работы: 3](#_Toc9645)

[2. Вариант задания: 3](#_Toc20479)

[3. Структура создания БД: 3](#_Toc31029)

[4. Данные в таблицах: 3](#_Toc14914)

[5. HTML-код: 4](#_Toc24961)

[7. Скриншоты: 13](#_Toc2877)

[8. Выводы по работе: 14](#_Toc21392)

# Цель работы:

Изучение трехуровневой архитектуры веб-приложений (Клиент – Веб-сервер – Сервер БД) и языка написания серверного сценария php для построения динамического контента web-сайтов.

# Вариант задания:

|  |  |
| --- | --- |
| № варианта | Тема |
| 24 | Форматы для хранения звука |

# **Структура создания БД:**

CREATE TABLE formats (

id\_format int NOT NULL,

name\_format varchar(50) NOT NULL,

extention varchar(10) NOT NULL,

bitrait int NOT NULL,

frequency int NOT NULL,

depth int NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

CREATE TABLE audiofiles (

id\_file int NOT NULL,

name\_file varchar(100) NOT NULL,

size bigint NOT NULL,

duration time NOT NULL,

id\_format int NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4\_0900\_ai\_ci;

# Данные в таблицах:

-- Заполнение таблицы "formats"

INSERT INTO `formats` (`name\_format`, `extention`, `bitrait`, `frequency`, `depth`)

VALUES

('WAV', '.wav', 1411, 44100, 16),

('MP3', '.mp3', 320, 44100, 16),

('FLAC', '.flac', 1411, 44100, 24);

-- Заполнение таблицы "audiofiles"

INSERT INTO `audiofiles` (`name\_file`, `size`, `duration`, `id\_format`)

VALUES

('Песня 1', 25432145, '00:03:45', 1),

('Песня 2', 12345678, '00:02:30', 2),

('Песня 3', 56789012, '00:04:10', 1),

('Песня 4', 34567890, '00:03:00', 3),

('Песня 5', 78901234, '00:05:20', 2);

# HTML-код:

<!DOCTYPE html>

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<meta name="Keywords" content="звук, хранение, хранение звука">

<title>Bobrovich\_4136\_WT</title>

<link rel=stylesheet href="style.css" type="text/css">

</head>

<body bgcolor="#9B9B9B">

<a href="main.html"><img src="logo.jpg"></a>

<h1 align="center">Форматы для хранения звука</h1>

<h2 align="center">Главная страница</h2>

<p><font size="4" face="Cambria">

<hr color="black">

Карта сайта:

<br><a href="main.html">Главная страница</a>

<br><a href="dop.html">Дополнительная информация</a>

<br><a href="sources.html">Источники</a>

<br><a href="lab3\_1.htm">Если скучно читать про форматы для хранения звука</a>

<br><a href="lab3\_2.htm">Если очень скучно читать про форматы для хранения звука</a>

<br><a href="forma.htm">Анкета посетителя сайта</a>

<hr color="black">

<ol>

<li><b>Цифровые аудиоформаты.</b></li>

<br>Цифровой аудиоформат — формат представления звуковых данных, используемый при цифровой звукозаписи, а также для дальнейшего хранения записанного материала на компьютере и других электронных носителях информации, так называемых звуковых носителях.

<br>Аудиофайл (файл, содержащий звукозапись) — компьютерный файл, состоящий из информации об амплитуде и частоте звука, сохранённой для дальнейшего воспроизведения на компьютере или проигрывателе.

<li><b>Разновидности цифровых аудиоформатов.</b></li>

<br>Существуют различные понятия звукового формата.

Формат представления звуковых данных в цифровом виде зависит от способа квантования аналогово-цифровым преобразователем (АЦП). В звукотехнике в настоящее время наиболее распространены два вида квантования:

<ul>

<li>импульсно-кодовая модуляция,</li>

<li>сигма-дельта-модуляция.</li>

</ul>

Зачастую разрядность квантования и частоту дискретизации указывают для различных звуковых устройств записи и воспроизведения как формат представления цифрового звука (24 бита/192 кГц; 16 бит/48 кГц).

Формат файла определяет структуру и особенности представления звуковых данных при хранении на запоминающем устройстве ПК. Для устранения избыточности аудиоданных используются аудиокодеки, при помощи которых производится сжатие аудиоданных. Выделяют три группы звуковых форматов файлов:

<ul>

<li>аудиоформаты без сжатия, такие как WAV, AIFF,</li>

<li>аудиоформаты со сжатием без потерь (APE, FLAC),</li>

<li>аудиоформаты со сжатием с потерями (MP3, Ogg).</li>

</ul>

Особняком стоят модульные музыкальные форматы файлов, созданные синтетически или из семплов заранее записанных живых инструментов. Они в основном служат для создания современной электронной музыки (MOD). Также сюда можно отнести формат MIDI, который не является звукозаписью, но с помощью секвенсора позволяет записывать и воспроизводить музыку, используя определённый набор команд в текстовом виде.

