

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**ITMO University**

Факультет «Факультет инфокоммуникационных технологий»

Направление подготовки «Инфокоммуникационные технологии и системы  
связи»

Отчет по дисциплине: «Web-программирование»

Практическая работа №4

Выполнила:

Колтунова Полина Владимировна

Группа №K3320

Проверила:

Марченко Елена Вадимовна

Санкт-Петербург

2024

## **ВВЕДЕНИЕ**

Цель работы: отработать навыки разработки веб-страниц, применив знания PHP, MySQL и основ веб-программирования.

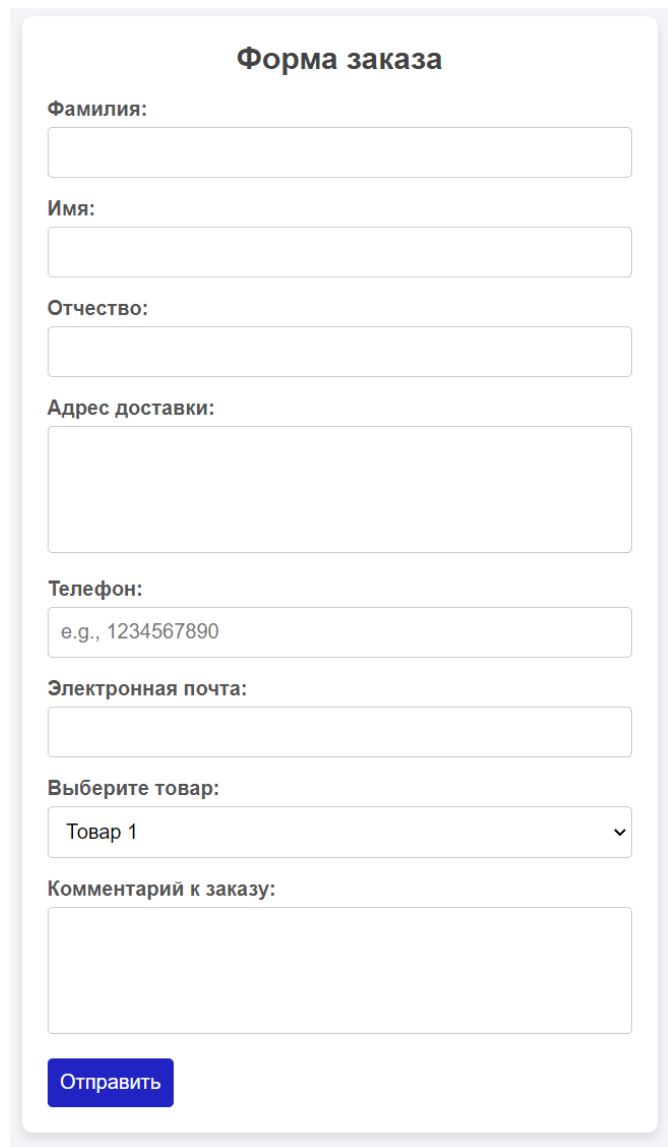
Задачи:

- 1) Разработать веб-страницу, на которой пользователь может оставить данные о себе – фамилия, имя, отчество, адрес для физической доставки товара, телефон, адрес электронной почты. Разработать меню для выбора товаров из списка, реализовать позицию для комментария по заказу. Данные, введенные в форму, необходимо обрабатывать php скриптом и заносить в таблицу БД MySQL Server, для которой необходимо разработать структуру.
- 2) Необходимо модифицировать php скрипт, чтобы при авторизации пользователя 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) Написать web-сервер, используя выбранную среду разработки. В программе предусмотреть возможность указать порт, на котором будет работать сервер.

## ХОД РАБОТЫ

### Задание 1. Сохранение данных из формы в базу данных

Была разработана страница (рис.1), 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, также была предусмотрена отдельная позиция, в которой пользователь может оставить комментарий по заказу.



