

Министерство науки и высшего образования Российской Федерации
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО
ITMO University

Отчет по лабораторной работе 4

По дисциплине WEB-программирование

Тема работы «Работа с БД и веб-серверами»

Обучающийся Петрова Наталья Глебовна

Факультет факультет инфокоммуникационных технологий

Группа K3321

Направление подготовки 11.03.02 Инфокоммуникационные технологии и системы связи

Образовательная программа Программирование в инфокоммуникационных системах

Обучающийся

17.12.2024

(дата)

(подпись)

Петрова Н. Г.

(Ф.И.О.)

Руководитель

(дата)

(подпись)

Марченко Е.В.

(Ф.И.О.)

СОДЕРЖАНИЕ

Стр.

ВВЕДЕНИЕ	3
1 Задание 1	4
1.1 Создание БД.....	4
1.2 Написание HTML В PHP скриптов	5
1.3 Запуск сервера.....	6
2 Задание 2	8
3 Задание 3	11
ЗАКЛЮЧЕНИЕ	13

ВВЕДЕНИЕ

Цель работы

Целью данной работы является освоение навыков работы с формами и БД MySQL Server.

Задачи:

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.

1 Задание 1

1.1 Создание БД

Первое задание было посвящено разработке веб-страницы для оформления заказа в интернет-магазине (в моем случае мебельном), которая содержит форму, в которой пользователь может вписать информацию о себе.

Идея была в том, чтобы данные, которые пользователь ввел, при нажатии на кнопку "отправить" попадали в базу данных с помощью метода POST в php файле.

Первым делом необходимо было создать саму базу данных, где будет появляться информация о пользователе.

Согласно требованиям, была создана БД lab4 и в ней реализована таблица orders, которая содержит соответствующие поля: id, фамилия, имя, отчество, адрес, номер телефона, эл. почта, наименование товара, комментарий, а также время оформления заказа. База данных была создана по адресу <http://localhost/phpmyadmin>

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int			Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
2	first_name	varchar(255)	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
3	last_name	varchar(255)	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
4	father_name	varchar(255)	utf8mb4_0900_ai_ci		Да	NULL			Изменить Удалить Ещё
5	address	text	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
6	phone	varchar(28)	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
7	email	varchar(255)	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
8	tovar_name	text	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
9	comments	text	utf8mb4_0900_ai_ci		Да	NULL			Изменить Удалить Ещё
10	time_order	timestamp			Да	CURRENT_TIMESTAMP		DEFAULT_GENERATED	Изменить Удалить Ещё

Рисунок 1.1 — Структура таблицы order в БД lab4

Из структуры видно, что есть необязательные для заполнения пользователем поля, такие как отчество и комментарий. И также поле time_order, которое фиксирует точное время отправки формы и заносит эту информацию в таблицу.

1.2 Написание HTML В PHP скриптов

Для того, чтобы данные, введенные пользователем отправлялись в БД, необходимо было написать PHP скрипт, который с помощью метода POST добавляет информацию в строки таблицы.

```
script.php
1  <?php
2  // Подключение к базе данных
3  $host = 'localhost';
4  $db = 'lab4';
5  $user = 'root';
6  $pass = '';
7  $charset = 'utf8mb4';
8
9  $dsn = "mysql:host=$host;dbname=$db;charset=$charset";
10 $options = [
11     PDO::ATTR_ERRMODE => PDO::ERRMODE_EXCEPTION,
12     PDO::ATTR_DEFAULT_FETCH_MODE => PDO::FETCH_ASSOC,
13     PDO::ATTR_EMULATE_PREPARES => false,
14 ];
15
16 try {
17     $pdo = new PDO($dsn, $user, $pass, $options);
18 } catch (\PDOException $e) {
19     die("Ошибка подключения к базе данных: " . $e->getMessage());
20 }
21
22 // Получение данных из формы
23 $last_name = htmlspecialchars($_POST['last_name']);
24 $first_name = htmlspecialchars($_POST['first_name']);
25 $father_name = htmlspecialchars($_POST['father_name']);
26 $address = htmlspecialchars($_POST['address']);
27 $phone = htmlspecialchars($_POST['phone']);
28 $email = htmlspecialchars($_POST['email']);
29 $tovar_name = implode(' ', $_POST['tovar_name']);
30 $comments = htmlspecialchars($_POST['comments']);
31
32 // Вставка данных в таблицу
33 $sql = "INSERT INTO orders (last_name, first_name, father_name, address, phone, email, tovar_name, comments)
34     VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
35 $stmt = $pdo->prepare($sql);
36
37 try {
38     $stmt->execute([$last_name, $first_name, $father_name, $address, $phone, $email, $tovar_name, $comments]);
39     echo "<h1>Данные вашего заказа зафиксированы! Ожидайте звонка оператора для подтверждения.</h1>";
40 } catch (\PDOException $e) {
41     die("Ошибка записи данных: " . $e->getMessage());
42 }
43 ?>
```

