Московский государственный технический университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»

Кафедра «Системы обработки информации и управления»



**Рубежный контроль № 1**

**по дисциплине «Автоматизация разработки и эксплуатации ПО»**

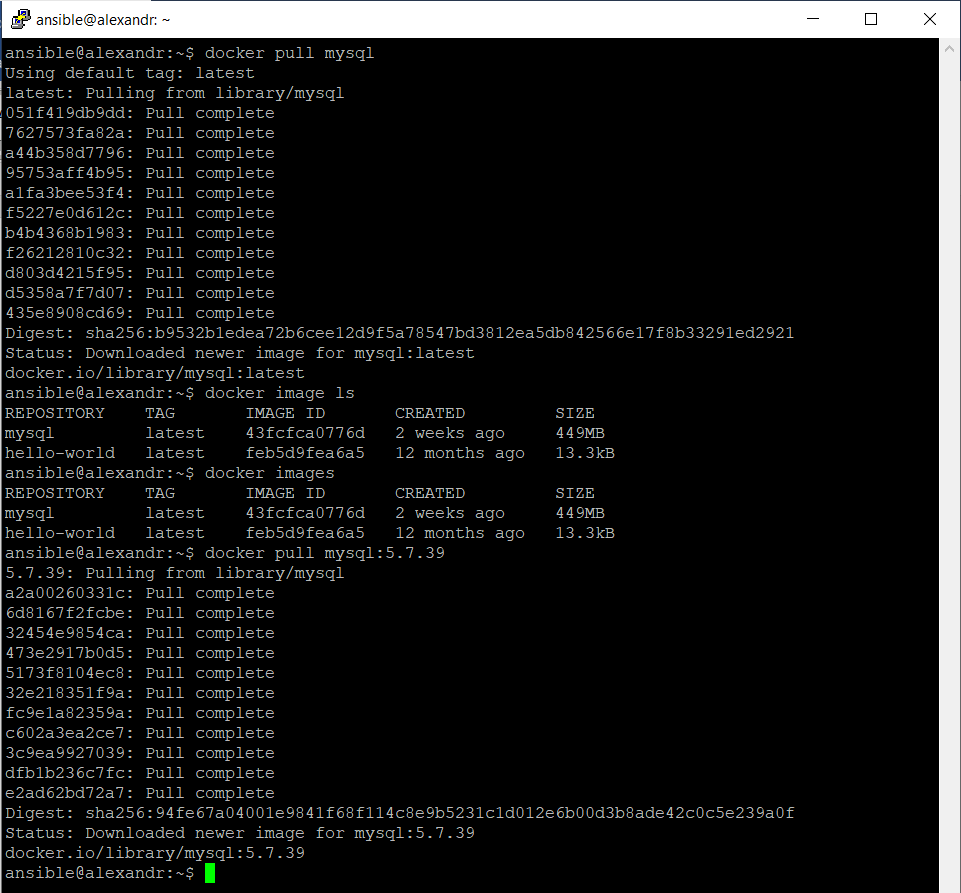
**«Контейнеризация docker»**

ИСПОЛНИТЕЛЬ:

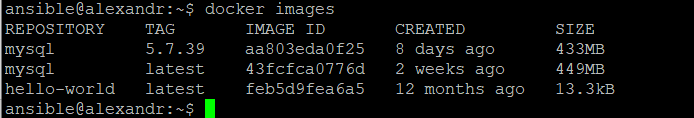
студент ИУ5-73Б

Алёшин А.Д.

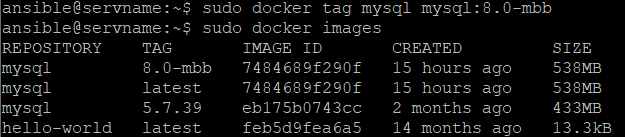
Загрузка образов:

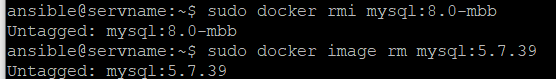


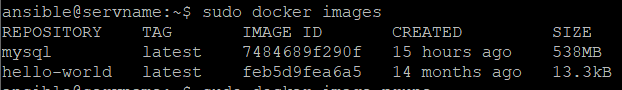
Убедимся, что теперь у нас есть 2 образа с разными метками:



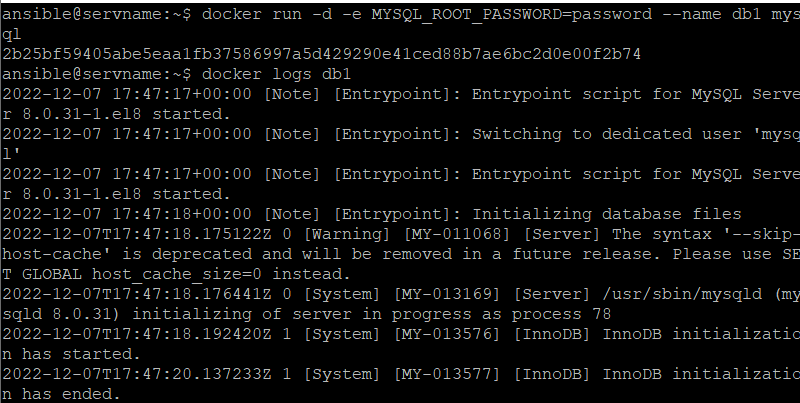
Метки и удаление образа







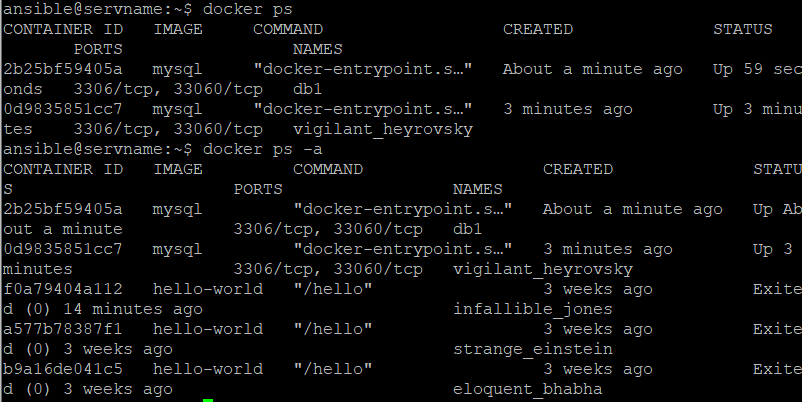
Запускаем контейнер - docker run , docker logs