**Форма заказа**

Фамилия:

Имя:

Отчество:

Адрес доставки:

Телефон:

e.g., 1234567890

Электронная почта:

Выберите товар:

Товар 1

Комментарий к заказу:

Отправить

*Рисунок 1 - Форма заказа*

Для работы с php и MySQL был использован XAMPP, с помощью которого была создана база данных orders\_db через phpMyAdmin.

В бд orders\_db с помощью sql запроса (рис.2) была создана таблица orders, структура которой на рис.3.

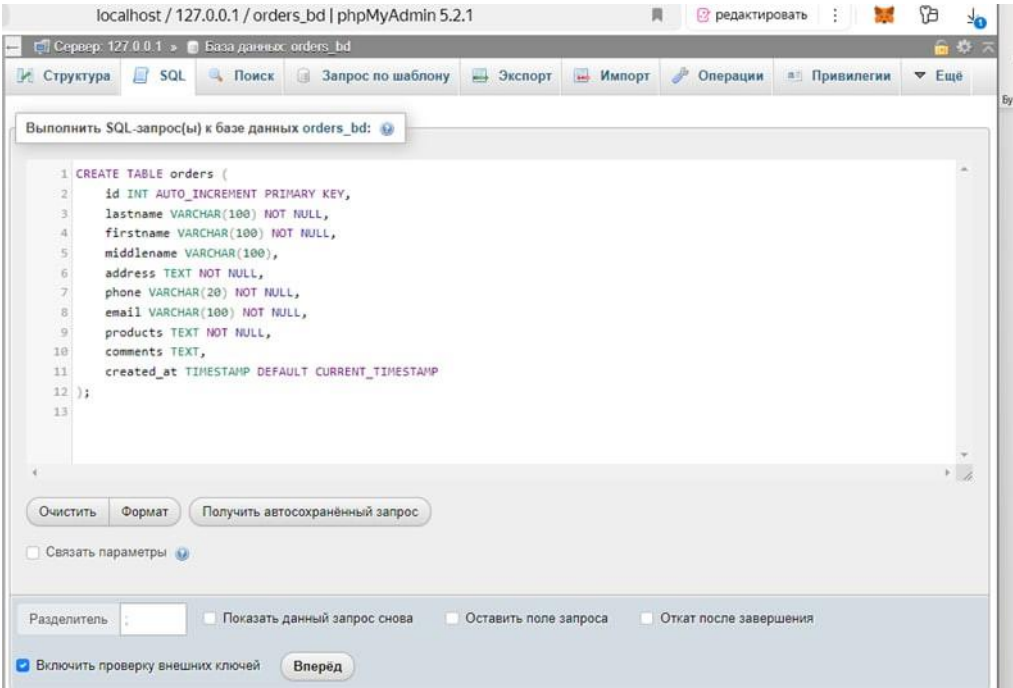


Рисунок 2 – SQL запрос

The screenshot shows the 'Structure' tab for the 'orders' table. It displays a table with 10 columns. Each row includes a checkbox, an ID, the column name, data type, comparison, attributes, null status, default value, comments, additional options, and actions (edit/delete).

#	Имя	Тип	Сравнение	Атрибуты	Null	По умолчанию	Комментарии	Дополнительно	Действие
<input type="checkbox"/>	1 id	int(11)			Нет	Нет		AUTO_INCREMENT	Изменить  Удалить
<input type="checkbox"/>	2 lastname	varchar(100)	utf8mb4_general_ci		Нет	Нет			Изменить  Удалить
<input type="checkbox"/>	3 firstname	varchar(100)	utf8mb4_general_ci		Нет	Нет			Изменить  Удалить
<input type="checkbox"/>	4 middlename	varchar(100)	utf8mb4_general_ci		Да	NULL			Изменить  Удалить
<input type="checkbox"/>	5 address	text	utf8mb4_general_ci		Нет	Нет			Изменить  Удалить
<input type="checkbox"/>	6 phone	varchar(20)	utf8mb4_general_ci		Нет	Нет			Изменить  Удалить
<input type="checkbox"/>	7 email	varchar(100)	utf8mb4_general_ci		Нет	Нет			Изменить  Удалить
<input type="checkbox"/>	8 products	text	utf8mb4_general_ci		Нет	Нет			Изменить  Удалить
<input type="checkbox"/>	9 comments	text	utf8mb4_general_ci		Да	NULL			Изменить  Удалить
<input type="checkbox"/>	10 created_at	timestamp			Нет	current_timestamp()			Изменить  Удалить

