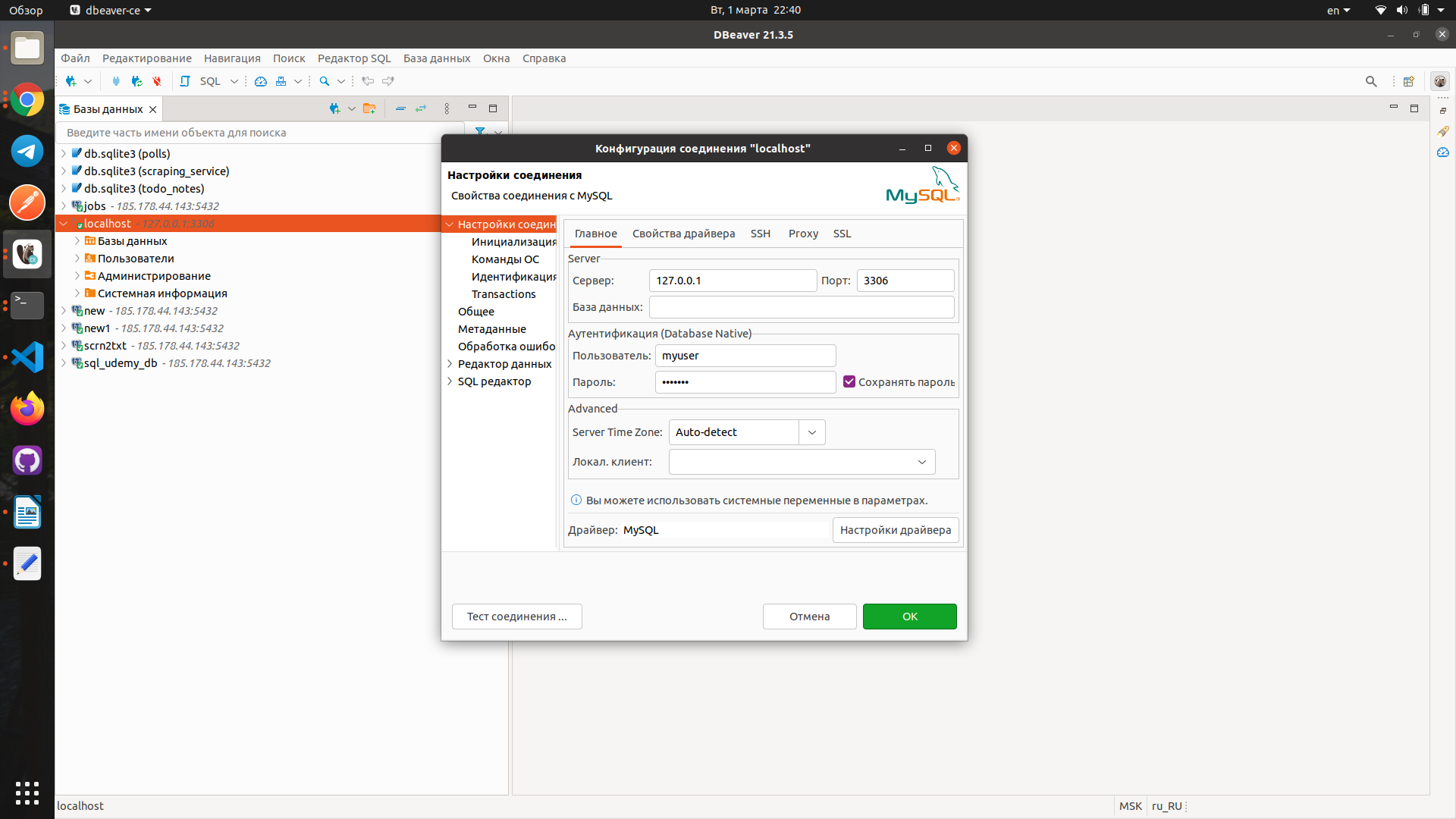
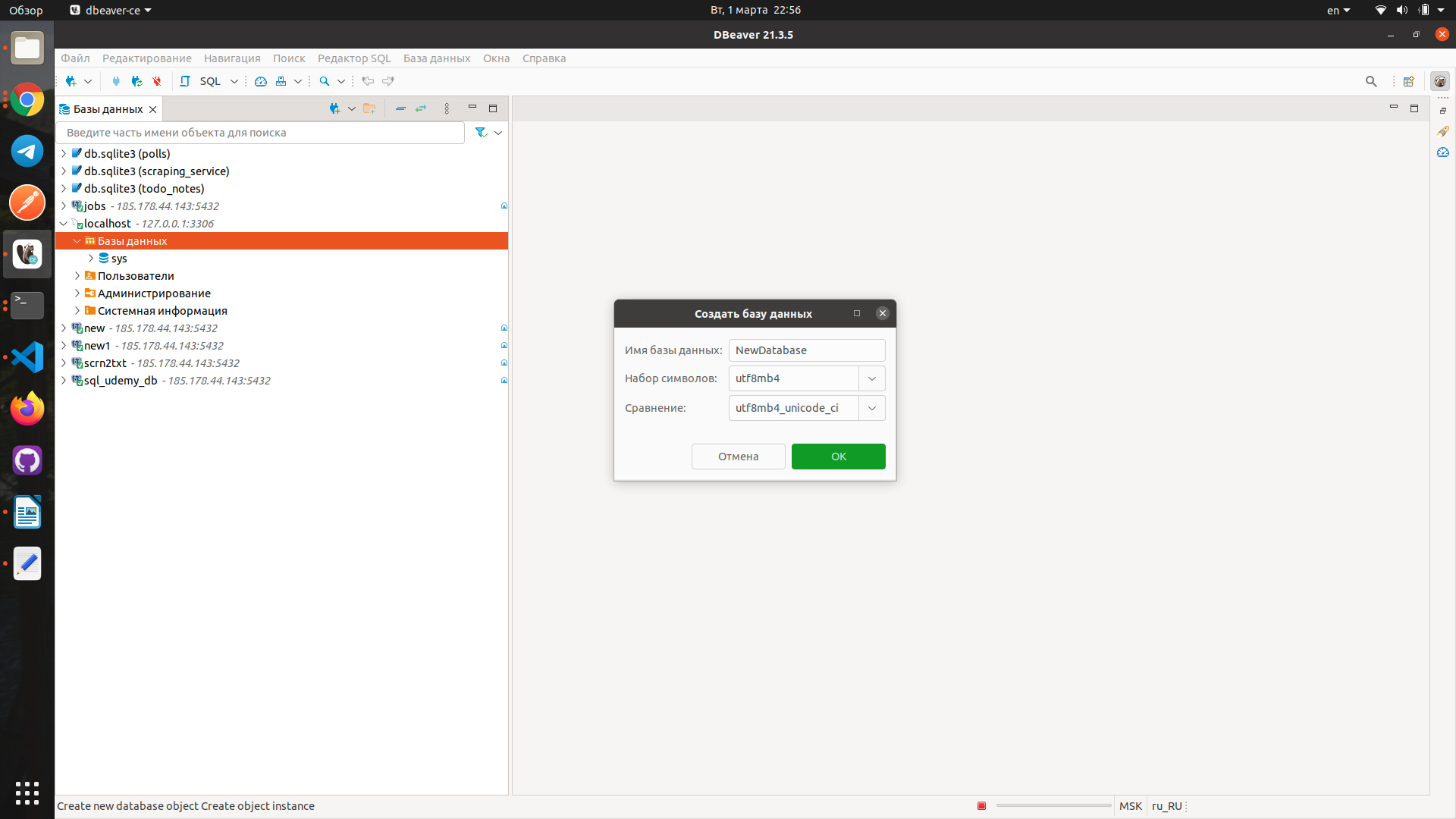