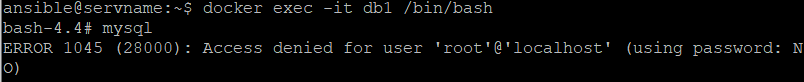
Списки контейнеров - docker ps

docker ps

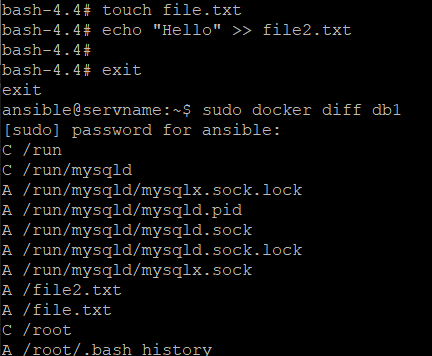
docker ps -a # включая завершенные



Подключаемся к работающему контейнеру - docker exec



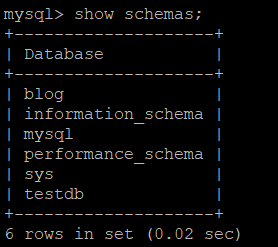
Список изменений - docker diff

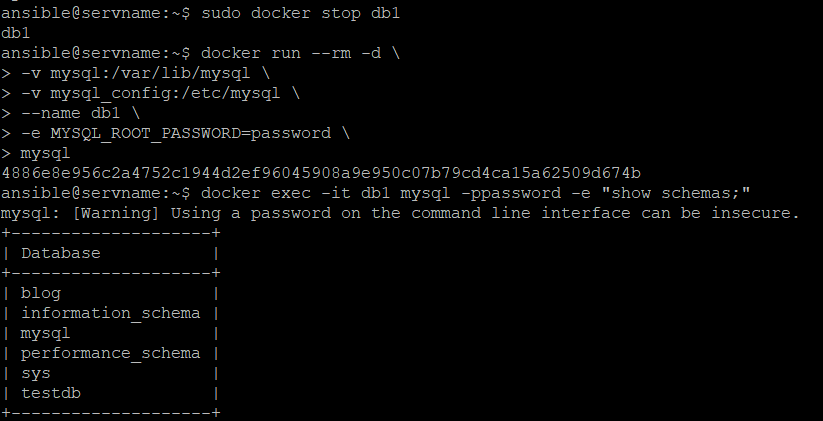


Завершаем контейнер - docker stop, docker kill , docker rm

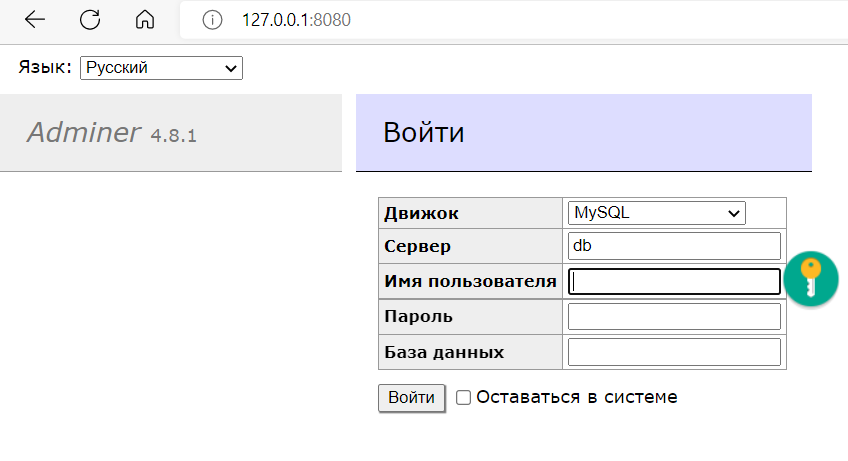


Не теряем данные - docker volume

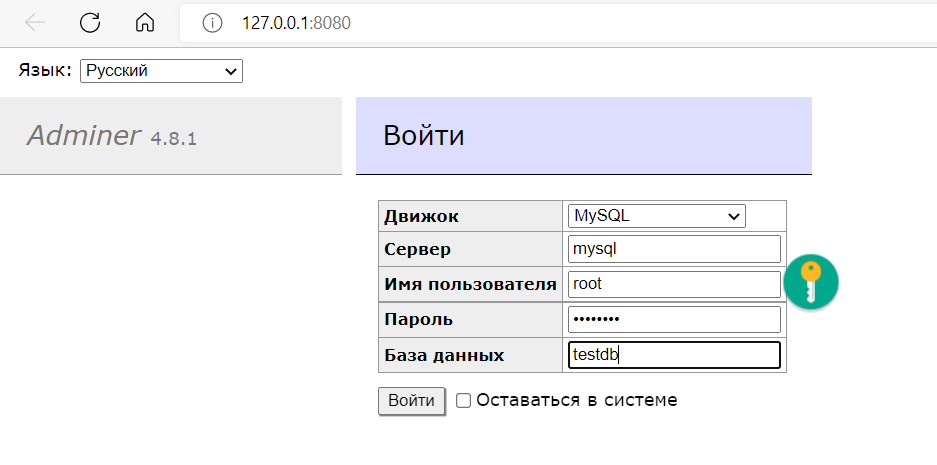


