Министерство образования и науки РФ

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра Информационные технологии и автоматизированные системы

Базы данных

Лабораторная работа № 3

Тема: «Вывод SQL запросов на страницу браузера»

Выполнил:

студент группы РИС-23-2б

Борисов Н.А.

Проверил:

доцент кафедры ИТАС

Петренко А.А.

г. Пермь – 2024

# Цель работы

Сделать веб-сайт для удобного и наглядного осуществления запросов к серверу базы данные MySQL.

# Задачи работы

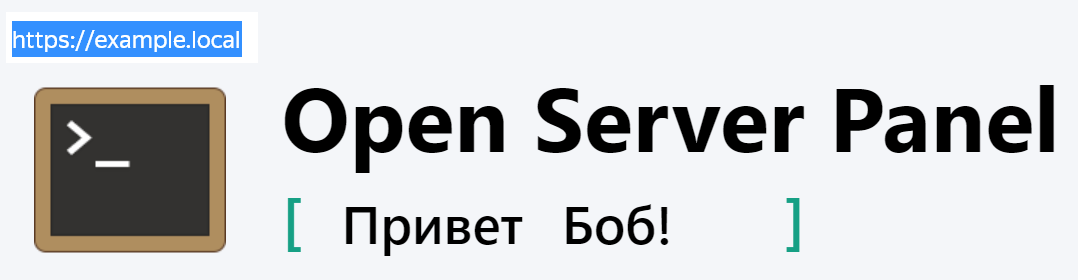
1. Скачать и установить Open Server Panel, выбрав только нужные компоненты для установки.
2. Изучить как происходит создание нового проекта.
3. Рассмотреть построение страницы сайта с помощью HTML.
4. Выбрать необходимые классы и методы из PHP для запросов к базе данных и динамической обработке страницы сайта.
5. Внедрить JavaScript для обновления конкретного элемента, чтобы уйти от создания новой страницы.

# Этапы выполнения

# Установка Open Panel Server

Скачать портативную программную среду для веб-разработки можно с сайта создателей программы [[1]](#СайтРазработчиковOSPanel) или с любого другого сайта, сравнив после контрольную сумму файла с данными, предоставленными на сайте самих разработчиков.

На данный момент времени версия дистрибутива 6.0.0. При процедуре установки следует выбрать лишь нужные компоненты для дальнейшей работы. Достаточно выбрать панель управления, Nginx 1.26 и PHP 8.1. По завершению установки и запуске панели, программа появится в системной трее и будет доступна. Теперь можно включить ранее установленные компоненты в разделе с модулями и проверить работоспособность тестовых сайтов.

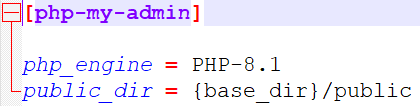


# Создание нового проекта

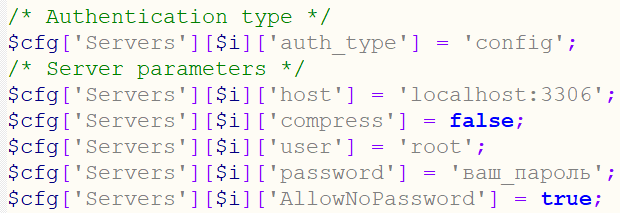
По использованию Open Server Panel существует своя документация [[2]](#ДокументацияOSPanel). Для примера можно попробовать установить PhpMyAdmin [[3]](#СайтРазработчиковPhpMyAdmin). Это веб-приложение с открытым кодом, написанное на языке PHP и представляющее собой веб-интерфейс для администрирования СУБД MySQL.

Переходим в каталог с проектами OSPanel (по умолчанию папка home) и создаём новую папку с названием, например, php-my-admin. В ней создаём папку .osp и папку public.