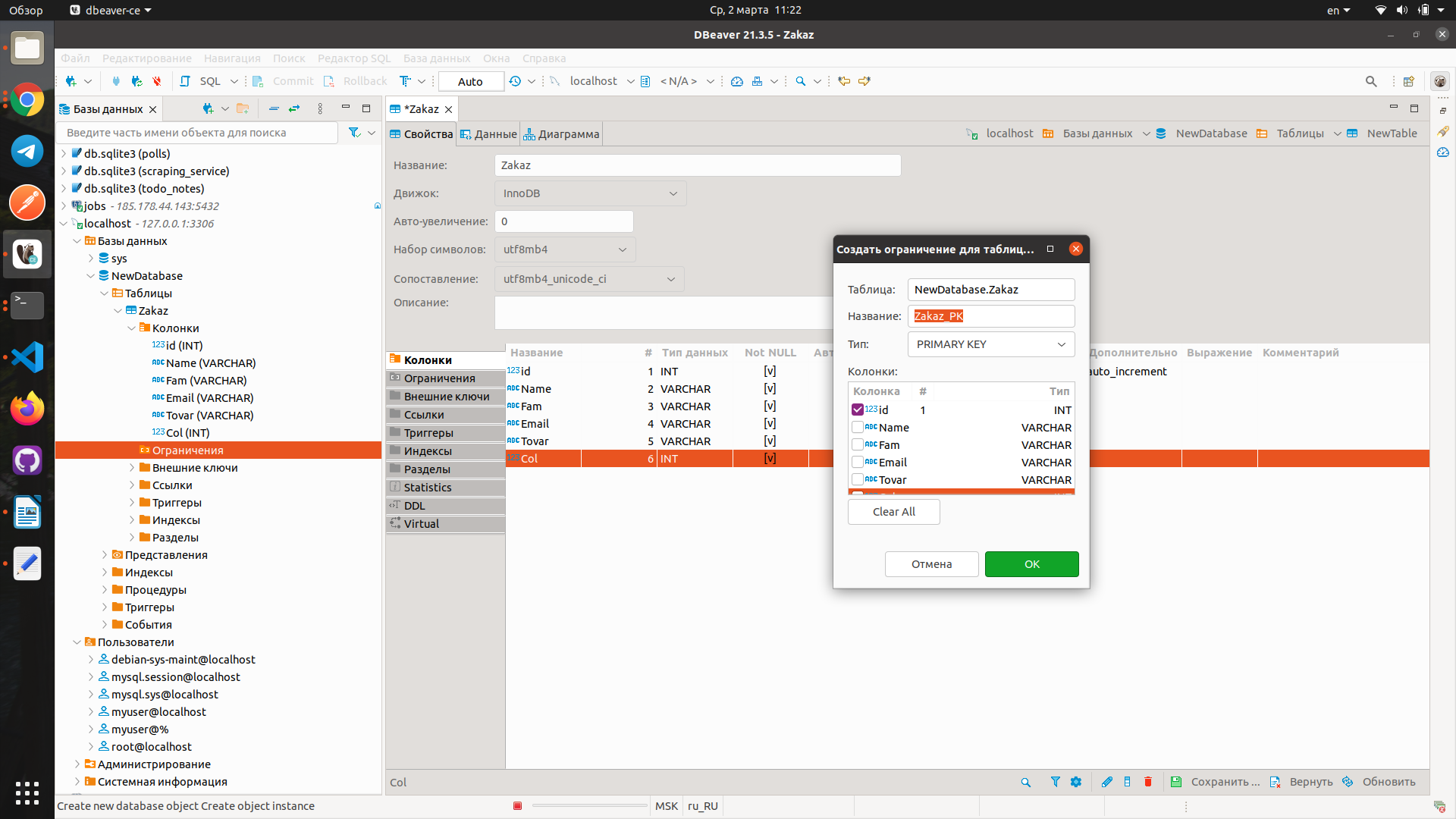
Для выполнения задания используем docker контейнер, запущенный с помощью образа `stanislavsiviskiy/gulpproject` - предварительно в него установили PHP 7.2 и MySQL(server + client).



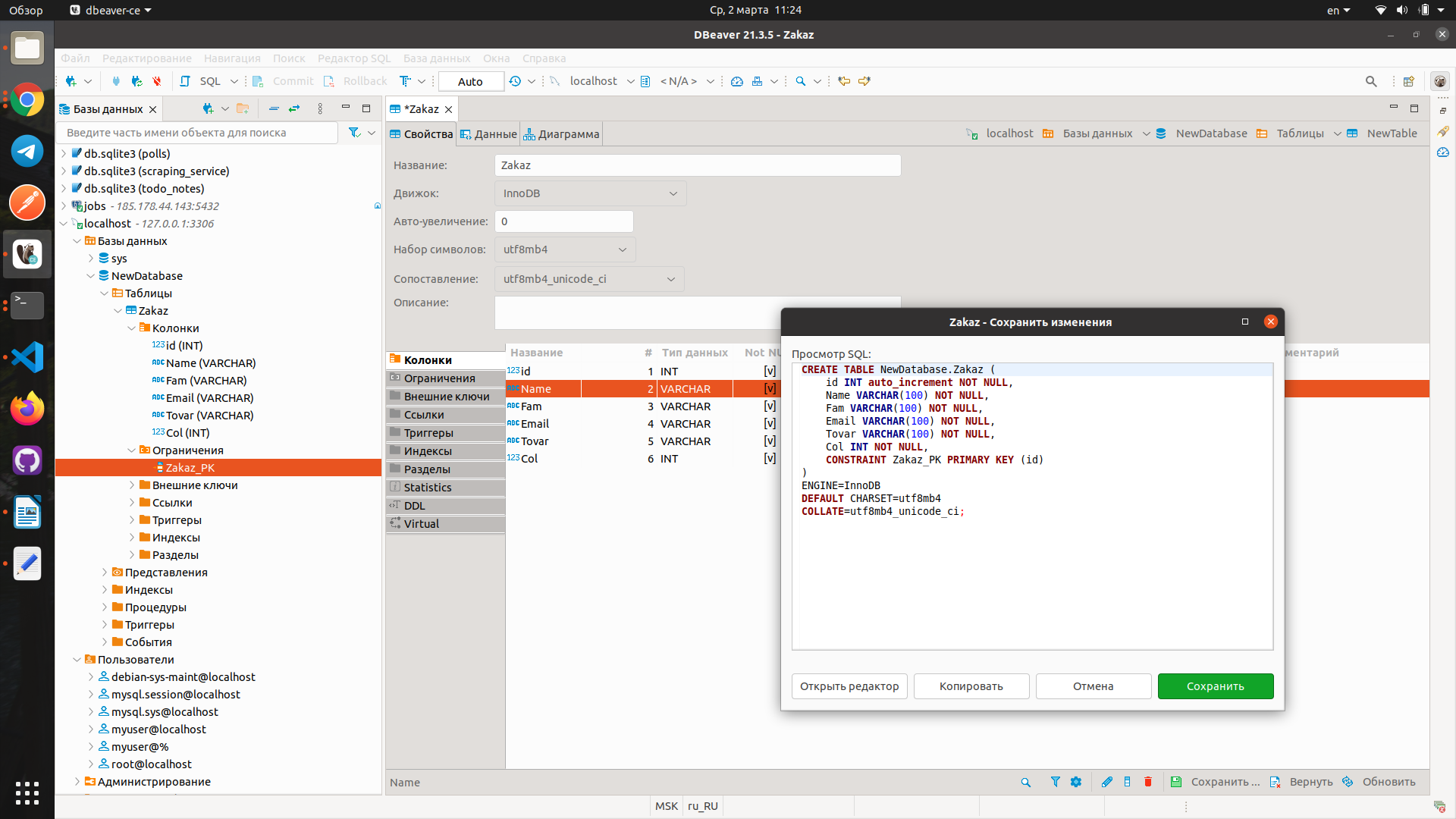
Вместо phpMyAdmin используем dbeaver. Настроили соединение с сервером MySQL, запущенным в нашем docker контейнере - при запуске контейнера был прокинут порт 3306 для соединения.



