

Санкт-Петербургский Национальный Исследовательский Университет
Информационных Технологий, Механики и Оптики
Факультет Инфокоммуникационных Технологий
Направление подготовки Программирование в инфокоммуникационных
системах

Лабораторная работа №4.

Выполнил
Сидненко Д.Ю.
Группа №К3320
Проверила
Марченко Е.В.

Санкт-Петербург, 2024

СОДЕРЖАНИЕ

ЦЕЛЬ РАБОТЫ	3
ХОД РАБОТЫ.....	4
Задание 1.....	4
Задания 2	7
Задание 3.....	9
ВЫВОД	11

ЦЕЛЬ РАБОТЫ

1) Разработать веб-страницу, на которой пользователь может оставить данные о себе – фамилия, имя, отчество, адрес для физической доставки товара, телефон, адрес электронной почты. Товары покупатель выбирает из списка (реализовать меню), предусмотреть отдельную позицию, в которой пользователь может оставить комментарий по заказу. Данные, введенные в форму, обрабатываются php скриптом и заносятся в таблицу БД MySQL Server. Разработать структуру таблицы, создать таблицу и скрипт php.

2) В движке wordpress используется форма для авторизации пользователей – пользователь вводит логин и пароль. Логин и хэш от пароля записываются в таблицу БД MySQL. Необходимо модифицировать php скрипт, чтобы дополнительно логин и пароль записывались в отдельную новую таблицу в БД MySQL. Предусмотреть два варианта записи: а) пароль записывается в исходном виде, б) инвертируются биты бат пароля (ноль меняется на единицу, единица меняется на ноль).

3) Выбрать среду разработки (язык программирования) на ваше усмотрение, используя его написать web-сервер. В программе предусмотреть возможность указать порт, на котором будет работать сервер. При обращении вида `http://127.0.0.1:888/` где 888 – заданный порт, сервер должен вернуть содержимое файла `index.html`, находящегося в том же каталоге, что и исполняемый файл написанной программы-сервера.

ХОД РАБОТЫ

Задание 1.

В начале выполнения работы была разработана веб страница, на которой пользователь может оставить данные о себе. Была создана база данных, которая хранит все отправленные формы. Форма обрабатывается php скриптом.