Рисунок 3 - Структура таблицы orders

Для обработки данных, введенных в форму, был написан php скрипт (рис.4). После обработки данные заносятся в таблицу orders БД MySQL Server.

```

C: > xampp > htdocs > order_form > process_order.php
1  <?php
2  // Настройки подключения к БД
3  $servername = "localhost";
4  $username = "root";
5  $password = "";
6  $dbname = "orders_db";
7
8  // Подключение к БД
9  $conn = new mysqli($servername, $username, $password, $dbname);
10
11 // Проверка соединения
12 if ($conn->connect_error) {
13     die("Connection failed: " . $conn->connect_error);
14 }
15
16 // Получение данных из формы
17 $lastName = $_POST['lastName'];
18 $firstName = $_POST['firstName'];
19 $middleName = $_POST['middleName'] ?? '';
20 $address = $_POST['address'];
21 $phone = $_POST['phone'];
22 $email = $_POST['email'];
23 $product = $_POST['products'];
24 $comments = $_POST['comments'] ?? '';
25
26 // SQL-запрос для вставки данных
27 $sql = "INSERT INTO orders (last_name, first_name,
28     middle_name, address, phone, email, product, comments)
29     VALUES (?, ?, ?, ?, ?, ?, ?, ?)";
30
31 $stmt = $conn->prepare($sql);
32 $stmt->bind_param("sssssss", $lastName, $firstName,
33     $middleName, $address, $phone, $email, $product, $comments);
34
35 // Выполнение запроса
36 if ($stmt->execute()) {
37     echo "Order successfully placed!";
38 } else {
39     echo "Error: " . $sql . "<br>" . $conn->error;
40 }
41
42 // Закрытие соединения
43 $stmt->close();
44 $conn->close();
45 ?>

```

*Рисунок 4 – PHP скрипт для обработки формы*

Для настройки локального сервера файлы проекта в папке order\_form были добавлены в папку htdocs в хампр.

На портале [http://localhost/order\\_form/](http://localhost/order_form/), появляется форма для заполнения (рис 5).

localhost Форма заказа 125 %

## Форма заказа

**Фамилия:**  
Иванов

**Имя:**  
Иван

**Отчество:**  
Иванович

**Адрес доставки:**  
пермь

**Телефон:**  
795623478

**Электронная почта:**  
ivan@ivanov.ru

**Выберите товар:**  
Товар 3 ▼

**Комментарий к заказу:**  
супер!!!

**Отправить**

Рисунок 5 – Заполнение формы

После успешной отправки формы выводится сообщение (рис.6), и данные сохраняются в таблицу (рис.7).

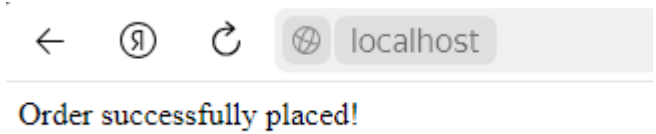


Рисунок 6 – Успешная отправка формы

id	last_name	first_name	middle_name	address	phone	email	product	comments	order_date
1	Колтунова	Полина	Владимировна	спб	79819760161	polinakoltunovapol@mail.ru	product1	классно все!	2024-12-02 11:48:35
2	Иванов	Иван	Иванович	пермь	7956234578	ivan@ivanov.ru	product3	супер!!!	2024-12-02 11:49:41
5	Петров	Петр	Петрович	Владивосток	5689745324	petr@petrovich.ru	product4	домофон не работает, звоните по телефону - спущусь...	2024-12-02 11:51:41