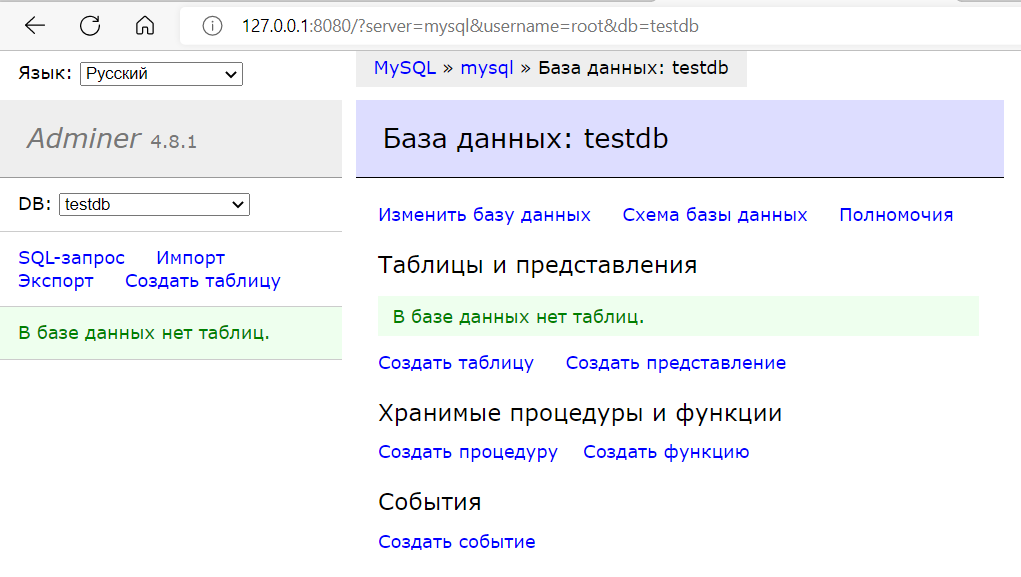


docker run -d -p 8080:8080 --name adminer adminer



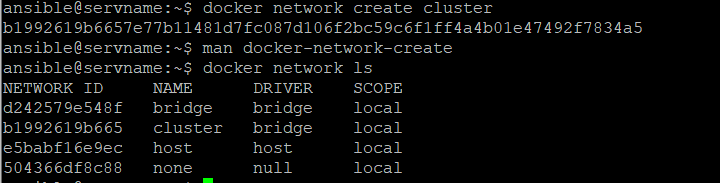
docker run -d -p 8080:8080 --link db1:mysql --name adminer adminer



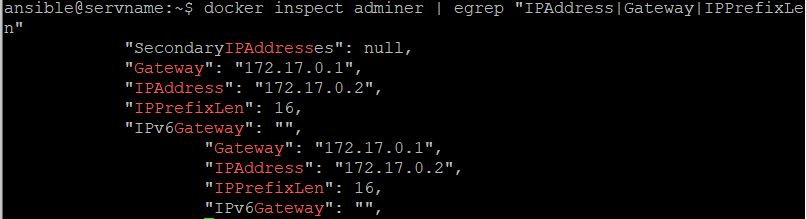


docker network create cluster

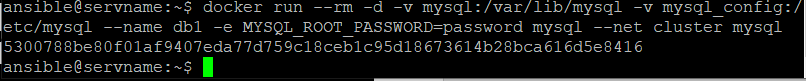
man docker-network-create



docker inspect adminer | egrep "IPAddress|Gateway|IPPrefixLen"



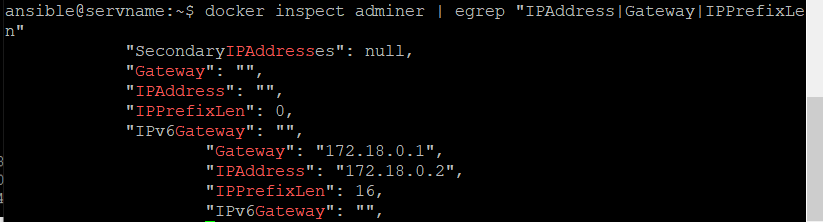
Перезапустим mysql и adminer, подключив к созданной нами сети