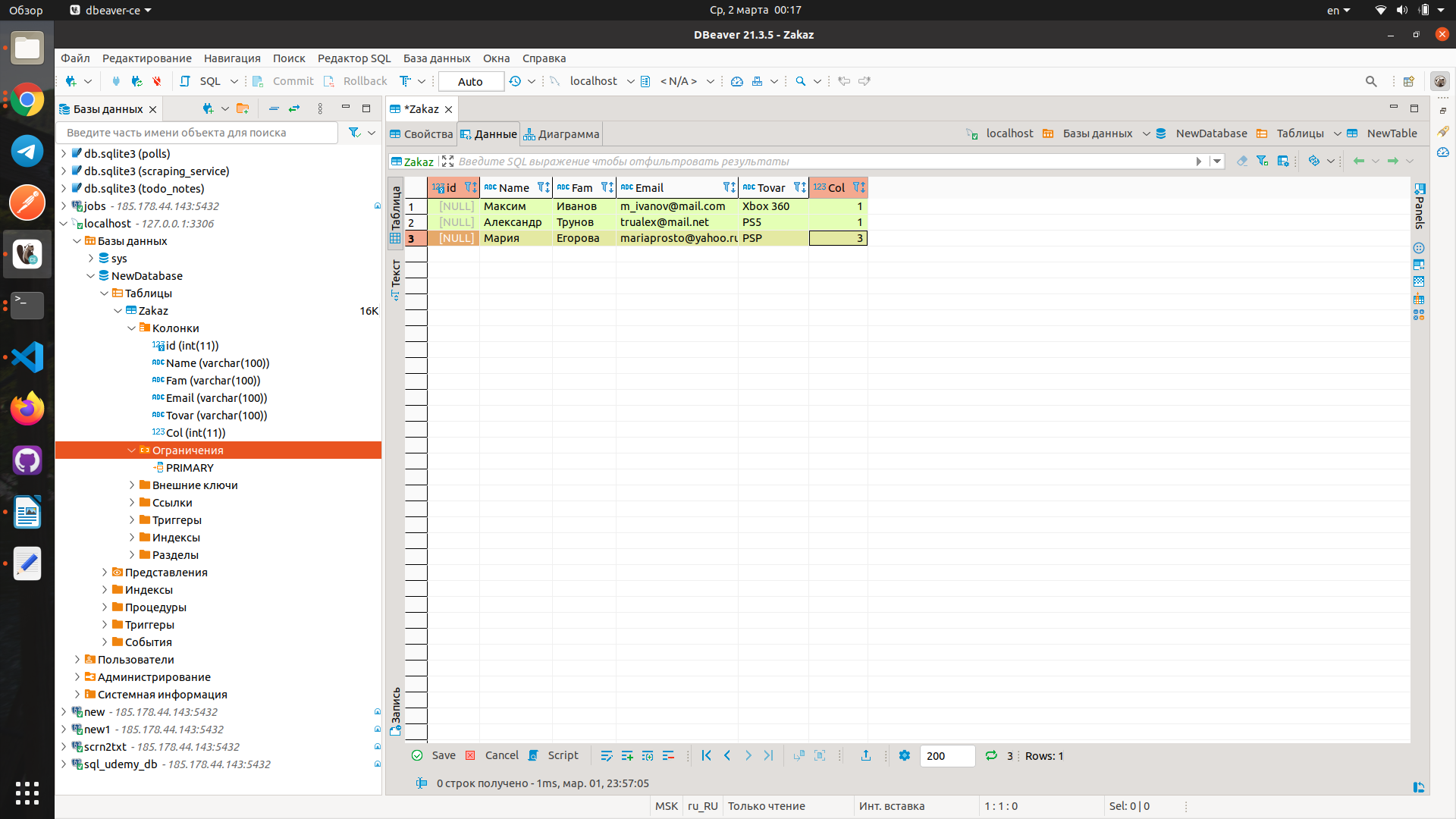
Правой кнопкой мыши выбираем в контекстном меню «создать объект `База данных`», указываем имя `NewDatabase` и выбираем нужную кодировку.