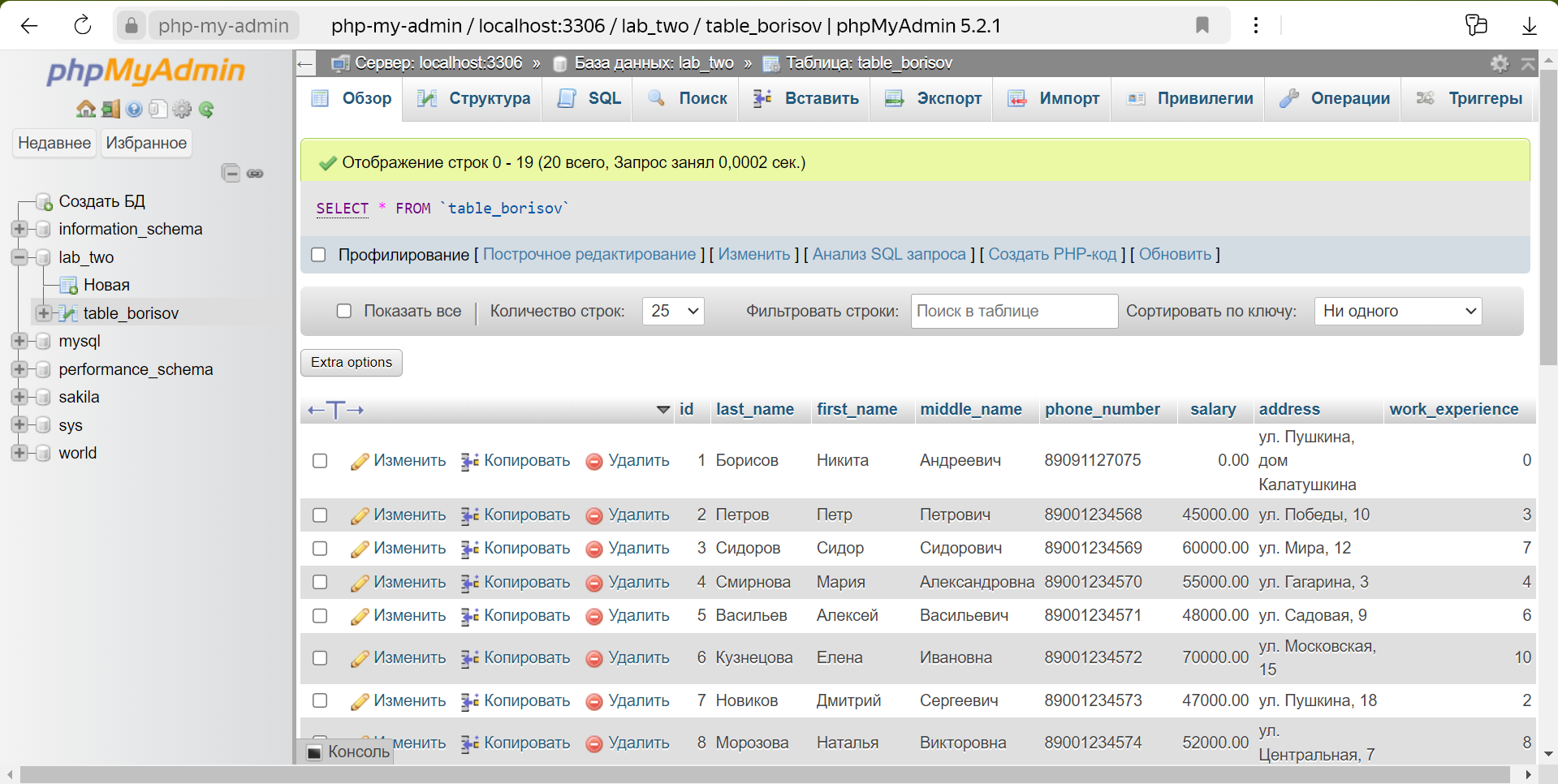
В папке .osp должена быть прописана конфигурация проекта в файле project.ini. Она должна включать в себя домен сайта в квадратных скобках, указание модуля php и путь до папки public.



Скачиваем архив с веб-приложением PhpMyAdmin и распаковываем его в папку public. Осталось только настроить его. Находим среди файлов config.sample.inc.php, убираем .sample и открываем файл для редактирования содержимого. Далее нужно найти среди исходных команд блок с параметрами сервера и прописать следующее:



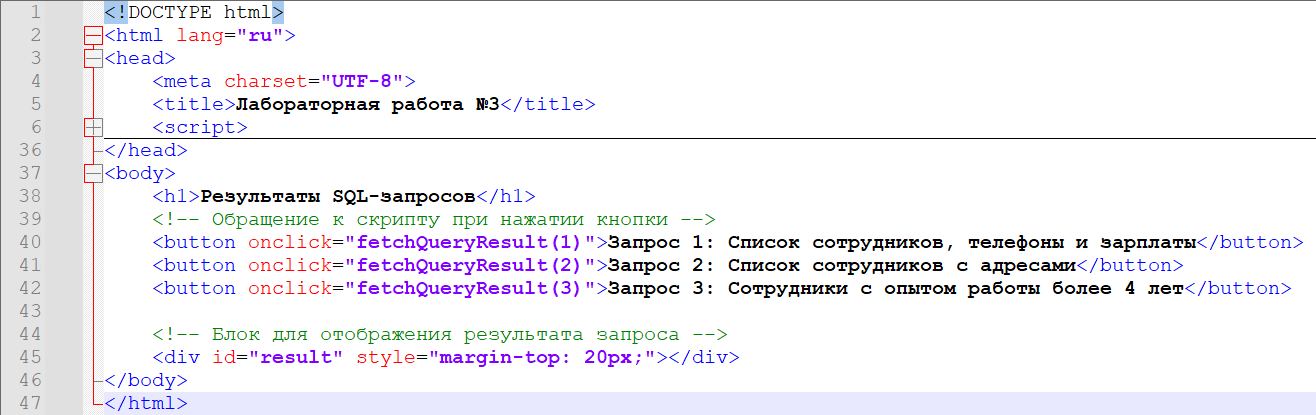
Теперь можно сохранить изменения, закрыть файл конфигурации, перезапустить OSPanel и проверить, правильно ли работает веб-приложение, получается ли подключиться к серверу локальной базы данных MySQL.