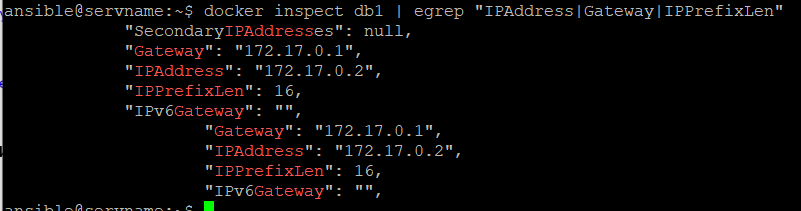


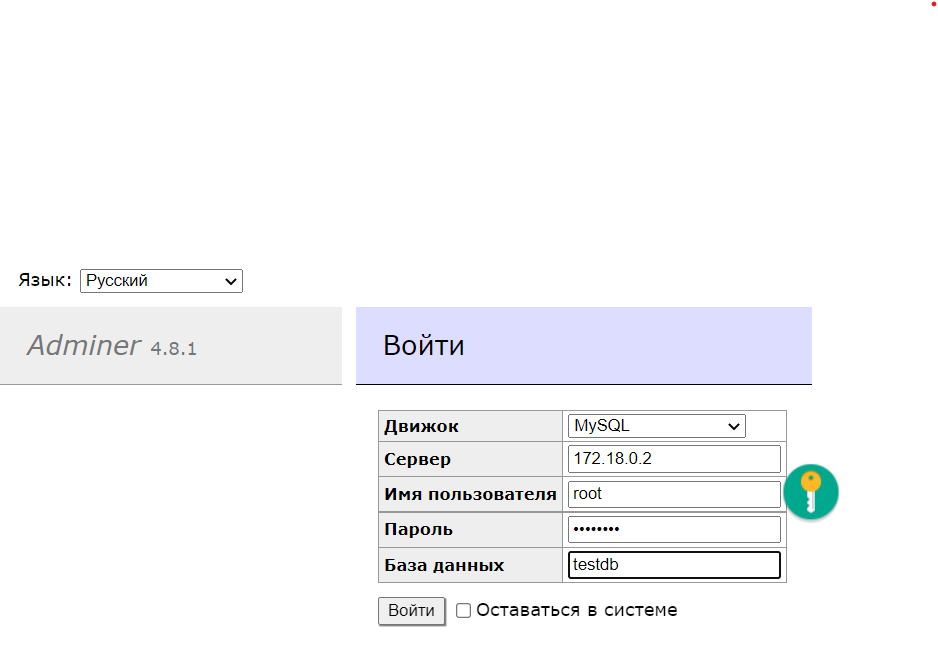


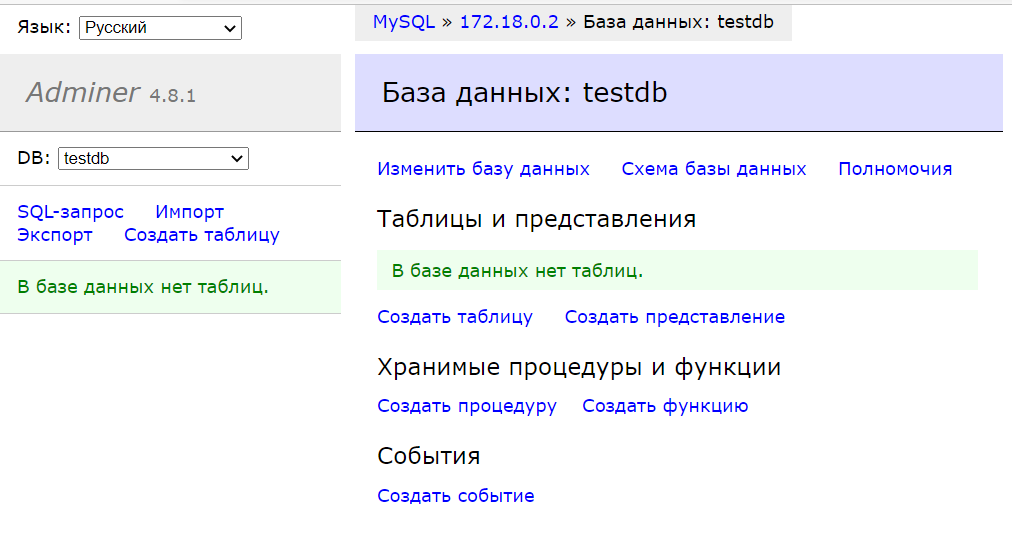
docker inspect adminer | egrep "IPAddress|Gateway|IPPrefixLen"

docker inspect db1 | egrep "IPAddress|Gateway|IPPrefixLen"