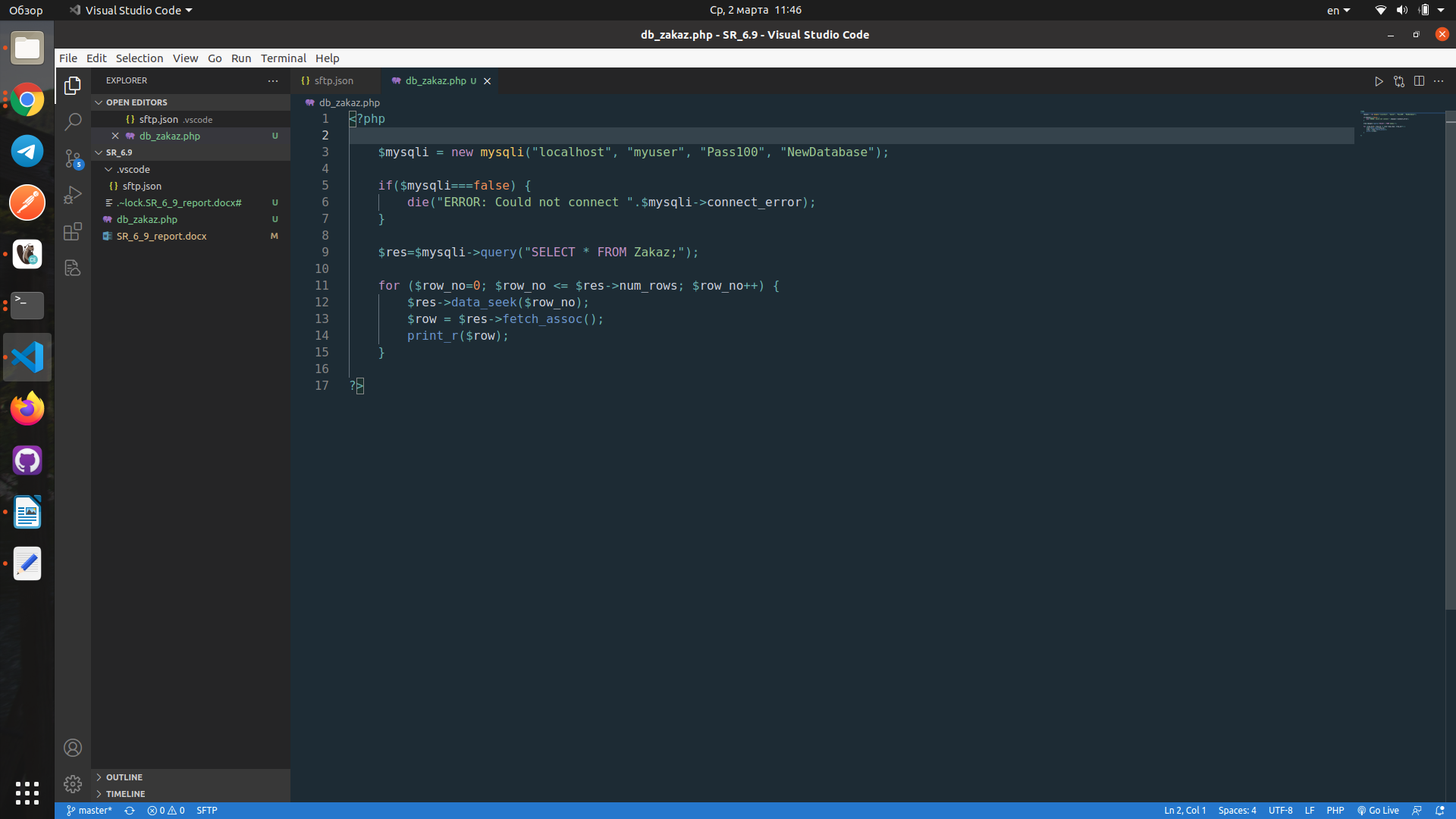
Далее в БД создаем таблицу `Zakaz`, в ней создаем колонки - одну за другой. Устанавливаем первичный ключ на колонку `id`.



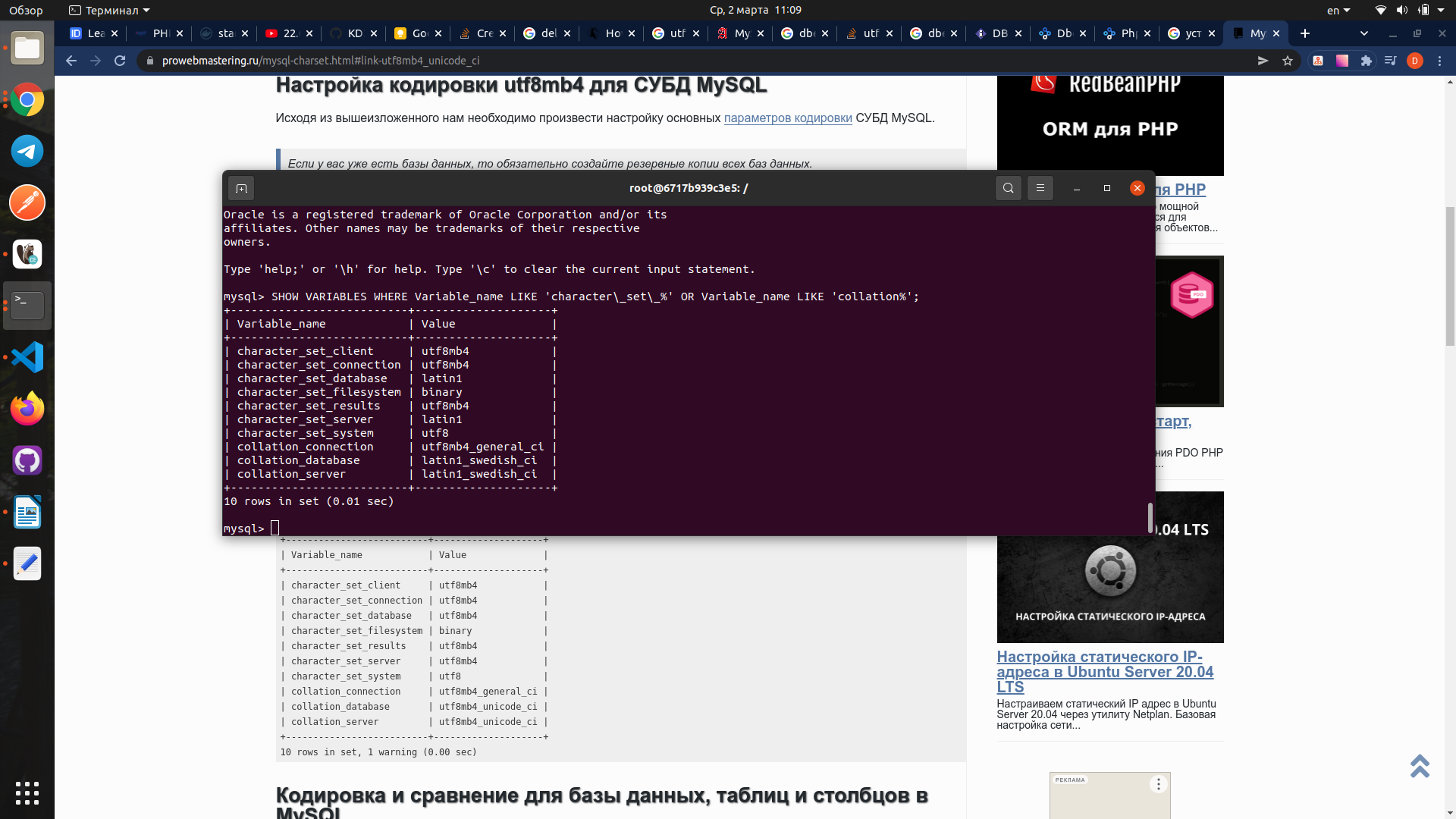
Заполняем названия для всех колонок и сохраняем таблицу — видим SQL-скрипт, который сработает «под капотом» для создания данной таблицы.