Рисунок 1.2 — php скрипт

Очень долго не могла получить доступ к БД, так как не могла войти в личный кабинет администратора, ввиду того, что особенностью входа является имя пользователя root и ПУСТОЙ пароль.

Таким образом html файл сформировал форму, которую пользователь заполняет, после чего полученные данные с помощью php скрипта были доставлены до соответствующих полей таблицы в БД.

```

<> index.html > ...
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Форма заказа</title>
7  </head>
8  <body>
9      <h1>Оформление заказа</h1>
10     <h5>Пожалуйста, заполните форму</h5>
11     <form action="script.php" method="POST">
12         <label for="last_name">Фамилия:</label>
13         <input type="text" id="last_name" name="last_name" required><br><br>
14
15         <label for="first_name">Имя:</label>
16         <input type="text" id="first_name" name="first_name" required><br><br>
17
18         <label for="father_name">Отчество:</label>
19         <input type="text" id="father_name" name="father_name"><br><br>
20
21         <label for="address">Адрес:</label>
22         <textarea id="address" name="address" required></textarea><br><br>
23
24         <label for="phone">Телефон:</label>
25         <input type="tel" id="phone" name="phone" required><br><br>
26
27         <label for="email">Электронная почта:</label>
28         <input type="email" id="email" name="email" required><br><br>
29
30         <label for="tovar_name">Выберите товары:</label>
31         <select id="items" name="tovar_name[]" multiple required>
32             <option value="Шкаф">Шкаф</option>
33             <option value="Стол">Стол</option>
34             <option value="Стулья">Стулья</option>
35             <option value="Диван">Диван</option>
36             <option value="Дверь">Дверь</option>
37         </select><br><br>
38
39         <label for="comments">Комментарий:</label>
40         <textarea id="comments" name="comments"></textarea><br><br>
41
42         <button type="submit">Отправить</button>
43     </form>
44 </body>
45 </html>

```

Рисунок 1.3 — html файл

1.3 Запуск сервера

Данные файлы были созданы в папке lab_4_task1, которая расположена по пути: WAMP/www/lab_4_task1. Именно расположение в папке www дает возможность запустить файл на сервере.

Затем необходимо включить WAMP и в поисковой строке прописать соответствующий адрес: http://localhost/lab_4_task1/, после этого появится

написанная ранее форма, которую необходимо заполнить. После успешного заполнения данные автоматически появятся в таблице.

← ↻ 🏠 ⓘ localhost/lab_4_task1/

Оформление заказа

Пожалуйста, заполните форму

Фамилия:

Имя:

Отчество:

Адрес:

Телефон:

Электронная почта:

Выберите товары:

Шкаф
Стоп
Стулья
Диван

Комментарий:

Перед доставкой отзовонитесь за пол часа, пожалуйста.

Рисунок 1.4 — Результат выполнения скриптов

← ↻ 🏠 ⓘ localhost/lab_4_task1/script.php 🔍 ☆

Данные вашего заказа зафиксированы! Ожидайте звонка оператора для подтверждения.

