Санкт-Петербургский Национальный Исследовательский Университет

Информационных Технологий, Механики и Оптики

Факультет Инфокоммуникационных Технологий

**Web-программирование**

**Лабораторная работа 4**

Выполнил

Колсанов Я. И.

Проверила

Марченко Е. В.

Санкт-Петербург, 2024

**Цель:** разработать веб-страницу для отправки формы с занесением в базу данных, в движке wordress модифицировать php скрипт согласно условию задания, написать web-сервер с возможность указания порта, на котором будет работать сервер.

**Задачи:**

* Разработать веб-страницу, на которой пользователь может оставить данные о себе, выбрать товары из списка и оставить комментарий. Данные, введенные в форму, должны обрабатываться php скриптом и заноситься в таблицу БД MySQL Server.
* В движке wordpress модифицировать php cкрипт, чтобы дополнительно логин и пароль записывались в отдельную новую таблицу в БД MySQL. Предусмотреть два варианта записи согласно условию.
* Написать web-сервер с возможностью 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**

Для выполнения задания будем использовать уже установленный MAMP, который позволяет быстро настроить локальную среду для работы с php и MySQL, а также управлять базами данных MySQL через phpMyAdmin.

Для начала создадим таблицу и разработаем ее структуру через phpMyAdmin. Для этого напишем соответствующий SQL запрос.

Изображение выглядит как текст, программное обеспечение, снимок экрана, веб-страница

Автоматически созданное описание

Рисунок 1 – Создание таблицы orders

Изображение выглядит как текст, снимок экрана, число, программное обеспечение

Автоматически созданное описание

Рисунок 2 – Структура таблицы orders

Создадим файл index.html с формой и php файл для обработки заказа и записи в базу данных.

Изображение выглядит как текст, снимок экрана, меню

Автоматически созданное описание

Рисунок 3 – Файл index.html

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 4 – Файл process\_order.php

Добавим эти два файла в папку проекта Applications/MAMP/htdocs/order\_form. По адресу http://localhost/order\_form/index.html открывается форма для заказа, заполняем ее.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Рисунок 5 – Форма для создания заказа

Нажимаем «Отправить» и убеждаемся в успешной работе скрипта.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Рисунок 6 – Успешная отправка формы

Также заказ добавился в таблицу orders в базе данных orders\_db.

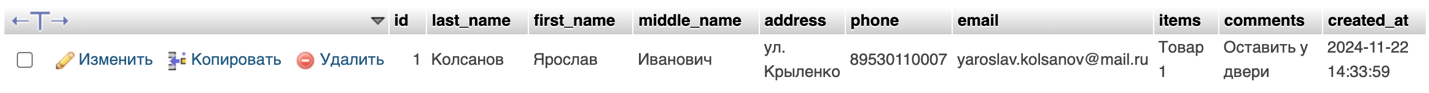


Рисунок 7 – Таблица orders

**Задание 2**

Для выполнения задания был использован проект предыдущей лабораторной работы. В файл htdocs/wordpress/wp-includes/pluggable.php были внесены изменения в функции wp\_authenticate, чтобы логин, пароль в исходном виде и инвертированный пароль записывались в отдельную новую таблицу wp\_users\_credentials в БД MySQL test-site\_db. Для этого при правильной авторизации (то есть $is\_valid == True) cоздается описанная выше таблица wp\_users\_credentials. Измененный код представлен ниже.

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 8 – Измененный код в функции wp\_authenticate файла pluggable.php

Запускаем сервер и переходим на страницу авторизации test.site/wp-login.php.

Изображение выглядит как текст, программное обеспечение, снимок экрана, веб-страница

Автоматически созданное описание

Рисунок 9 – Страница авторизации

Входим под созданными раннее логином и паролем, а затем проверяем базу данных. Таблица wp\_users\_credentials успешно создана с добавлением строки с логином и паролем в двух вариантах записи.

Изображение выглядит как снимок экрана, текст, программное обеспечение, веб-страница

Автоматически созданное описание

Рисунок 10 – Новая таблица

**Задание 3**

Для написания web-сервера с возможностью указания порта, на котором сервер будет работать, был выбран язык Python и среда разработки Visual Studio Code. Использованы встроенные библиотеки http.server, socketserver и os. http.server.SimpleHTTPRequestHandler используется для обработки HTTP-запросов. Метод do\_GET переопределён, чтобы возвращать index.html при обращении к корневому пути. socketserver.TCPServer создаёт сервер, прослушивающий указанный порт. httpd.serve\_forever() запускает сервер и позволяет обрабатывать запросы на протяжении всего времени работы. Ниже представлен весь код создания сервера.

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 11 – Файл server.py

Также в файле index.html была создана простая веб-страница.

Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание

Рисунок 12 – Файл index.html

Запускаем сервер, вводим порт 888 и открываем http://127.0.0.1:888/.

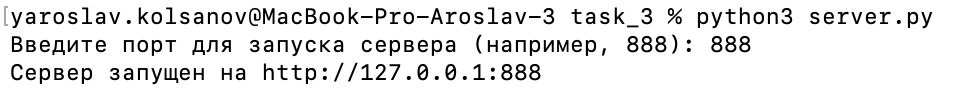


Рисунок 13 – Запуск сервера в терминале

Изображение выглядит как текст, Шрифт, линия, снимок экрана

Автоматически созданное описание

Рисунок 14 – Отображение файла index.html в браузере

**Вывод**

В результате выполнения лабораторной работы была разработана веб-страница для отправки формы с занесением в базу данных, в движке wordress модифицирован php скрипт согласно условию задания, а также написан web-сервер на языке Python с возможностью указания порта, на котором будет работать сервер.