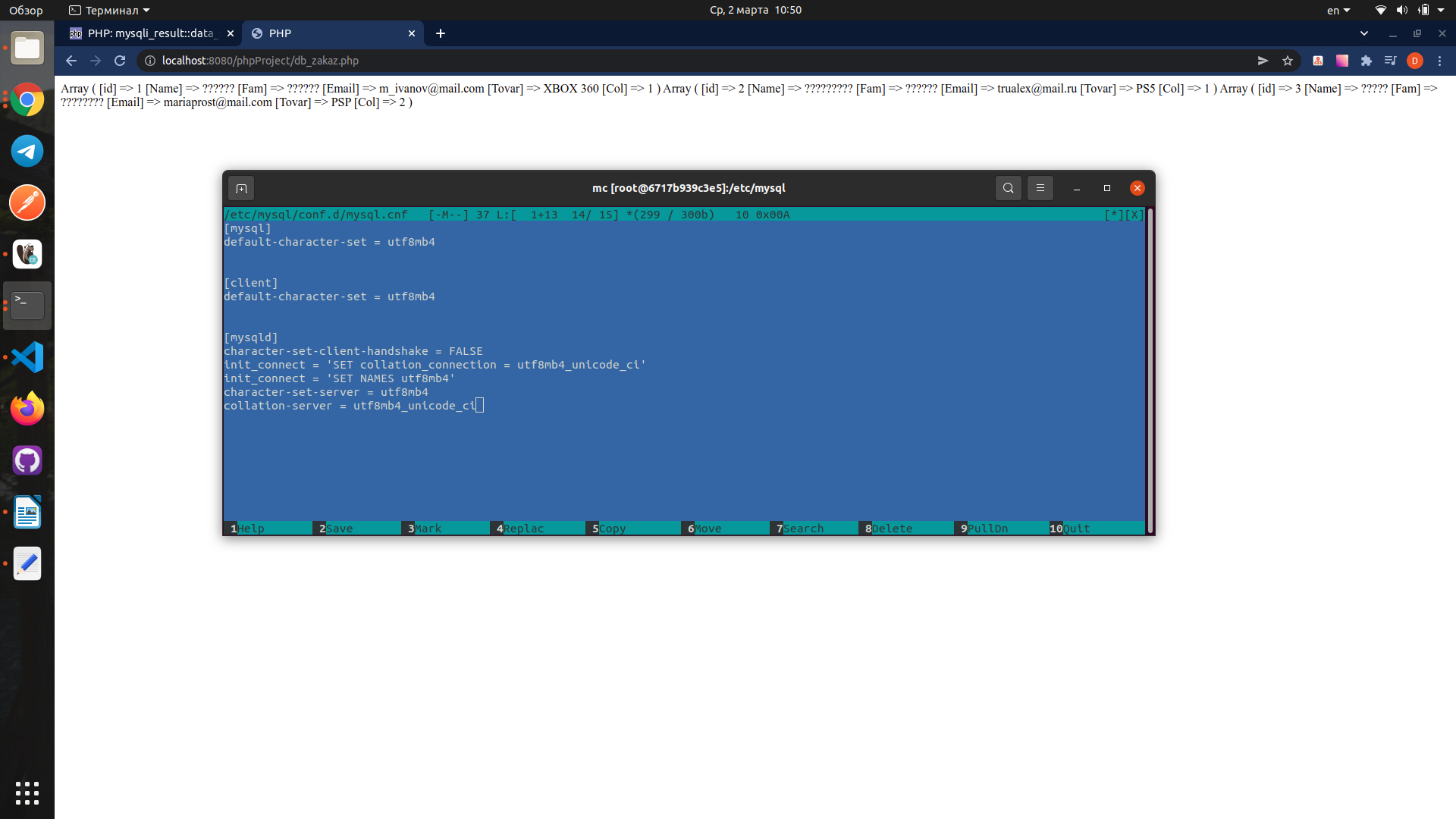
Внесли три записи в таблицу. После сохранения таблицы в БД — колонка `id` заполнится автоматически, так как мы оставили ее значение пустым для всех записей таблицы.



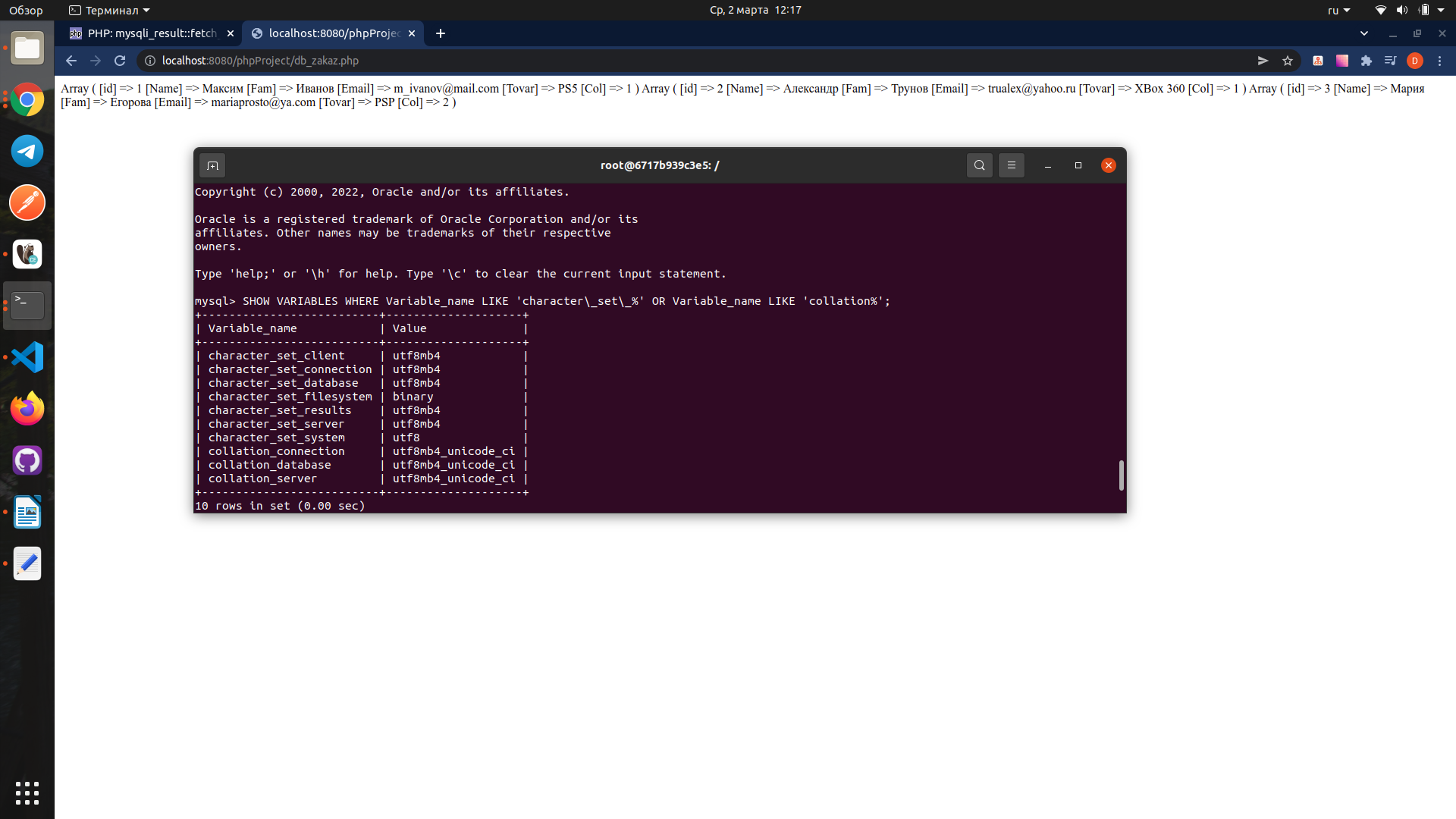
Выполнили PHP-скрипт (из вебинара) для вывода данных из таблицы Zakaz.