Форматы носителей цифрового звука применяют как для массового распространения звуковых записей (CD, SACD), так и в профессиональной звукозаписи (DAT, минидиск).

Для систем пространственного звучания также можно выделить форматы звука, в основном являющиеся звуковым многоканальным сопровождением к кинофильмам. Такие системы имеют целые семейства форматов, созданные двумя крупными конкурирующими компаниями — Digital Theater Systems Inc. (DTS) и Dolby Laboratories Inc. (Dolby Digital).

Также форматом называют количество каналов в системах многоканального звука (5.1; 7.1). Изначально такая система была разработана для кинотеатров, но впоследствии была расширена для систем домашнего кинотеатра.

</ol>

<?php

// Подключение к базе данных

$servername = "localhost";

$username = "root";

$password = "12345";

$dbname = "lab6";

$conn = new mysqli($servername, $username, $password, $dbname);

// Обработка форм

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST["action"])) {

// Обработка формы добавления

if ($\_POST["action"] == "add") {

$id\_format = $\_POST["id\_format"];

$name\_format = $\_POST["name\_format"];

$extention = $\_POST["extention"];

$bitrait = $\_POST["bitrait"];

$frequency = $\_POST["frequency"];

$depth = $\_POST["depth"];

$name\_file = $\_POST["name\_file"];

$size = $\_POST["size"];

$duration = $\_POST["duration"];

// Проверяем, существует ли id\_format в таблице formats

$sql\_check = "SELECT 1 FROM formats WHERE id\_format = ?";

$stmt\_check = $conn->prepare($sql\_check);

$stmt\_check->bind\_param("i", $id\_format);

$stmt\_check->execute();

$result\_check = $stmt\_check->get\_result();

if ($result\_check->num\_rows > 0) {

// id\_format существует, добавляем запись в audiofiles

$sql = "INSERT INTO audiofiles (id\_format, name\_file, size, duration)

VALUES (?, ?, ?, ?)";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("issd", $id\_format, $name\_file, $size, $duration);

$stmt->execute();

// Перенаправляем пользователя на эту же страницу, чтобы обновить таблицу

header("Location: " . $\_SERVER["PHP\_SELF"]);

exit;

} else {

// id\_format не существует, выводим сообщение об ошибке

echo "Ошибка: Формат с ID $id\_format не существует в базе данных.";

}

}

// Обработка формы редактирования

elseif ($\_POST["action"] == "edit") {

$id\_file = $\_POST["id\_file"];

$id\_format = $\_POST["id\_format"];

$name\_file = $\_POST["name\_file"];

$size = $\_POST["size"];

$duration = $\_POST["duration"];

$sql = "UPDATE audiofiles

SET id\_format = ?, name\_file = ?, size = ?, duration = ?

WHERE id\_file = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("issdi", $id\_format, $name\_file, $size, $duration, $id\_file);

$stmt->execute();

header("Location: " . $\_SERVER["PHP\_SELF"]);

exit;

}

// Обработка формы удаления

elseif ($\_POST["action"] == "delete") {

$id\_file = $\_POST["id\_file"];

$sql = "DELETE FROM audiofiles WHERE id\_file = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("i", $id\_file);

$stmt->execute();

header("Location: " . $\_SERVER["PHP\_SELF"]);

exit;

}

}

// Выборка данных из базы

$sql = "

SELECT

f.id\_format,

f.name\_format,

f.extention,

f.bitrait,

f.frequency,

f.depth,

a.id\_file,

a.name\_file,

a.size,

a.duration

FROM formats f

JOIN audiofiles a

ON f.id\_format = a.id\_format

";

$result = $conn->query($sql);

// Вывод таблицы

echo "<table border='1'>";

echo "<tr>

<th>ID Формата</th>

<th>Название Формата</th>

<th>Расширение</th>

<th>Битрейт</th>

<th>Частота</th>

<th>Глубина</th>

<th>ID Файла</th>

<th>Название Файла</th>

<th>Размер</th>

<th>Длительность</th>

<th>Действия</th>

</tr>";

if ($result->num\_rows > 0) {

while($row = $result->fetch\_assoc()) {

echo "<tr>

<td>".$row["id\_format"]."</td>

<td>".$row["name\_format"]."</td>

<td>".$row["extention"]."</td>

<td>".$row["bitrait"]."</td>

<td>".$row["frequency"]."</td>

<td>".$row["depth"]."</td>

<td>".$row["id\_file"]."</td>

<td>".$row["name\_file"]."</td>

<td>".$row["size"]."</td>

<td>".$row["duration"]."</td>

<td>

<a href='".$\_SERVER["PHP\_SELF"]."?edit=".$row["id\_file"]."'>Редактировать</a>

<form method='post' action='".$\_SERVER["PHP\_SELF"]."'>

<input type='hidden' name='id\_file' value='".$row["id\_file"]."'>

<input type='hidden' name='action' value='delete'>

<input type='submit' value='Удалить'>

</form>

</td>

</tr>";