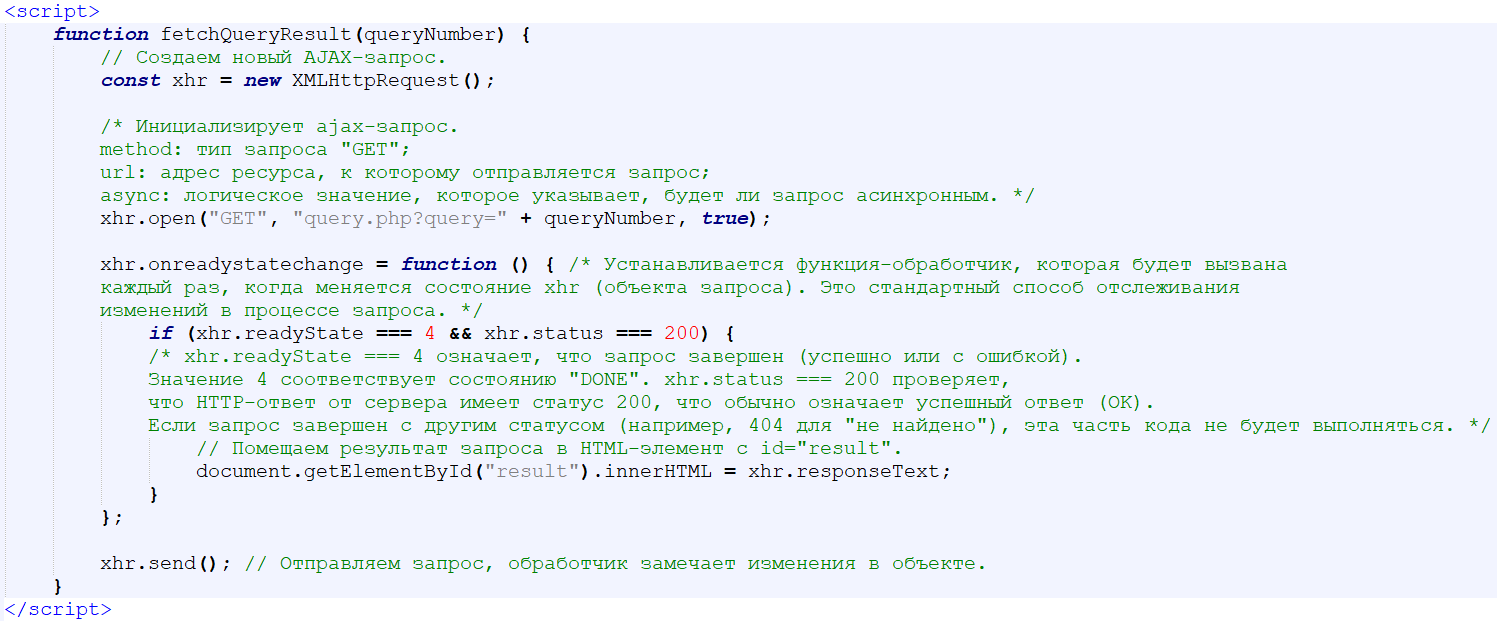
# Разработка веб-приложения

Также как и в случае с созданием пробного проекта, понадобится создать новую папку lab-three и прописать конфигурационный файл project.ini. Нужно создать файл index.html – куда будет попадать пользователь после обращения по домену сайта, а также query.php для создания таблицы результата.

Начнём с разбора html-файла.



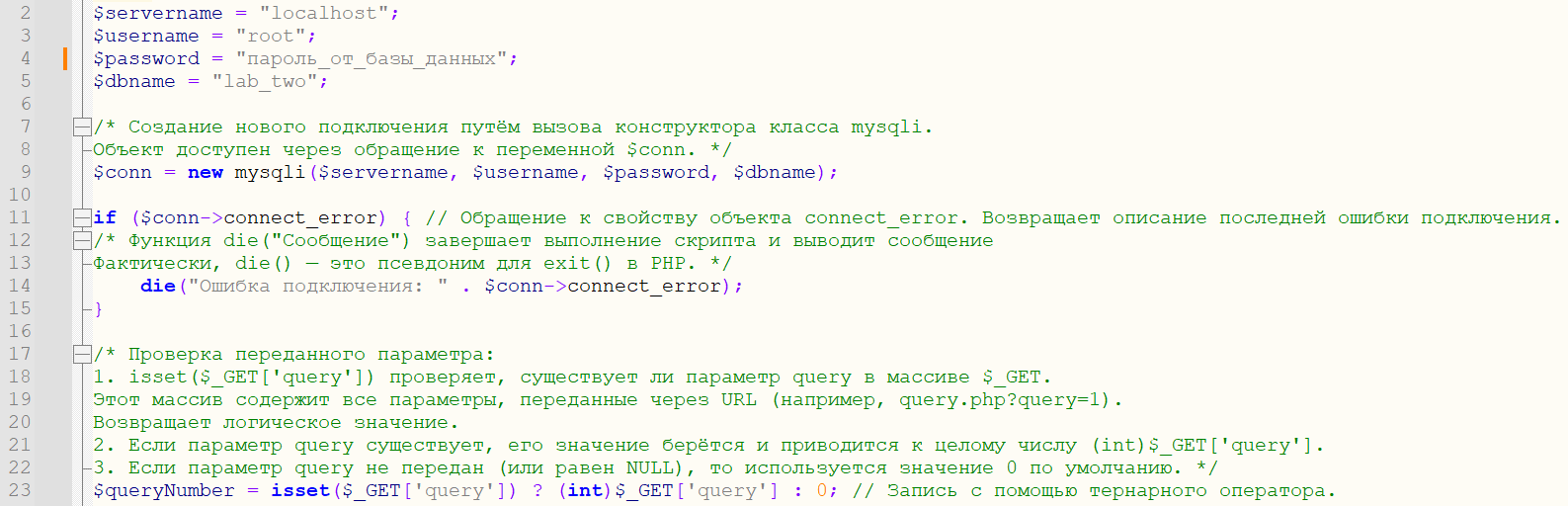
Запрос осуществляется нажатием одной из трёх кнопок на странице сайта. Они описаны между метками body в разметке сайта. При нажатии пользователем любой из кнопок [[4]](#СозданиеКнопокВHTML) запускается функция JavaScript, в которую передаётся числовой параметр – номер запроса. Описание скрипта идёт прямо в html-файле в блоке script.