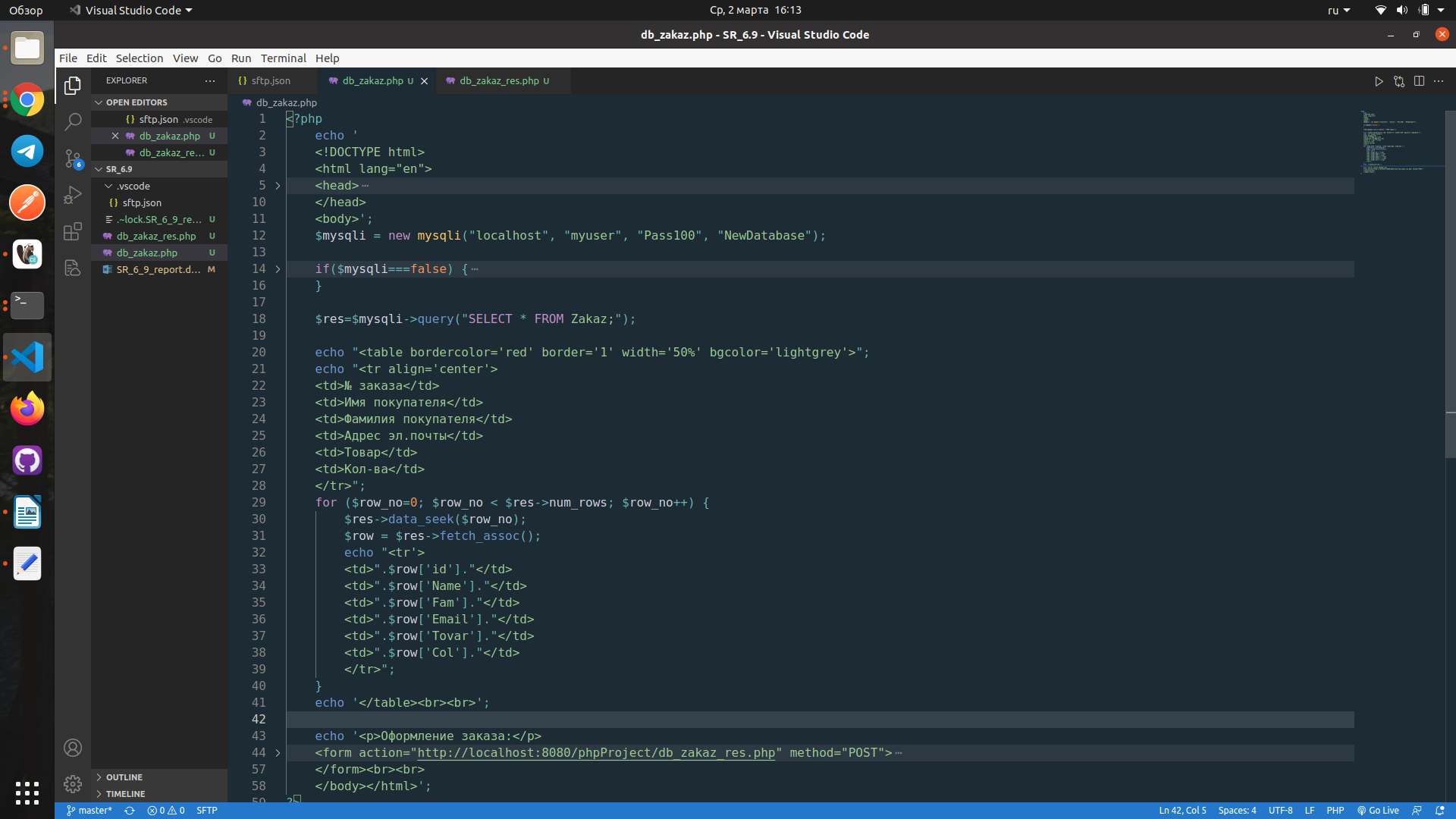
Обнаружили проблему — в конфигурации сервера БД не установлена нужная кодировка. По этой причине php-скрипт выводит кириллицу на веб-страницу некорректно. На снимке в терминале наша конфигурация. Ее необходимо привести к показанной на страницы вебсайта на заднем фоне.



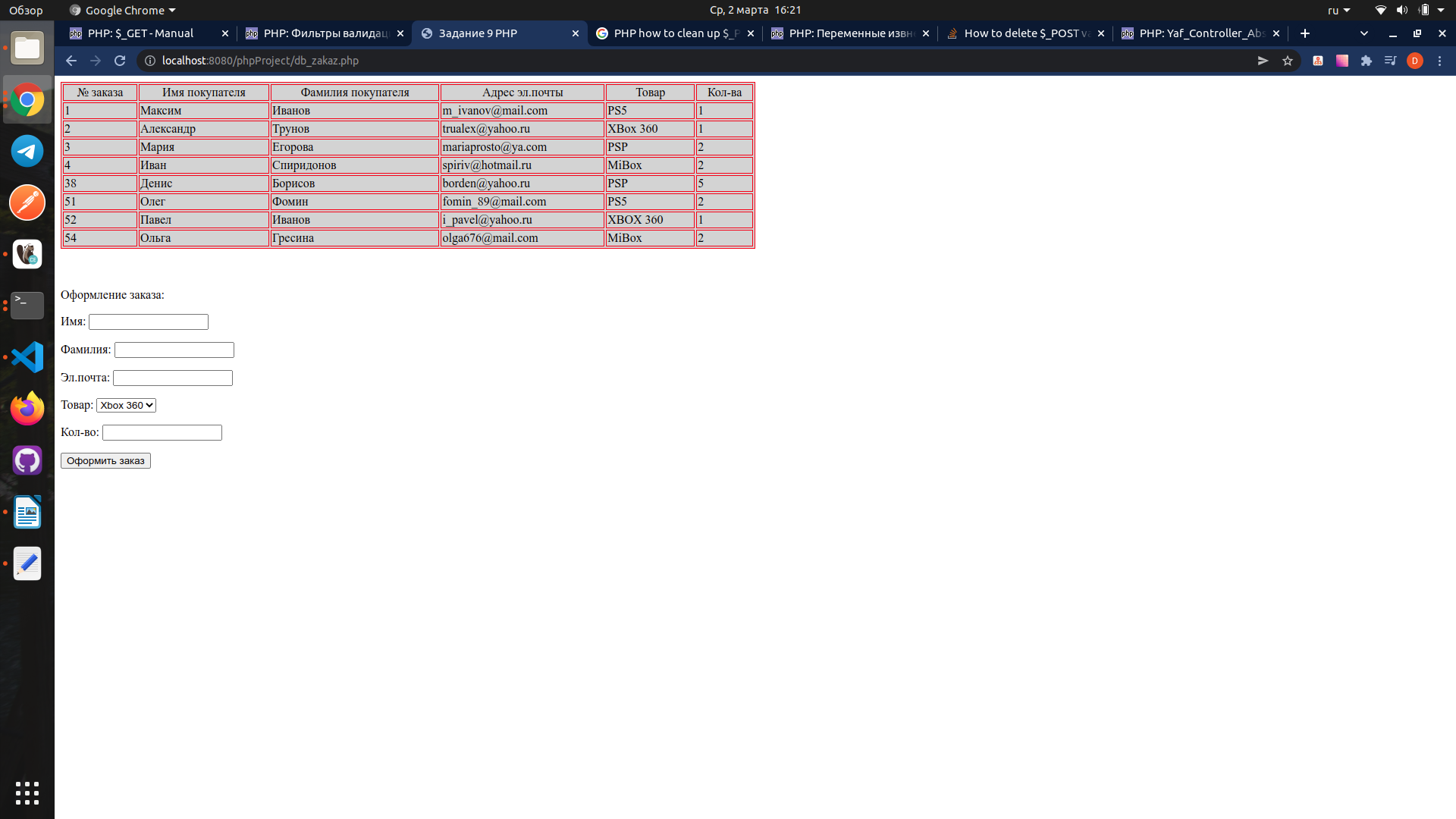
Поправили конфигурацию сервера БД (файл - /etc/mysql/conf.d/mysql.cnf). Перезапустили сервер (service mysql restart).