Рисунок 1.5 — Результат выполнения скриптов

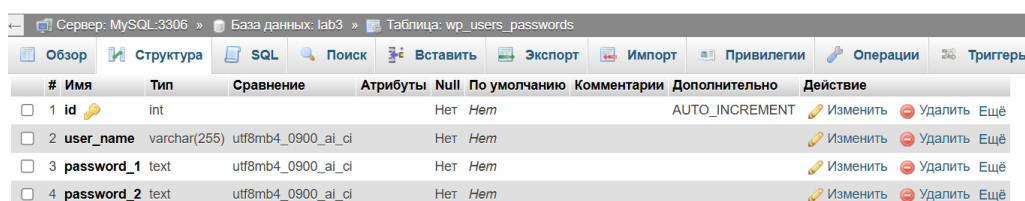
<input type="checkbox"/>	Изменить	Копировать	Удалить	5	Екатерина	Петрова	Ивановна	ул. Веденеева д1 кв1	89213132299	Ekaterina10@gmail.com	Шкаф, Стоп	Перед доставкой отзовонитесь за пол часа, пожалуйста...	2024-12-17 21:25:53
--------------------------	----------	------------	---------	---	-----------	---------	----------	----------------------	-------------	-----------------------	------------	---	---------------------

Рисунок 1.6 — Запись информации в таблицу

2 Задание 2

Второе задание было посвящено написанию скрипта, который позволяет сохранять в базу данных логин и пароль пользователя при авторизации на wordpress.

Для реализации данного задания была создана новая таблица wp_users_passwords в уже существующей БД lab3, которая была создана в лабораторной работе №3.



The screenshot shows a MySQL database management tool interface. The top bar indicates the server is MySQL:3306, the database is lab3, and the selected table is wp_users_passwords. Below the bar is a menu with options: Обзор, Структура, SQL, Поиск, Вставить, Экспорт, Импорт, Привилегии, Операции, and Триггеры. The main area displays the table structure with columns: #, Имя, Тип, Сравнение, Атрибуты, Null, По умолчанию, Комментарии, Дополнительно, and Действие. The table has four columns: id (int, AUTO_INCREMENT), user_name (varchar(255)), password_1 (text), and password_2 (text).

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
1	id	int			Нет	Нет		AUTO_INCREMENT	Изменить Удалить Ещё
2	user_name	varchar(255)	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
3	password_1	text	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё
4	password_2	text	utf8mb4_0900_ai_ci		Нет	Нет			Изменить Удалить Ещё

Рисунок 2.1 — Создание таблицы

Далее был написан код PHP, который является частью WordPress-плагина. В данном случае он нам позволяет регистрировать введенные данные. Этот код добавляет функцию lab4_task2 в очередь выполнения событий WordPress. Когда пользователь успешно входит в систему (срабатывает событие wp_login), вызывается функция lab4_task2. Таким образом, после каждого успешного входа система сохраняет два пароля: оригинальный и инвертированный.

Наиболее интересным моментом является запись второго пароля, так как первоначальный записывается сразу в том виде, в котором он был введен пользователем в строку для пароля.

Запись второго варианта пароля состоит из нескольких этапов (работа функции **invert**):

1. Первым делом происходит перевод ASCII-кода символа в двоичное представление.
2. Происходит инверсия: замена 0 на 1 и наоборот.

3. Затем полученная строка в двоичном представлении конвертируется обратно в десятичное число, после чего результат передается функции `wp_hash_password`, которая хеширует полученное значение. Это стандартный метод WordPress для создания безопасного пароля.

Далее функция **lab4_task2** делает следующее:

1. Получает два значения паролей: первое – непосредственно введенный пользователем пароль (`password_1`), второе – инвертированный вариант первого пароля, полученный через вызов функции `invert` (`password_2`).