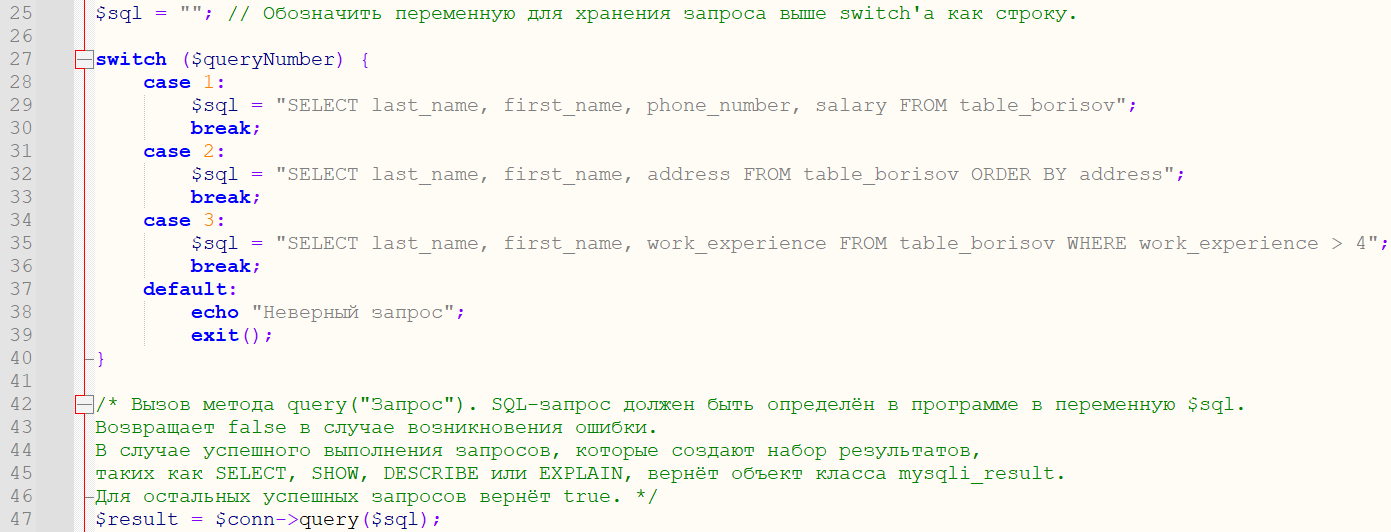
Перейдём к более детальному разбору логики действий. Главная цель использования JS – перезапись конкретного элемента страницы, а не создание новой, пользуясь только PHP. Здесь используется AJAX-запрос [[5]](#AJAXЗапросыИXMLHttpRequest). Ajax представляет технологию для отправки запросов к серверу из клиентского кода JavaScript без перезагрузки страницы. Сам термин расшифровывается как Asynchronous JavaScript And XML. Одним из способов для отправки ajax-запросов является применение объекта XMLHttpRequest.

Как видно на рисунке с кодом скрипта, создаётся новый объект и инициализируется. Все действия происходят после выполнения запроса с помощью метода send(). После этого прописанная функция-обработчик замечает изменения в объекте, и далее следует проверка успешности выполнения действий. Если запрос был успешен, блок с идентификатором result будет перезаписан.