Конфигурация успешно изменена. Теперь кириллические символы отображаются верно при выполнении нашего php-скрипта.

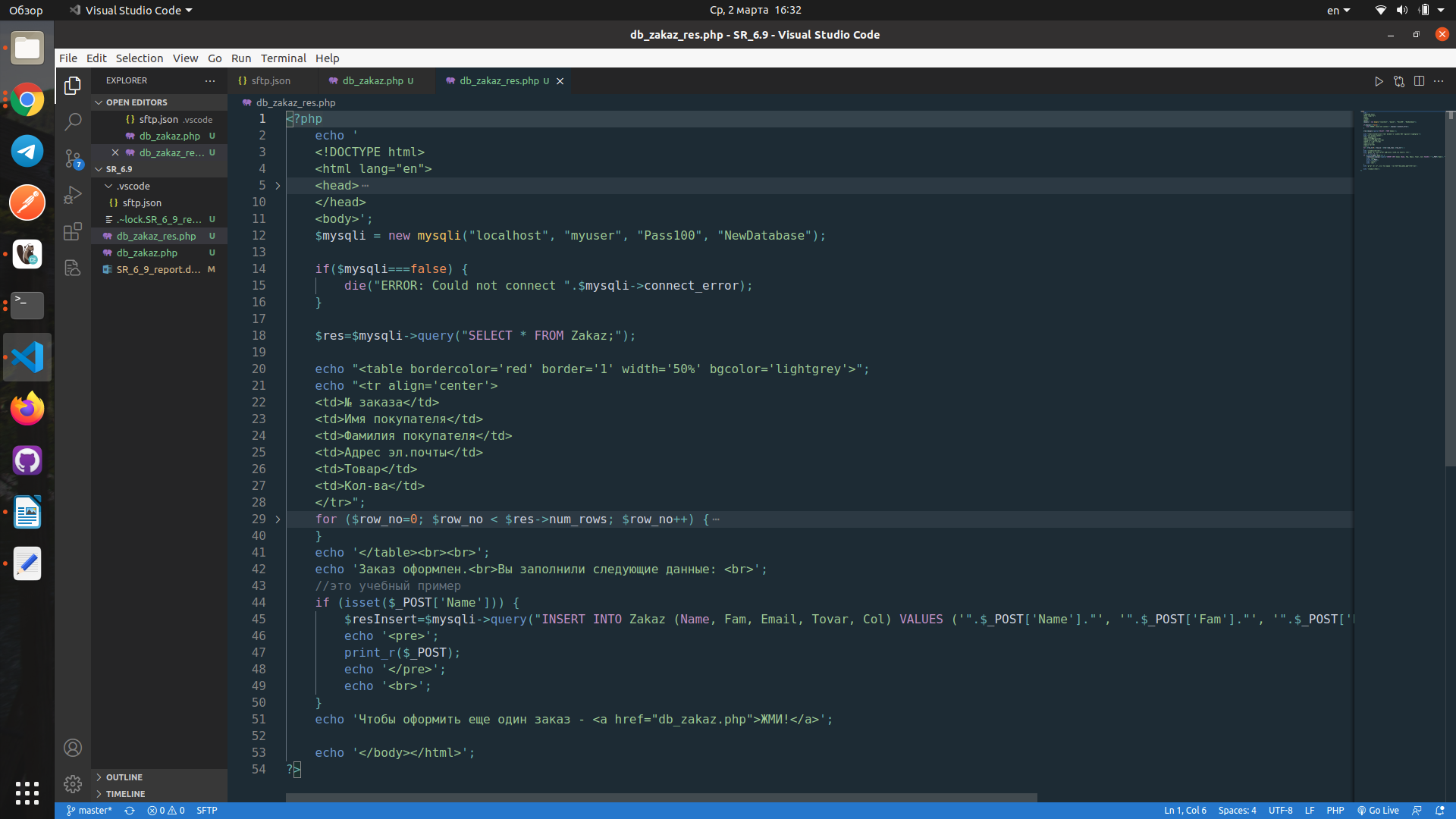


Дополняем наш PHP-скрипт (db\_zakaz.php) кодом для отрисовки нашей таблицы Zakaz в html.