}

} else {

echo "<tr><td colspan='11'>Нет данных</td></tr>";

}

echo "</table>";

// Форма для добавления новой записи

echo "<h2>Добавить новую запись</h2>";

echo "<form method='post' action='" . $\_SERVER["PHP\_SELF"] . "'>";

echo "ID Формата: <input type='text' name='id\_format'><br>";

echo "Название Формата: <input type='text' name='name\_format'><br>";

echo "Расширение: <input type='text' name='extention'><br>";

echo "Битрейт: <input type='text' name='bitrait'><br>";

echo "Частота: <input type='text' name='frequency'><br>";

echo "Глубина: <input type='text' name='depth'><br>";

echo "Название Файла: <input type='text' name='name\_file'><br>";

echo "Размер: <input type='text' name='size'><br>";

echo "Длительность: <input type='text' name='duration'><br>";

echo "<input type='hidden' name='action' value='add'>";

echo "<input type='submit' value='Добавить'>";

echo "</form>";

// Форма для редактирования записи

if (isset($\_GET["edit"])) {

$id\_file = $\_GET["edit"];

$sql = "SELECT \* FROM audiofiles WHERE id\_file = ?";

$stmt = $conn->prepare($sql);

$stmt->bind\_param("i", $id\_file);

$stmt->execute();

$result = $stmt->get\_result();

$row = $result->fetch\_assoc();

echo "<h2>Редактировать запись</h2>";

echo "<form method='post' action='" . $\_SERVER["PHP\_SELF"] . "'>";

echo "ID Файла: <input type='text' name='id\_file' value='" . $row["id\_file"] . "' readonly><br>";

echo "ID Формата: <input type='text' name='id\_format' value='" . $row["id\_format"] . "'><br>";

echo "Название Файла: <input type='text' name='name\_file' value='" . $row["name\_file"] . "'><br>";

echo "Размер: <input type='text' name='size' value='" . $row["size"] . "'><br>";

echo "Длительность: <input type='text' name='duration' value='" . $row["duration"] . "'><br>";

echo "<input type='hidden' name='action' value='edit'>";

echo "<input type='submit' value='Сохранить'>";

echo "</form>";

}

// Форма для удаления записи

if (isset($\_GET["delete"])) {

$id\_file = $\_GET["delete"];

echo "<h2>Удалить запись</h2>";

echo "<form method='post' action='" . $\_SERVER["PHP\_SELF"] . "'>";

echo "Вы действительно хотите удалить эту запись?";

echo "<input type='hidden' name='id\_file' value='" . $id\_file . "'>";

echo "<input type='hidden' name='action' value='delete'>";

echo "<input type='submit' value='Да, удалить'>";

echo "</form>";

}

$conn->close();

?>

</font>

</p>

</body>

</html>

1. **Код серверных скриптов:**

1. Подключение к базе данных:

$servername = "localhost";

$username = "root";

$password = "12345";

$dbname = "lab6";

$conn = new mysqli($servername, $username, $password, $dbname);

2. Обработка форм (добавление, редактирование, удаление):

if ($\_SERVER["REQUEST\_METHOD"] == "POST" && isset($\_POST["action"])) {

// Обработка формы добавления

if ($\_POST["action"] == "add") {

// Код для добавления новой записи

}

// Обработка формы редактирования

elseif ($\_POST["action"] == "edit") {

// Код для редактирования записи

}

// Обработка формы удаления

elseif ($\_POST["action"] == "delete") {

// Код для удаления записи

}

}

3. Выборка данных из базы:

$sql = "

SELECT

f.id\_format,

f.name\_format,

f.extention,

f.bitrait,

f.frequency,

f.depth,

a.id\_file,

a.name\_file,

a.size,

a.duration

FROM formats f

JOIN audiofiles a

ON f.id\_format = a.id\_format

";

$result = $conn->query($sql);

4. Вывод HTML-таблицы с данными:

echo "<table border='1'>";

// Формирование HTML-таблицы

echo "</table>";

5. Вывод форм для добавления, редактирования и удаления записей:

echo "<h2>Добавить новую запись</h2>";

echo "<form method='post' action='" . $\_SERVER["PHP\_SELF"] . "'>";