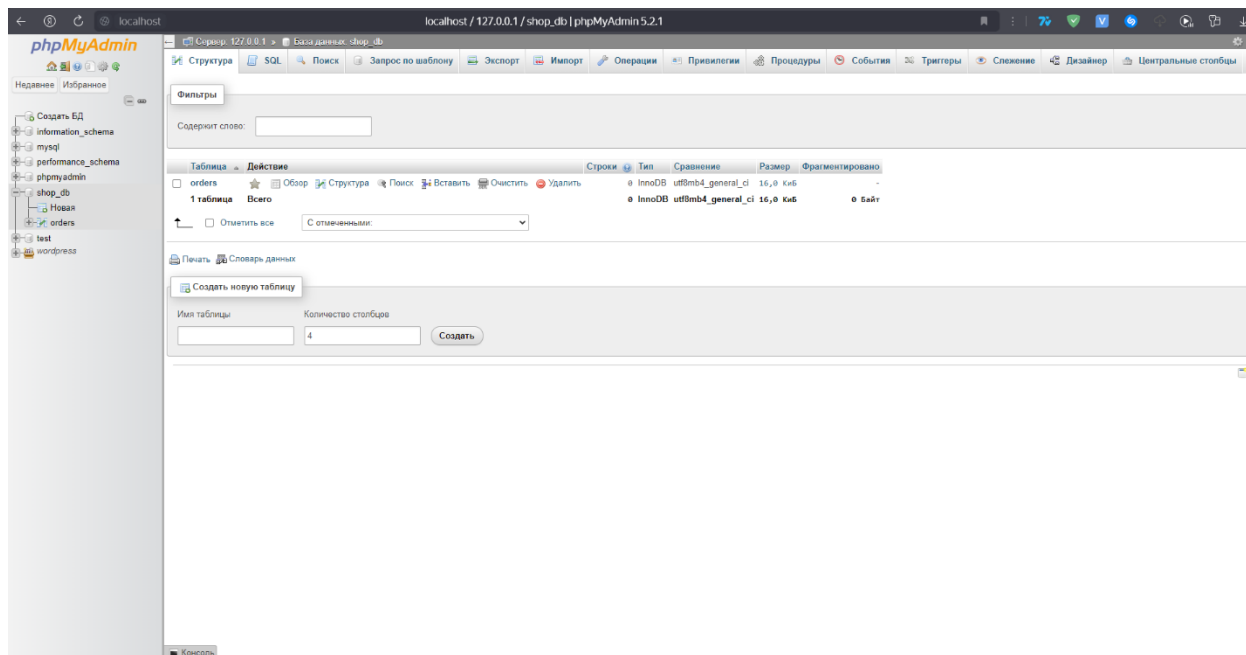


Рисунок 1 – Создание таблицы orders

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Форма заказа</title>
</head>
<body>
  <h1>Оформление заказа</h1>
  <form action="process.php" method="post">
    <label for="surname">Фамилия:</label>
    <input type="text" id="surname" name="surname" required><br><br>

    <label for="name">Имя:</label>
    <input type="text" id="name" name="name" required><br><br>

    <label for="patronymic">Отчество:</label>
    <input type="text" id="patronymic" name="patronymic"><br><br>

    <label for="address">Адрес доставки:</label>
    <textarea id="address" name="address" required></textarea><br><br>

    <label for="phone">Телефон:</label>
    <input type="tel" id="phone" name="phone" required><br><br>

    <label for="email">Email:</label>
    <input type="email" id="email" name="email" required><br><br>

    <label for="product">Товары:</label>
    <select id="product" name="product" required>
      <option value="Товар 1">Товар 1</option>
      <option value="Товар 2">Товар 2</option>
      <option value="Товар 3">Товар 3</option>
    </select><br><br>

    <label for="comment">Комментарий:</label>
    <textarea id="comment" name="comment"></textarea><br><br>

    <button type="submit">Отправить заказ</button>
  </form>
</body>
</html>

```

Рисунок 2 – Html код формы

```

<?php
// Подключение к базе данных
$host = 'localhost';
$dbname = 'shop_db';
$username = 'root';
$password = ''; // Укажите ваш пароль, если есть

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

// Проверка подключения
if ($conn->connect_error) {
    die("Ошибка подключения: " . $conn->connect_error);
}

// Получение данных из формы
$surname = $_POST['surname'];
$name = $_POST['name'];
$patronymic = $_POST['patronymic'];
$address = $_POST['address'];
$phone = $_POST['phone'];
$email = $_POST['email'];
$product = $_POST['product'];
$comment = $_POST['comment'];

// SQL-запрос для вставки данных
$sql = "INSERT INTO orders (surname, name, patronymic, address, phone, email, product, comment)
VALUES ('$surname', '$name', '$patronymic', '$address', '$phone', '$email', '$product', '$comment')";

// Выполнение запроса
if ($conn->query($sql) === TRUE) {
    echo "Ваш заказ успешно оформлен!";
} else {
    echo "Ошибка: " . $sql . "<br>" . $conn->error;
}