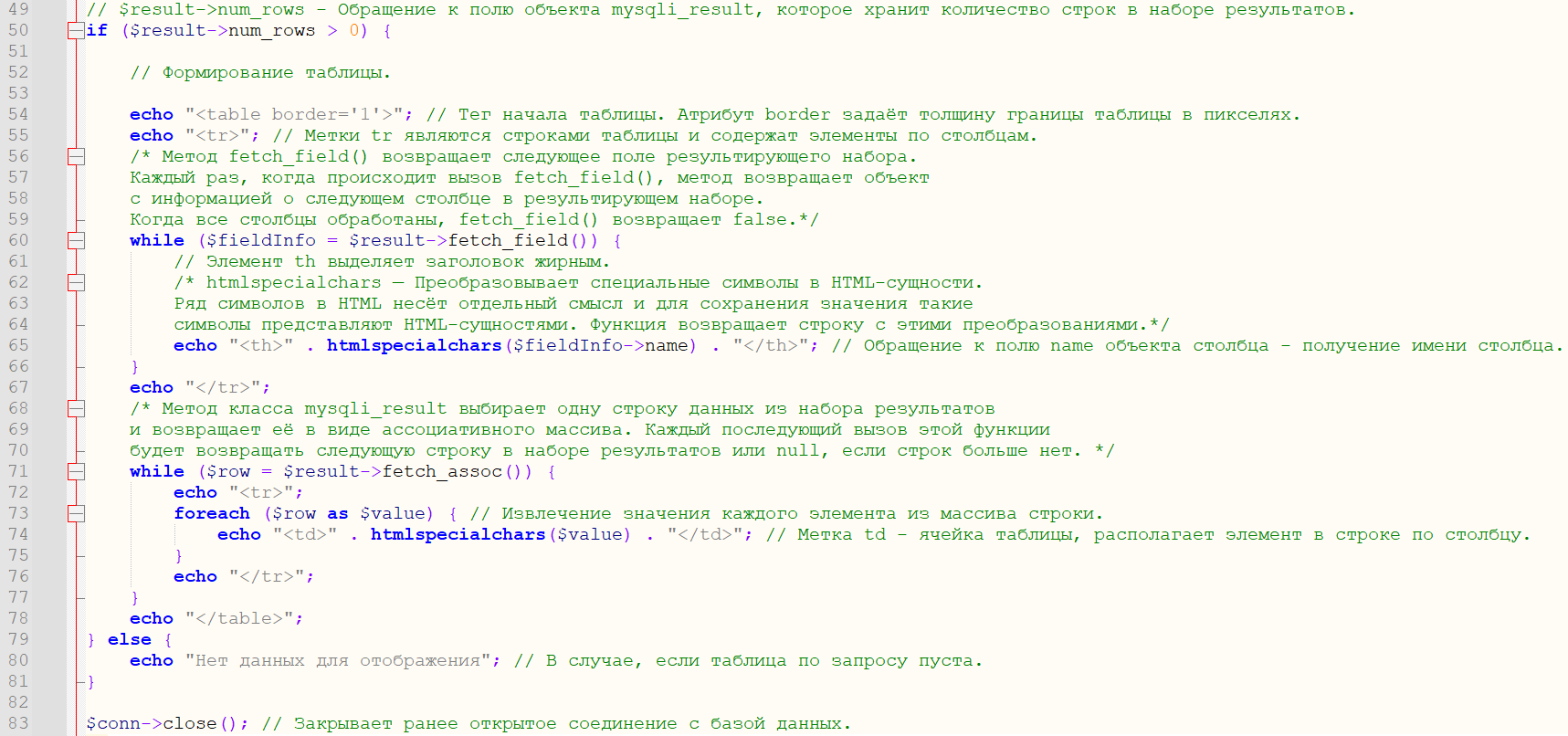
Вернёмся к инициализации объекта. В параметре URL прописана ссылка, которая одновременно передаёт номер запроса и запускает PHP-скрипт на сервере. Перейдём к детальному и пошаговому анализу кода.



Создаётся новый объект класса mysqli [[6]](#Классmysqli). Параметры конструктора описаны выше для удобства их редактирования. Идёт проверка на осуществление подключения до базы данных. Скрипт завершает своё выполнение, когда случается ошибка, и выдаёт информацию о ней. Далее идёт очень интересная запись, которая используется для корректного получения переданного в скрипт значения номера запроса. Осуществляется проверка параметра query в массиве $\_GET, используется тернарный оператор [[7]](#PHPТернарныйОператор) и приведение к типу int. Если параметр не передан, результат операции будет иметь значение 0.



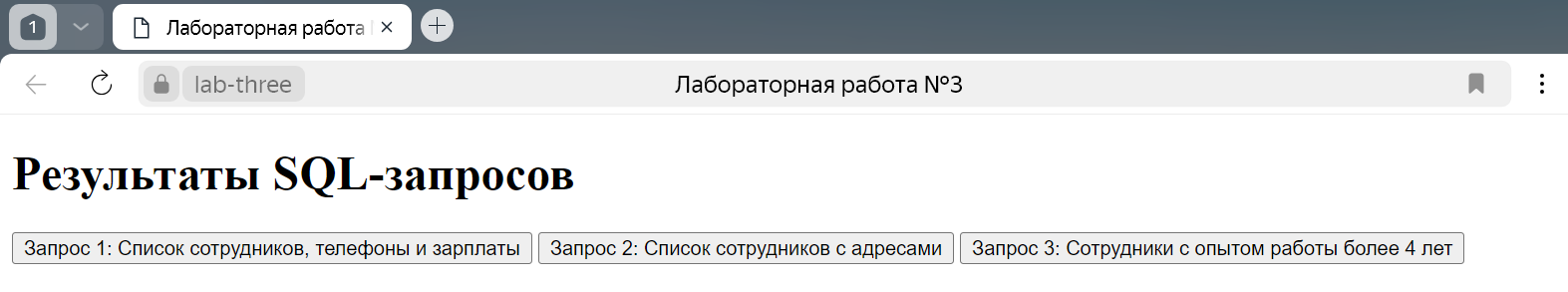
С помощью конструкции switch от полученного номера запроса формируется строка, содержащая SQL-запрос для базы данных. Если номер запроса не существует, скрипт завершает своё выполнение. Если всё успешно, происходит запрос на сервер базы данных и формируется ответ как объект типа mysqli\_result [[8]](#Классmysqli_result).