2. Эти данные сохраняются в таблицу базы данных `wp_users_passwords`. В эту таблицу записываются имя пользователя и оба пароля (оригинальный и инвертированный).

```
plugin_4.1.php
1  <?php
2  /*
3  Plugin Name: plugin_lab_4.1
4  Description: Запись информации о пользователях: логин и пароль
5  Version: 1.0
6  Author: Петрова Наталья
7  */
8
9  function invert($input) {
10     $bin = '';
11     for ($i = 0; $i < strlen($input); $i++) {
12         $bin .= decbin(ord($input[$i]));
13     }
14
15     $bin_inv = '';
16     for ($i = 0; $i < strlen($bin); $i++) {
17         $bin_inv .= $bin[$i] == '0' ? '1' : '0';
18     }
19     return wp_hash_password(bindec($bin_inv));
20 }
21
22
23 function lab4_task2($user_name) {
24     global $wpdb;
25     $password_1 = $_POST['pwd'];
26     $password_2 = invert($password_1);
27
28     $wpdb->insert(
29         'wp_users_passwords',
30         [
31             'user_name' => $user_name,
32             'password_1' => $password_1,
33             'password_2' => $password_2,
34         ]
35     );
36 }
37
38 add_action('wp_login', 'lab4_task2');
39 ?>
```

Рисунок 2.2 — Код плагина

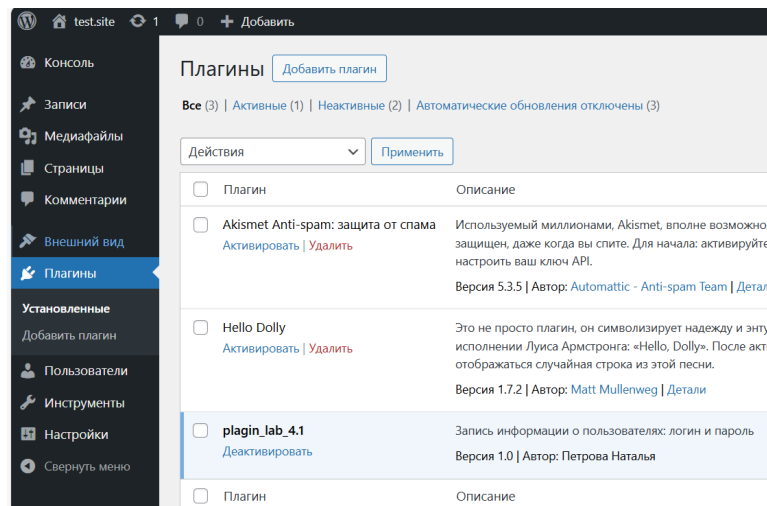


Рисунок 2.3 — Проверка наличия плагина на wordpress

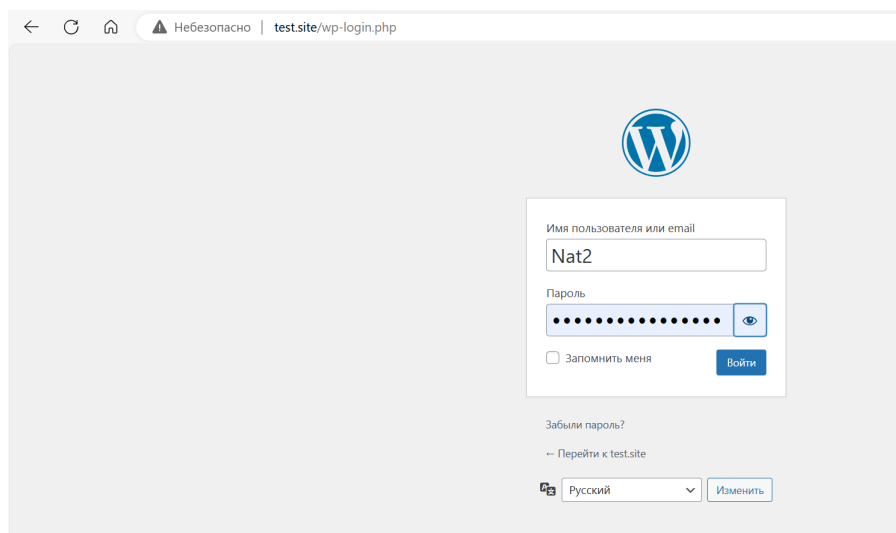


Рисунок 2.4 — Ввод логина и пароля в форму на wordpress



Рисунок 2.5 — Запись логина и пароля

Важно заметить, что такая практика хранения паролей в открытом виде крайне небезопасна!