// Закрытие подключения
$conn->close();
?>

```

Рисунок 3 – Php скрипт формы

Extra options

	id	surname	name	patronymic	address	phone	email	product	comment
<input type="checkbox"/> Изменить Копировать Удалить	1	Сидненко	Дмитрий	Юрьевич	443237	+79197453407	DmitriySidnenko@yandex.ru	Товар 2	nlg

☐ Отметить все С отмеченными: Изменить Копировать Удалить Экспорт

☐ Показать все Количество строк: 25 Фильтровать строки: Поиск в таблице

Использование результатов запроса

Метка: ☐ Доступна для всех пользователей

Рисунок 4 – Результат выполнения

Задания 2

В этом задании был модифицирован php код движка WordPress, чтобы при авторизации данные сохранялись в таблицу, а также инвертировались биты пароля.

The screenshot shows a database management interface. At the top, a SQL query is entered: `SELECT * FROM `wp_user_password_log``. Below the query bar are several action buttons: ☐ Профилитрование, [Построчное редактирование], [Изменить], [Анализ SQL запроса], [Создать PHP-код], and [Обновить]. Below these buttons, the table structure is displayed with columns: `id`, `username`, `original_password`, `inverted_password`, and `created_at`. The interface includes a section titled "Использование результатов запроса" with a button "Создать представление". Below this is a section titled "Создать закладку" with a text input field labeled "Метка:" and a checkbox labeled "Доступна для всех пользователей". At the bottom of this section is a button "Создать закладку".

Рисунок 5 – Создание базы данных

```

209
210 // Хук для записи логина и пароля в новую таблицу при авторизации
211 add_action('wp_login', 'log_user_password_on_login', 10, 2);
212
213 function log_user_password_on_login($user_login, $user) {
214     global $wpdb; // Подключение к базе данных WordPress
215
216     // Получение логина и хэшированного пароля
217     $username = $user_login;
218     $hashed_password = $user->user_pass;
219
220     // Для примера исходный пароль перехватить не удаётся, добавим заглушку
221     // Для "исходного" пароля используем фиктивное значение или получаем его при регистрации
222     $original_password = 'Lvbnhbq0819'; // Это значение заменить на реальный пароль при регистрации
223
224     // Генерация инверсии пароля
225     $inverted_password = invert_password_bits($original_password);
226
227     // Вставка данных в таблицу wp_user_password_log
228     $wpdb->insert(
229         'wp_user_password_log',
230         [
231             'username' => $username,
232             'original_password' => $original_password,
233             'inverted_password' => $inverted_password,
234         ]
235     );
236 }
237
238 // функция для инверсии битов пароля
239 function invert_password_bits($password) {