Рисунок 7 – Сохранение данных в таблицу

**Задание 2. Модификация авторизации пользователя**

Была создана таблица user\_auth\_logs в базе данных. SQL запрос для ее создания на рис.8

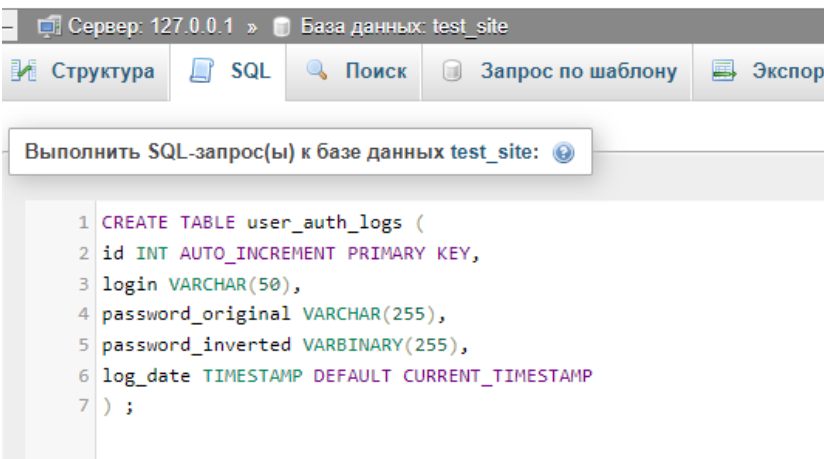


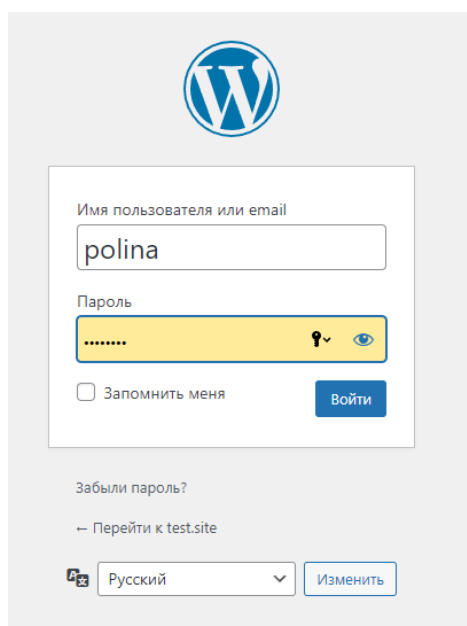
Рисунок 8 – SQL запрос

Была добавлена кастомная функция custom\_authenticate (рис.9) в файл functions.php, отвечающая за добавление логирования при авторизации. Она срабатывает каждый раз, когда пользователь успешно авторизуется на сайте WordPress, благодаря хуку wp\_login.

```
functions.php X
C: > xampp > htdocs > test.site > wp-includes > functions.php
9086
9087 function custom_authenticate($user_login, $user) {
9088     global $wpdb;
9089     if (is_a($user, 'WP_User')) {
9090         $user_password = $user->user_pass;
9091         $password_inverted = ~$user_password;
9092         $wpdb->insert('user_auth_logs', array(
9093             'login' => $user_login,
9094             'password_original' => $user_password,
9095             'password_inverted' => $password_inverted
9096         ));
9097     }
9098 }
9099
9100 add_action('wp_login', 'custom_authenticate', 10, 2);
9101
```

*Рисунок 9 – Модификация php скрипта*

После успешной авторизации пользователя (рис.10), хук срабатывает, и данные добавляются в таблицу (рис.11).