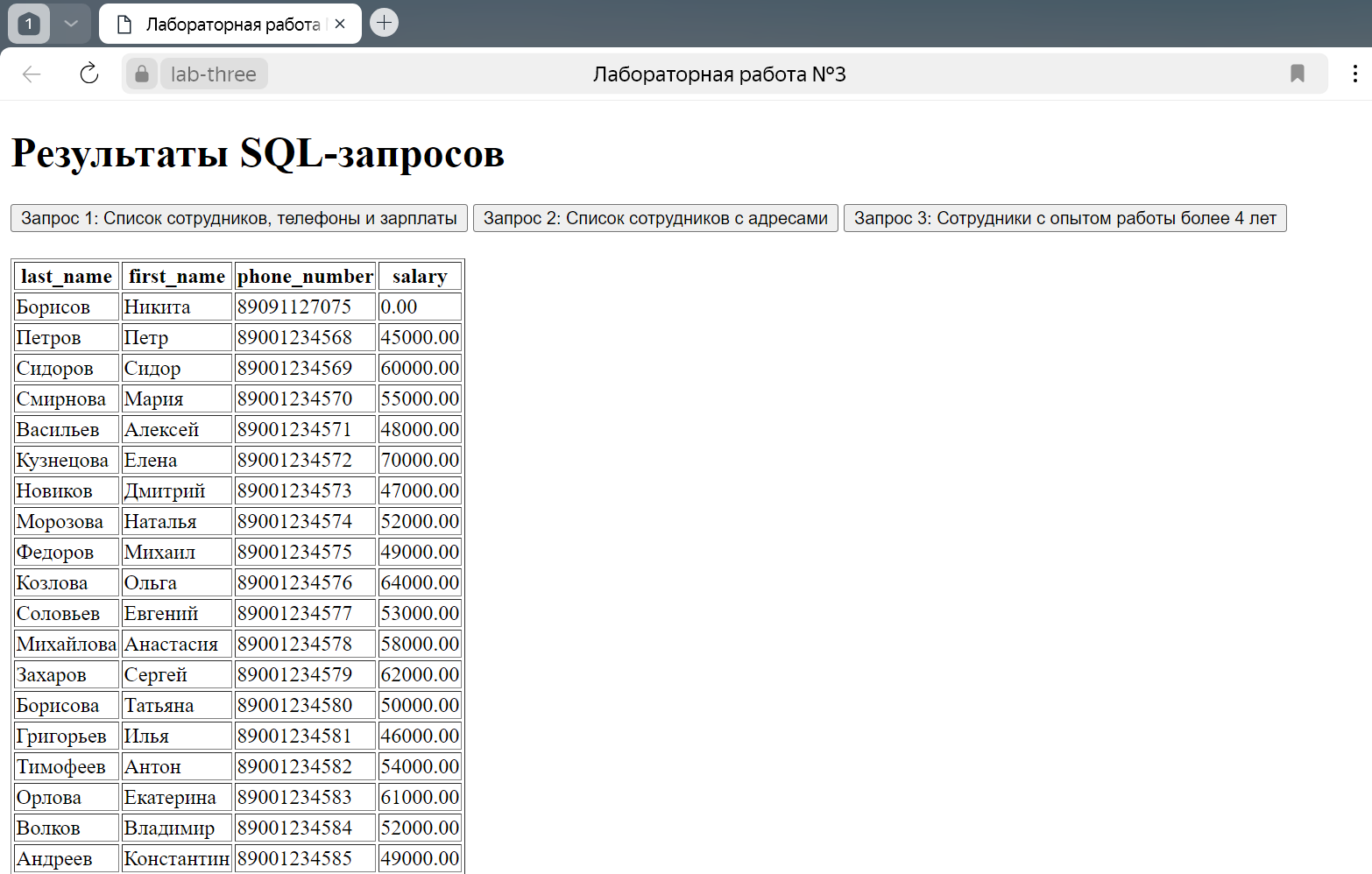


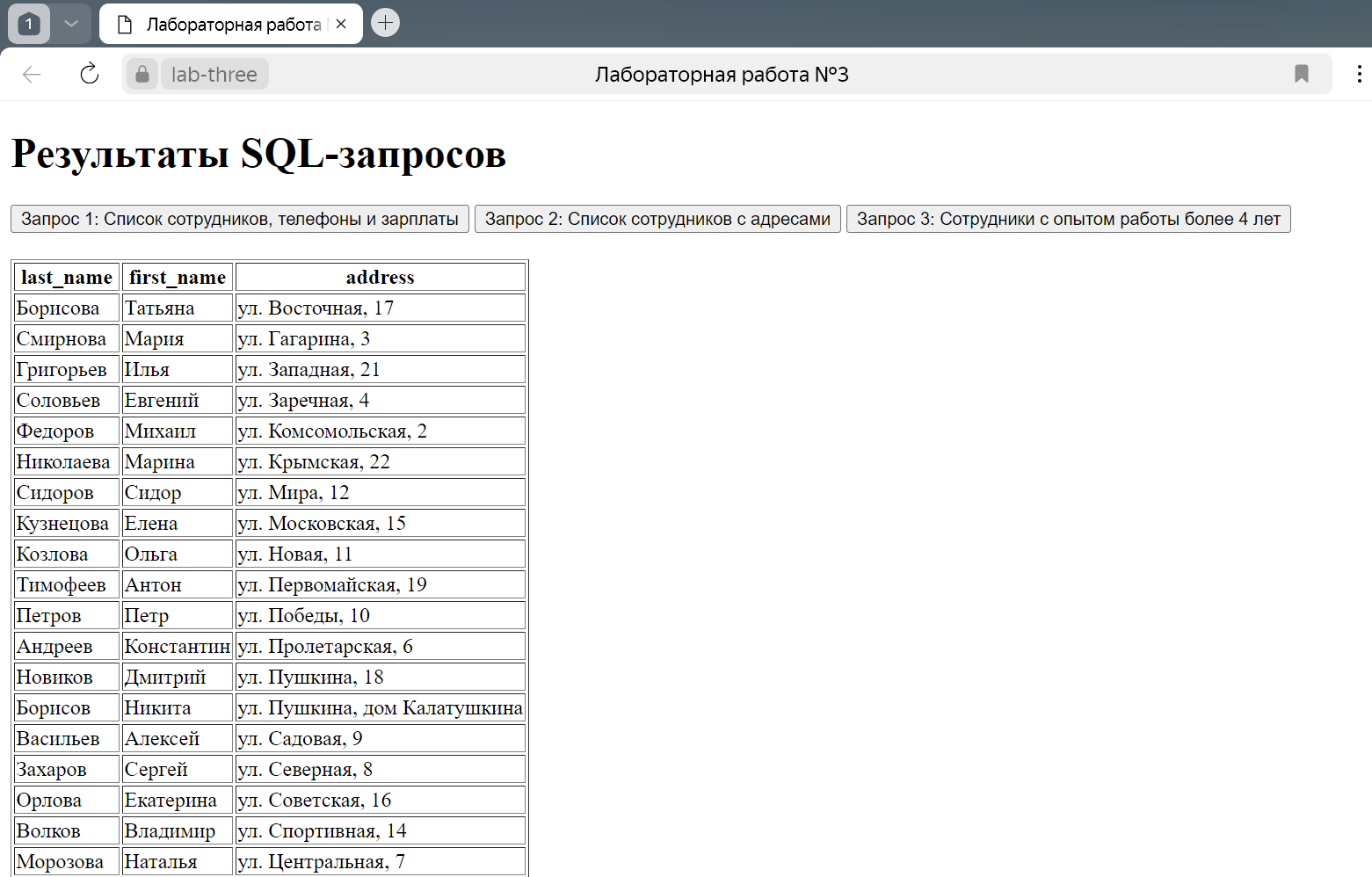
Следующий блок кода рассчитан на построение таблицы [[9]](#СозданиеТаблицВHTML). Первым делом проверяется, не является ли таблица пустой. Если же она имеет хотя бы одну строку, идёт генерация результата как HTML код с помощью команды echo. Для таблицы используются соответствующие метки, такие как table, tr, th и td. Про их значение можно узнать из комментариев в коде.

