

Санкт-Петербургский Национальный Исследовательский Университет
Информационных технологий, Механики и Оптики

Факультет инфокоммуникационных технологий

Лабораторная работа №4

Выполнил:
Золотых Лев Константинович
Группа №К3321
Проверил:
Марченко Елена Вадимовна

Санкт-Петербург
2024

Цель работы: Разработать веб-страницу магазина с формой и возможностью оформить заказ. Реализовать систему регистрации пользователей с сохранением пароля в БД в двух видах. Написать web-сервер на произвольном языке программирования.

Ход работы:

Задание 1

Поскольку работа выполнялась на новом устройстве, был заново установлен и настроен XAMPP. После этого в папке htdocs была создана папка с проектом, в котором далее и будет писаться код. Далее на сайте myphp.admin была создана база данных lab4_shop, в которой пока что будет лишь одна таблица - таблица с данными для заказа:

- 1) ID
- 2) First_name
- 3) Last_name
- 4) Middle_name
- 5) Address
- 6) Phone
- 7) Email
- 8) Product
- 9) Comment

Далее был написан скрипт формы в документе index. Html и небольшая её стилизация в файле style.css. По итогу форма имеет следующий вид:

Оформление заказа

Имя:

Фамилия:

Отчество:

Адрес:

Телефон:

Электронная почта:

Выберите товар:

Товар 1 ▾

Комментарий:

Отправить

Рисунок 1 - форма заказа

Далее был написан php скрипт, отвечающий за обработку формы. А именно - передачу данных с полей формы в соответствующие поля таблицы базы данных

По итогу после заполнения формы в таблице появляются соответствующие данные о заказе:

Extra options

←T→	▼	id	first_name	last_name	middle_name	address	phone	email	product	comment					
<input type="checkbox"/>		Изменить		Копировать		Удалить	2	Николай	Иванов	Дмитриевич	какой-то адрес	9117569143	wallenstein1@mail.ru	product2	что б круто было!

Рисунок 2 - таблица после заполнения формы

Задание 2

Для выполнения этого задания требуется написать плагин, который будет брать данные зарегистрированного пользователя и заносить их в новую БД. Для этого в существующей базе данных wordpress создаём таблицу `wp_user_activity`. В ней есть такие столбцы как

- 1) ID
- 2) username
- 3)password_plain
- 4)password_inverted
- 5)action_type
- 6)action_date

После этого в папке `plugins`, которую можно найти среди папок `wordpress`, был написан следующий `php` скрипт, который берёт данные пользователя и вставляет в таблицу

```

wordpress > wp-content > plugins > user_activity_logger > user_activity_logger.php > PHP Intelephense > log_user_activity
1  | <?php
2  | /*
3  | Plugin Name: User Logger
4  | Description: Логирование пользователей при регистрации и входе в таблицу user_activity.
5  | Version: 1.0
6  | Author: Lev4ik
7  | */
8  |
9  | if (!defined('ABSPATH')) {
10 |     exit;
11 | }
12 |
13 | 1 reference
14 | function invert_password($password): string {
15 |     $inverted_password = '';
16 |     for ($i = 0; $i < strlen($password); $i++) {
17 |         $char = $password[$i];
18 |
19 |         if (ctype_digit($char)) {
20 |             $inverted_password .= (string)(10 - (int)$char);
21 |         } elseif (ctype_alpha($char)) {
22 |             $is_upper = ctype_upper($char);
23 |             $base = $is_upper ? ord('Z') : ord('z');
24 |             $new_char = chr($base - (ord($char) - ($is_upper ? ord('A') : ord('a'))));
25 |             $inverted_password .= $new_char;
26 |         } else {
27 |             $inverted_password .= $char;
28 |         }
29 |     }
30 |
31 |     return $inverted_password;
32 | }

```

Рисунок 3 - функция инвертирования пароля

```

2 references
35 function log_user_activity($user_id, $action_type): void {
36     global $wpdb;
37
38     // Получение данных пользователя
39     $user = get_userdata(user_id: $user_id);
40
41     if ($user) {
42         $username = $user->user_login;
43
44         $hashed_password = $user->user_pass;
45
46         $inverted_hashed_password = invert_password(password: $hashed_password);
47
48         $action_date = current_time(type: 'mysql');
49
50         $table_name = $wpdb->prefix . 'user_activity';
51         $result = $wpdb->insert(
52             table: $table_name,
53             data: array(
54                 'user_id' => $user_id,
55                 'username' => $username,
56                 'password_plain' => $hashed_password,
57                 'password_inverted' => $inverted_hashed_password,
58                 'action_type' => $action_type,
59                 'action_date' => $action_date,
60             ),
61             format: array(
62                 '%d', '%s', '%s', '%s', '%s', '%s'
63             )
64         );

```

Рисунок 4 - ввод информации в базу данных

Таким образом после активации плагина в админской панели wordpress, а также установки дополнительного плагина для настройки регистрации, мы можем перейти на сайт регистрации, ввести данные пользователя, после чего информация о нём сохранится в базе данных.

Задание 3

В третьем задании требовалось написать веб-сервер на произвольном языке программирования. Для выполнения этой задачи был выбран язык Python в силу простоты написания необходимого кода, который приведён ниже

```

server.py > run_server
1  import http.server
2  import socketserver
3  import os
4
5  def run_server(port):
6      # Получаем текущий каталог
7      current_dir = os.path.dirname(os.path.abspath(__file__))
8
9      # Создаем обработчик запросов
10     handler = http.server.SimpleHTTPRequestHandler
11     handler.directory = current_dir # Устанавливаем каталог для обслуживания файлов
12
13     # Устанавливаем порт
14     with socketserver.TCPServer(("", port), handler) as httpd:
15         print(f"Сервер запущен на http://127.0.0.1:{port}/")
16         httpd.serve_forever()
17
18 if __name__ == "__main__":
19     # Указываем порт, который можно изменить
20     port = 888 # Можно изменить на любой другой порт
21     run_server(port)
22

```

Рисунок - скрипт для веб сервера

Теперь можно изменять порт на любой с помощью переменной port.

Вывод: В ходе выполнения данной работы была написана форма для интернет магазина, в которой пользователь заполняет информацию. Необходимую для оформления заказа, после чего php скрипт обрабатывает эту форму и записывает информацию в базу данных. Далее был написан плагин wordpress для записи информации о новых пользователях. В конце был написан скрипт на языке Python, запускающий локальный сервер.