240     $inverted_password = '';
241
242     // Определим карту замены символов (буквы и цифры)
243     $replacement_map = [
244         'a' => 'z', 'b' => 'y', 'c' => 'x', 'd' => 'w', 'e' => 'v', 'f' => 'u',
245         'g' => 't', 'h' => 's', 'i' => 'r', 'j' => 'q', 'k' => 'p', 'l' => 'o',
246         'm' => 'n', 'n' => 'm', 'o' => 'l', 'p' => 'k', 'q' => 'j', 'r' => 'i',
247         's' => 'h', 't' => 'g', 'u' => 'f', 'v' => 'e', 'w' => 'd', 'x' => 'c',
248         'y' => 'b', 'z' => 'a', // для строчных букв
249
250         'A' => 'Z', 'B' => 'Y', 'C' => 'X', 'D' => 'W', 'E' => 'V', 'F' => 'U',
251         'G' => 'T', 'H' => 'S', 'I' => 'R', 'J' => 'Q', 'K' => 'P', 'L' => 'O',
252         'M' => 'N', 'N' => 'M', 'O' => 'L', 'P' => 'K', 'Q' => 'J', 'R' => 'I',
253         'S' => 'H', 'T' => 'G', 'U' => 'F', 'V' => 'E', 'W' => 'D', 'X' => 'C',
254         'Y' => 'B', 'Z' => 'A', // для заглавных букв
255
256         '0' => '9', '1' => '8', '2' => '7', '3' => '6', '4' => '5', '5' => '4',
257         '6' => '3', '7' => '2', '8' => '1', '9' => '0', // для цифр
258     ];
259 }

```

Рисунок 6 – Модификация скрипта Php

Необходимо отметить, что не получилось реализовать инверсию битов пароля, вместо этого была реализована инверсия символов пароля.

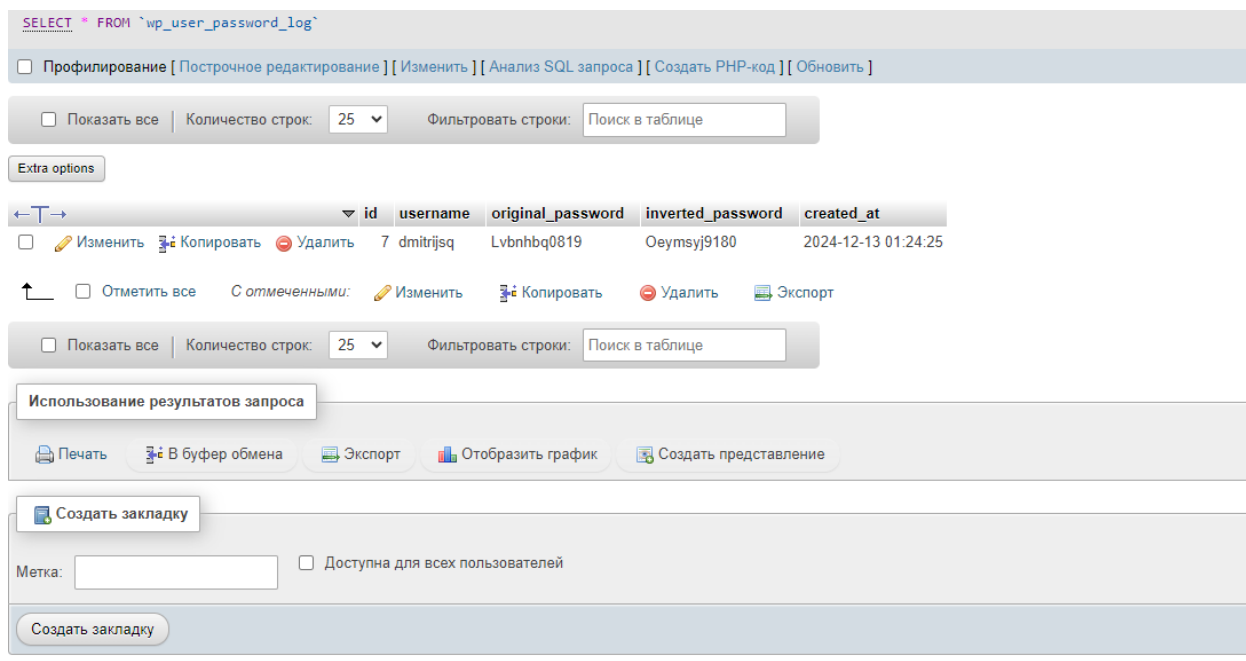


Рисунок 7 – Добавление записи авторизации

Задание 3.

В этом задании был выбран язык программирования Python для написания web-сервера. В программе предусмотрена возможность указать порт, на котором будет работать сервер.

```

1  import http.server
2  import socketserver
3  import os
4
5  def run_server(port): 1 usage
6      # Определяем адрес и порт
7      handler = http.server.SimpleHTTPRequestHandler
8      # Создаём сервер с указанным портом
9      with socketserver.TCPServer(server_address: ("", port), handler) as httpd:
10         print(f"Server running at http://127.0.0.1:{port}")
11         # Запуск сервера
12         httpd.serve_forever()
13
14  if __name__ == "__main__":
15     port = 888 # Например, 888
16     # Убедитесь, что файл index.html находится в том же каталоге
17     if os.path.exists("index.html"):
18         run_server(port)
19     else:
20         print("Error: index.html not found in the current directory.")
21

```

Рисунок 8 – Код python файла



Рисунок 9 – Запуск сервера на порту 888

ВЫВОД

В процессе выполнения лабораторной работы была разработана веб-страница, на которой пользователь может оставлять данные, с возможностью сохранения в таблицу БД. Также был модифицирован скрипт php движка WordPress для того, чтобы данные об авторизации заносились в таблицу БД. После этого был написан web-сервер на языке программирования Python.