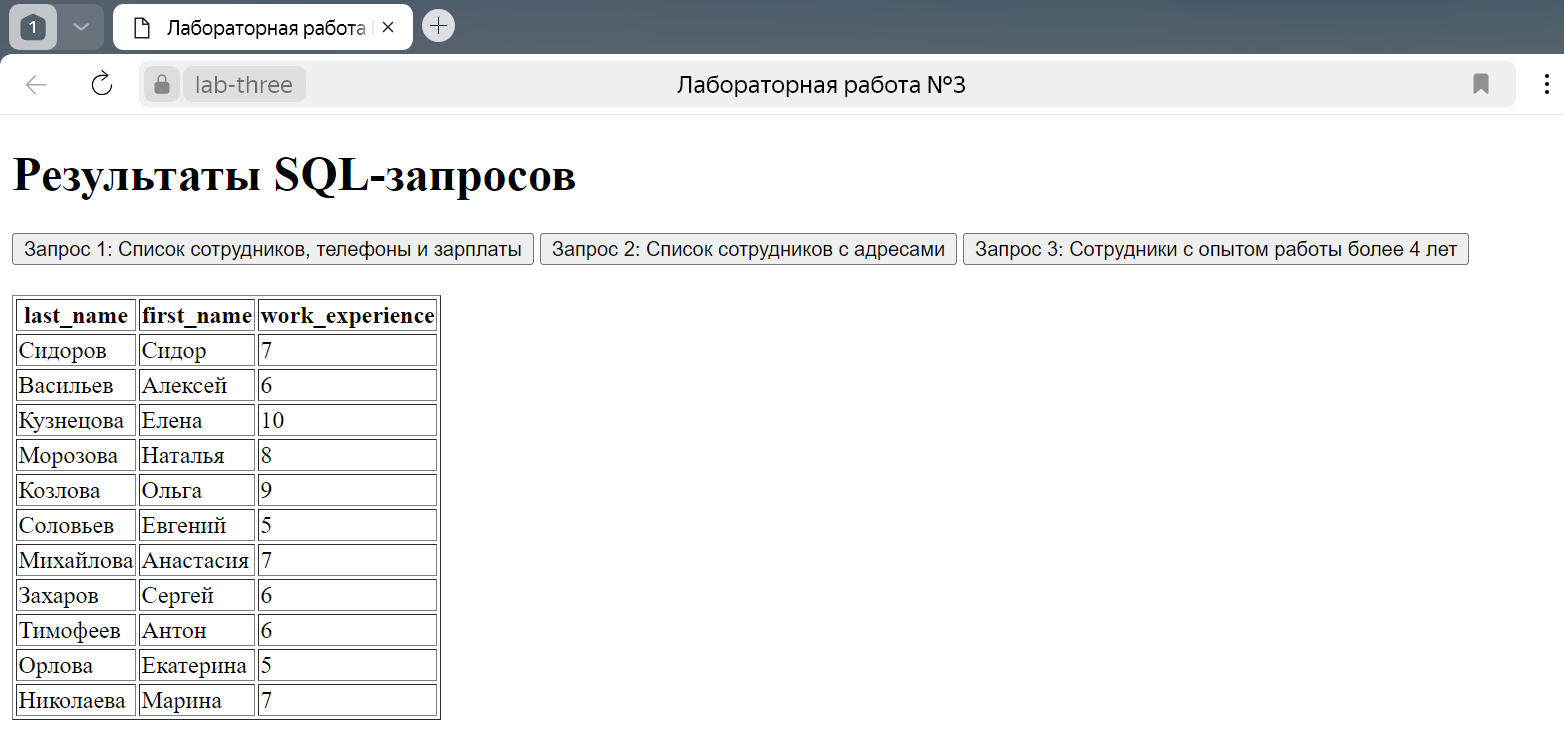
Сначала происходит вывод названий столбцов с помощью цикла while и обращения к методу fetch\_field(). Затем уже идёт заполнение строк в столбцы. Каждая строка выделяется с помощью метода fetch\_assoc() и представляет собой ассоциативный массив. С помощью цикла foreach происходит извлечение из массива каждого элемента строки – он будет помещён в метки ячейки таблицы.

# Результаты разработки веб-приложения.









# Заключение

В ходе выполненной работы была представлена возможность познакомиться с разработкой веб-приложений, а также обработка запросов с помощью подключения к базе данных через скрипт.

Были так или иначе затронутые такие языки веб-программирования как HTML, PHP и JavaScript. Каждый используется для своих целей, но в сумме с их помощью можно делать интересные интерактивные сайты.

Было создано веб-приложение, к которому можно обратиться по домену, перенаправляющему на локальный сервер Open Server Panel.

Open Server Panel – мощный инструмент для разработки веб-сайта любой сложности, предоставляющий удобное использование модулей.

# Список используемой литературы

1. Сайт разработчиков Open Server Panel / [Электронный ресурс] // ospanel.io [сайт]. — URL: <https://ospanel.io/download> (дата обращения 28.10.2024).
2. Документация по OSPanel / [Электронный ресурс] // github.com [сайт]. — URL: <https://github.com/OSPanel/OpenServerPanel/wiki/Документация> (дата обращения 28.10.2024).
3. Сайт разработчиков PhpMyAdmin / [Электронный ресурс] // www.phpmyadmin.net [сайт]. — URL: <https://www.phpmyadmin.net> (дата обращения 28.10.2024).
4. Создание кнопок в HTML / [Электронный ресурс] // metanit.com [сайт]. — URL: <https://metanit.com/web/html5/3.3.php> (дата обращения 28.10.2024).
5. AJAX-запросы и XMLHttpRequest / [Электронный ресурс] // metanit.com [сайт]. — URL: <https://metanit.com/web/javascript/13.1.php> (дата обращения 28.10.2024).
6. Класс mysqli / [Электронный ресурс] // www.php.net [сайт]. — URL: <https://www.php.net/manual/ru/class.mysqli.php> (дата обращения 28.10.2024).
7. PHP: Тернарный оператор / [Электронный ресурс] // puzzleweb.ru [сайт]. — URL: <https://puzzleweb.ru/php/17_if_else2.php> (дата обращения 28.10.2024).
8. Класс mysqli\_result / [Электронный ресурс] // www.php.net [сайт]. — URL: <https://www.php.net/manual/ru/class.mysqli-result.php> (дата обращения 28.10.2024).
9. Создание таблиц в HTML / [Электронный ресурс] // metanit.com [сайт]. — URL: <https://metanit.com/web/html5/2.8.php> (дата обращения 28.10.2024).