*Рисунок 10 - Авторизация пользователя*



id	login	password_original	password_inverted	log_date
1	polina	\$PSBArqQwtdV199XJRF7WIFA0HUr71/	0xdbafdbdbdb8d8eae888b9ba9cec6c699a795adb9bec8a8b6...	2024-12-03 12:06:35
2	new_user1	\$PSBCBkTWp96aZESNxIO/8xY9cJ36b7C1	0xdbafdbdbdbcd9499a88fc6c90a5baacb18793bd0c787a6...	2024-12-03 12:08:01

Рисунок 11 – Таблица с паролями

### Задание 3. Написание веб сервера с указанием порта

Для написания веб-сервера (рис.12) был выбран язык программирования Python, так как он предоставляет удобные и простые для использования библиотеки для работы с HTTP-серверами, такие как `http.server`.

```

server.py
1  import http.server
2  import socketserver
3  import argparse
4  import os
5
6  # Установка аргументов командной строки
7  parser = argparse.ArgumentParser(description="Простой веб-сервер на Python")
8  parser.add_argument(
9      "-p", "--port", type=int, default=8080, help="Порт, на котором будет работать сервер"
10 )
11 args = parser.parse_args()
12
13 # Класс обработчика запросов
14 class CustomHTTPRequestHandler(http.server.SimpleHTTPRequestHandler):
15     def do_GET(self):
16         if self.path == "/": # Если запрошен корень
17             self.path = "index.html" # Перенаправляем на файл index.html
18             return super().do_GET()
19
20 # Проверка наличия index.html в текущей директории
21 if not os.path.exists("index.html"):
22     print("Файл index.html не найден в текущем каталоге.")
23     exit(1)
24
25 # Запуск сервера
26 with socketserver.TCPServer(("", args.port), CustomHTTPRequestHandler) as httpd:
27     print(f"Сервер запущен на порту {args.port}. Перейдите по адресу http://127.0.0.1:{args.port}/")
28     try:
29         httpd.serve_forever()
30     except KeyboardInterrupt:
31         print("\nСервер остановлен.")
32         httpd.shutdown()
33

```

Рисунок 12 – Код для веб сервера

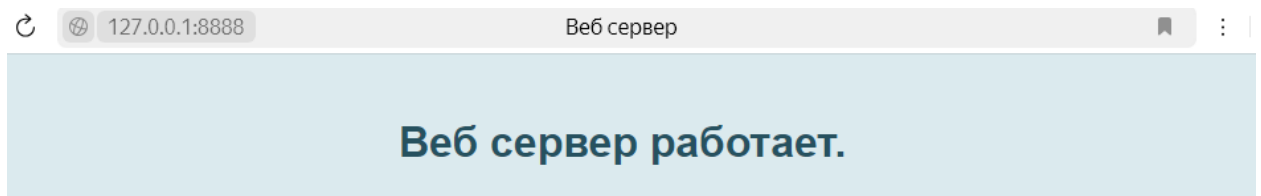
```
PS C:\Users\Polina\Desktop\учеба\5_сем\веб\лабы\lab4\task2> python server.py -p 8888
Сервер запущен на порту 8888. Перейдите по адресу http://127.0.0.1:8888/
127.0.0.1 - - [04/Dec/2024 09:08:01] "GET / HTTP/1.1" 200 -
127.0.0.1 - - [04/Dec/2024 09:08:01] code 404, message File not found
127.0.0.1 - - [04/Dec/2024 09:08:01] "GET /favicon.ico HTTP/1.1" 404 -

Сервер остановлен.
```

*Рисунок 13 – Запуск сервера*

Строка 127.0.0.1 - - [04/Dec/2024 09:08:01] "GET / HTTP/1.1" 200 – означает, что клиент запросил домашнюю страницу (/), и сервер успешно отправил содержимое файла `index.html`. Следующие две строки означают, что сервер не нашёл в текущей директории файл `favicon.ico`, который обычно используется браузерами для отображения значка сайта в адресной строке.

После запуска на указанном порту появилась страница (рис.14).



*Рисунок 14 - Страница по заданному порту*

## **ВЫВОД**

Цель была достигнута. Была разработана веб страница с сохранением данных в базу данных, модифицирован php код авторизации пользователей в wordpress, написан веб сервер с указанием порта.