3 Задание 3

Целью третьего задания было написание сервера, который бы запускал html файл. Для выполнения задания был написан веб-сервер на языке Python, так как данный язык более удобен и знаком. В качестве html файла был выбран написанный мною файл из первой лабораторной работы.

```
server.py > ...
1  import http.server
2  import socketserver
3  from urllib.parse import urlparse
4
5  # Адрес и порт сервера
6  ADDRESS = '127.0.0.1'
7  PORT = 8080
8
9  def main():
10     # Создаем обработчик запросов
11     handler = http.server.SimpleHTTPRequestHandler
12
13     # Запускаем TCP-сервер
14     with socketserver.TCPServer((ADDRESS, PORT), handler) as httpd:
15         print(f'Сервер запущен на {urlparse(f"http://{ADDRESS}:{PORT}").geturl()}')
16
17         try:
18             httpd.serve_forever() # Бесконечный цикл обработки запросов
19         except KeyboardInterrupt:
20             print('Сервер остановлен.')
21
22 if __name__ == '__main__':
23     main()
```

Рисунок 3.1 — Код сервера на языке Python

```
index.html > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <meta charset="utf-8">
5  <title>Праздничная веб страница</title>
6  </head>
7  <body style = "background-color: #lightblue; color: #blue">
8  <p style = "font: 3em Arial; text-align: center">HAPPY NEW YEAR!!!</p>
9  <p style = "font: 3em Arial; text-align: center">4 лабораторная выполнена!!!</p>
10 <body text = "#FF7F50">
11 <p>
12 
13
14
15 </body>
16 </html>
17
```

Рисунок 3.2 — HTML код страницы

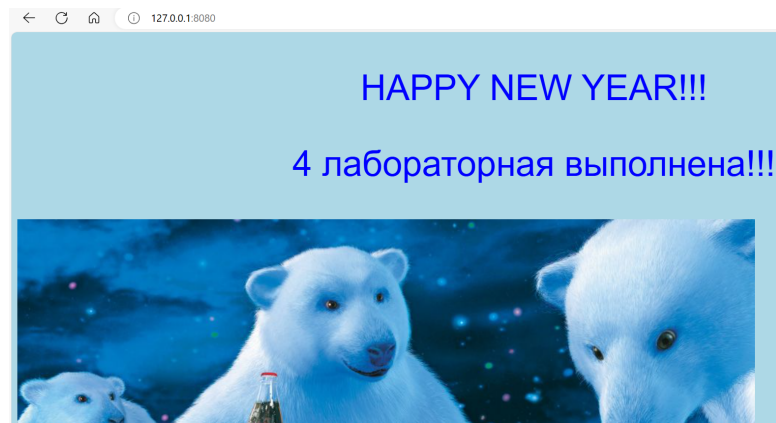


Рисунок 3.3 — Результат выполнения

```
PS C:\Users\Natac\OneDrive\Рабочий стол\Тест_3> python -u "c:\Users\Natac\OneDrive\Рабочий стол\Тест_3\server.py"  
Сервер запущен на http://127.0.0.1:8080  
127.0.0.1 - - [17/Dec/2024 21:51:13] "GET / HTTP/1.1" 304 -  
Сервер остановлен.
```

Рисунок 3.4 — Результат работы сервера (включение/выключение)

ЗАКЛЮЧЕНИЕ

В результате выполнения лабораторной работы были получены навыки работы с веб-серверами, а также улучшены навыки работы с php и html скриптами. Помимо этого были получены навыки работы с БД и привязка к ним информации из файлов.