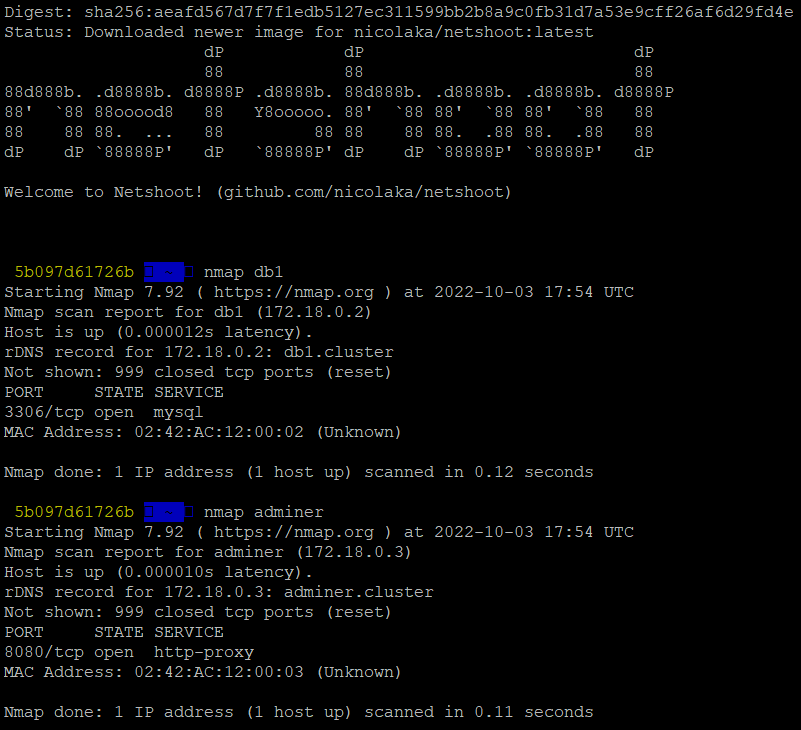






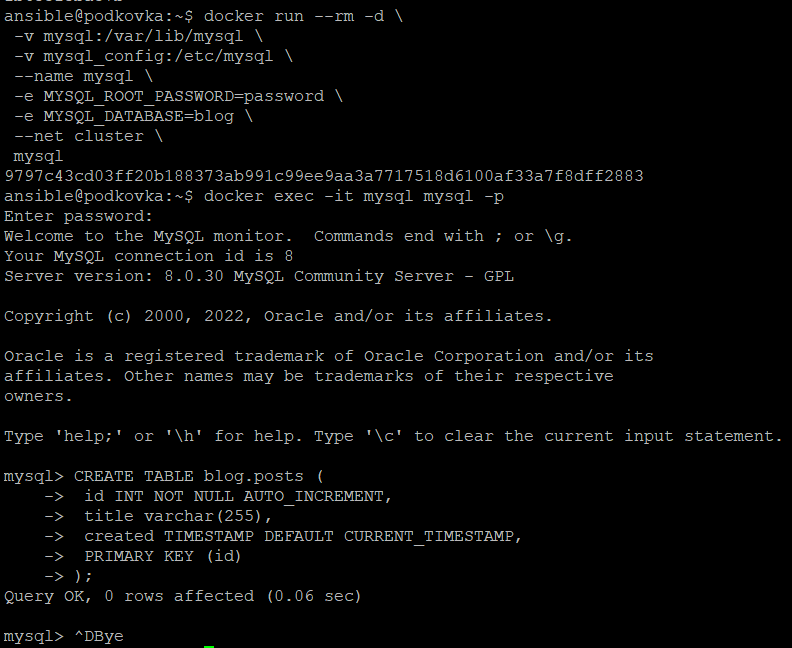


Запустим netshoot в интерактивном режиме и проверим сеть контейнеров с помощью nmap