// Формирование формы для добавления

echo "</form>";

if (isset($\_GET["edit"])) {

// Формирование формы для редактирования

}

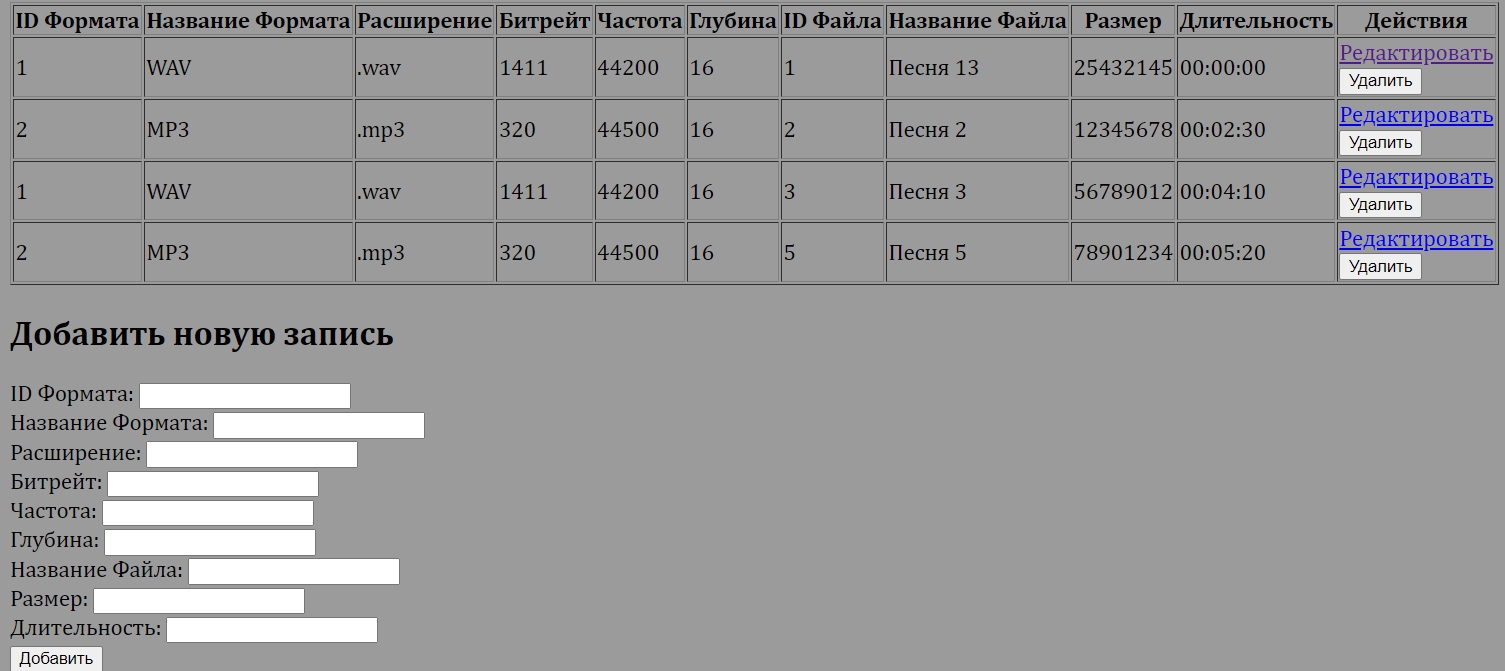
if (isset($\_GET["delete"])) {

// Формирование формы для удаления

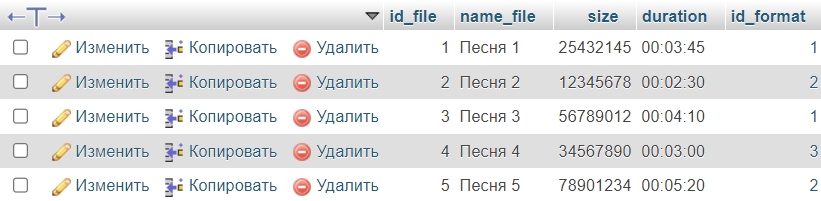
}

# Скриншоты:

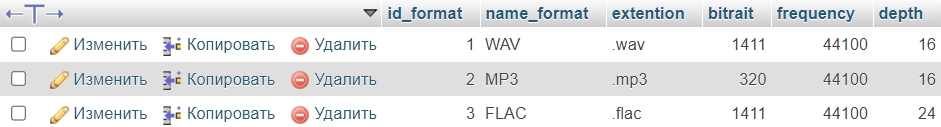
**Рис. 1: Результирующая таблица на сайте**



**Рис. 2: Таблица audiofiles в sql**



**Рис. 3: Таблица formats в sql**



**Рис. 4: Данные таблицы formats в sql**



**Рис. 5: Данные таблицы audiofiles в sql**



# Выводы по работе:

Изучил трехуровневую архитектуру веб-приложений (Клиент – Веб-сервер – Сервер БД) и язык написания серверного сценария php для построения динамического контента web-сайтов.