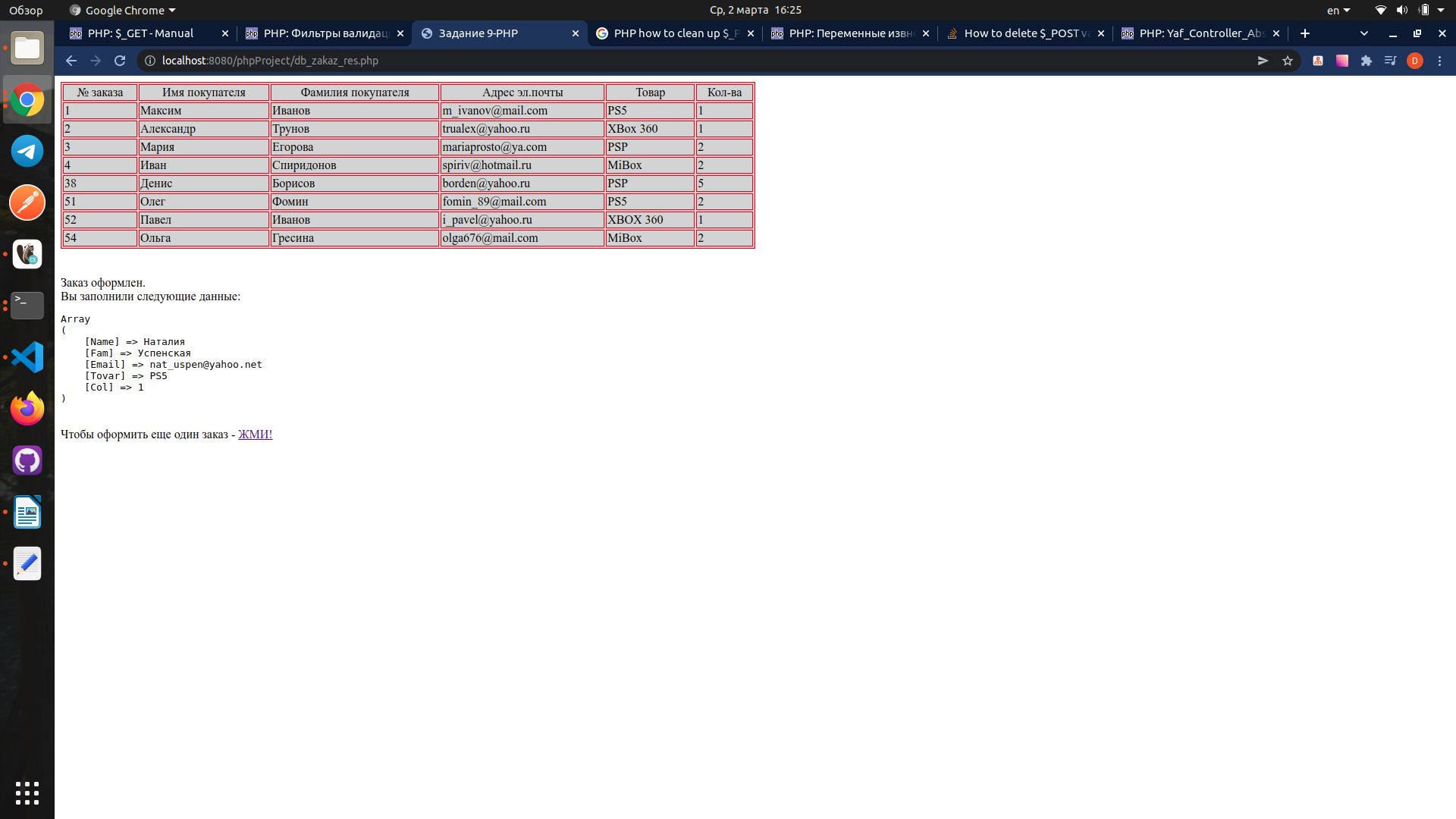


А также, следуя примеру Станислава из вебинара, создаем html-форму для добавления данных в нашу таблицу.

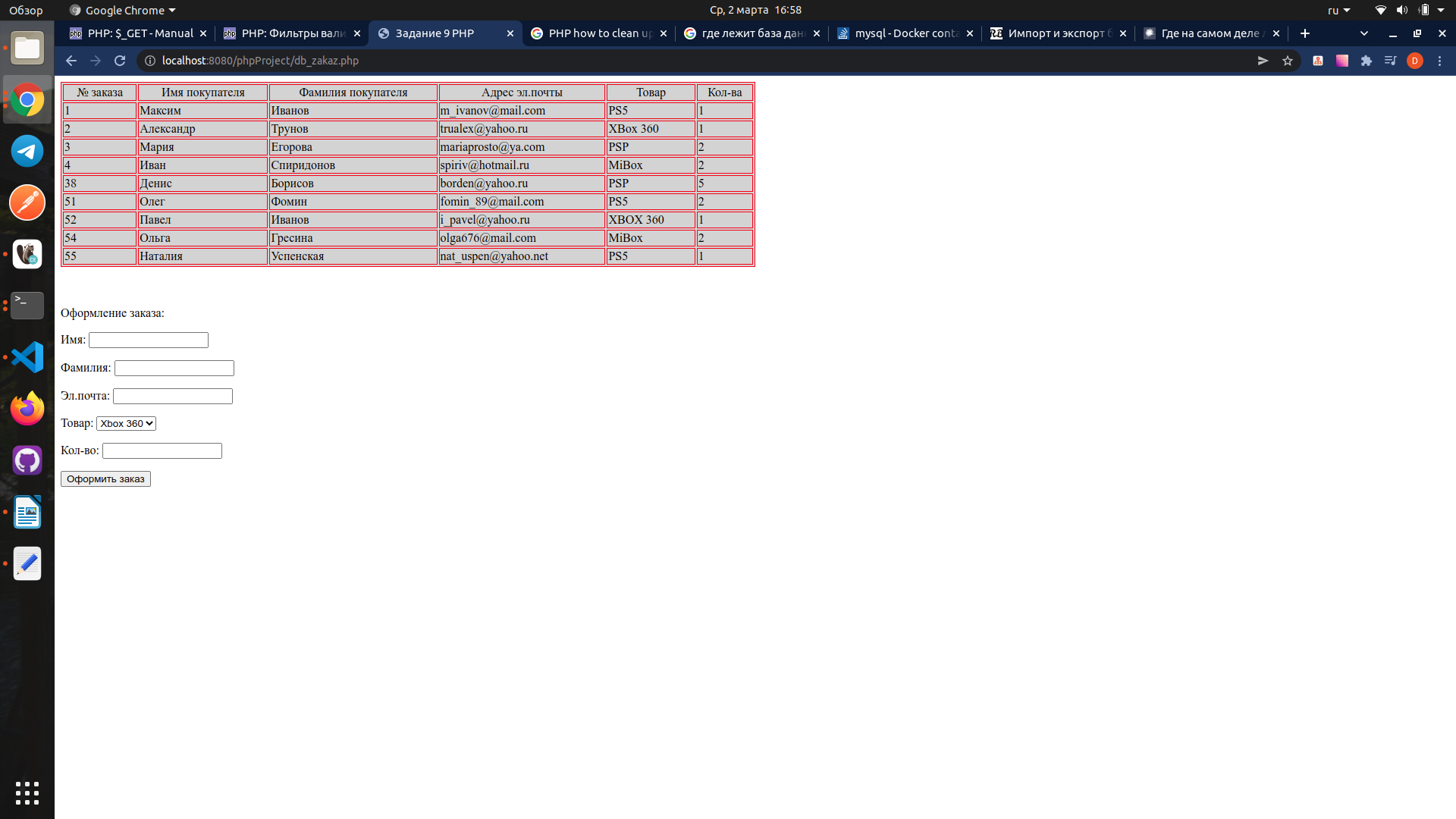
После отправки формы пользователь перенаправляется на другую html страницу (db\_zakaz\_res.php) — где ему показаны результаты ввода его данных и предлагается совершить переход по ссылке (ЖМИ!), чтобы оформить еще один заказ.



Таким образом мы возвращаем его на первую страницу (db\_zakaz.php)— для того чтобы затереть POST параметры во избежание повторной записи в БД от простого обновления html-cтраницы.



После перехода по ссылке пользователь возвращается на первую страницу и ему видны данные его заказа в последней строке таблицы:



Спасибо за внимание! :]