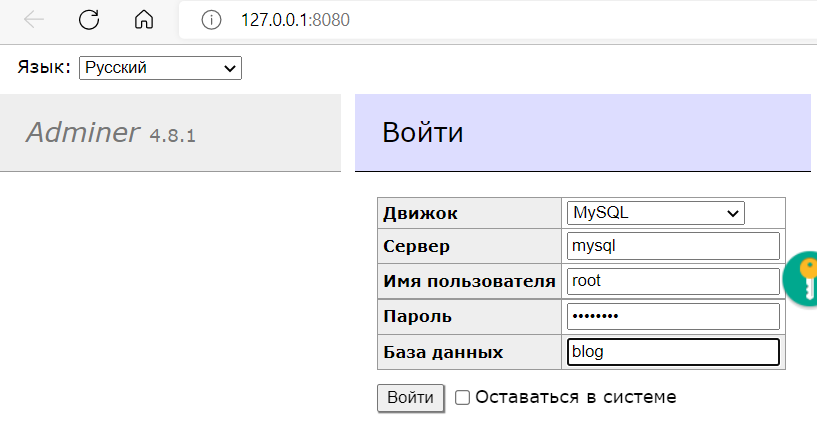


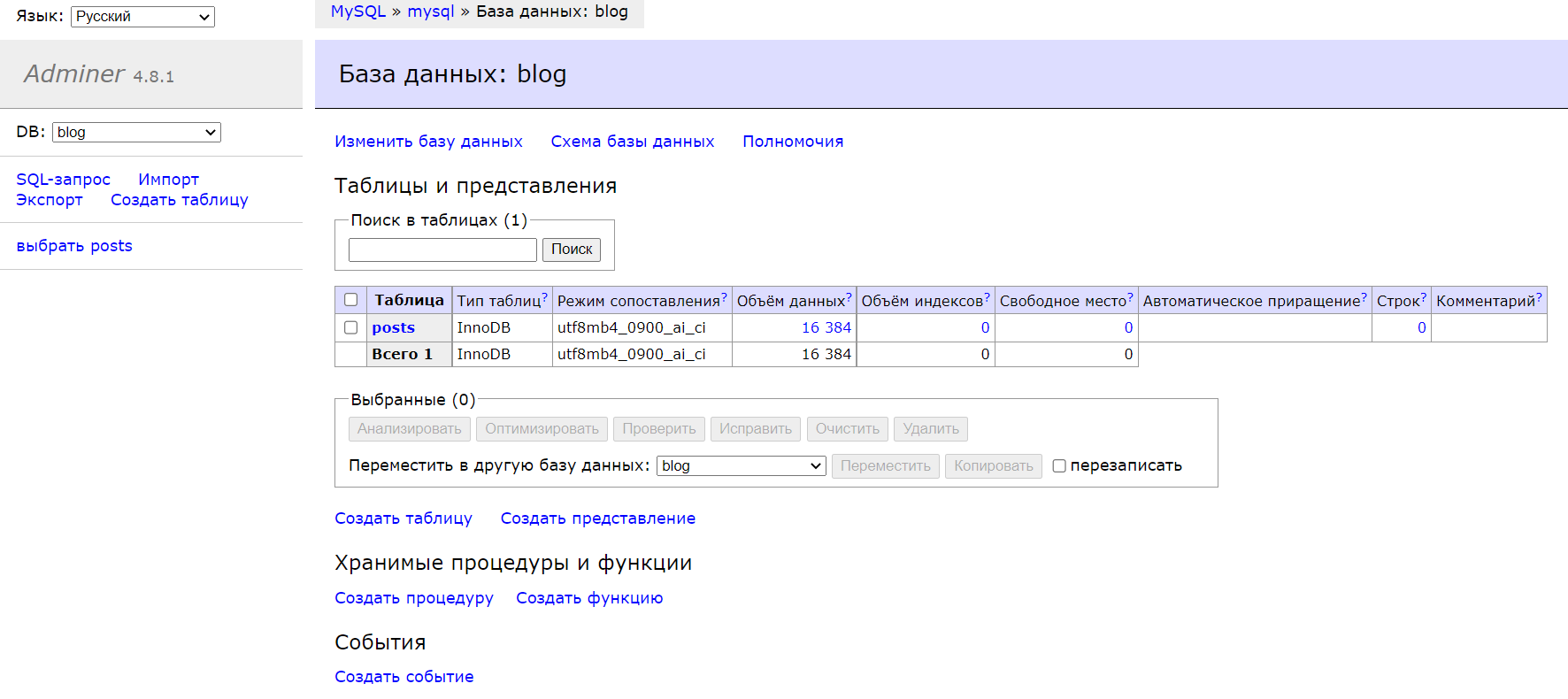
Часть 2

Создадим таблицу posts в базе blog

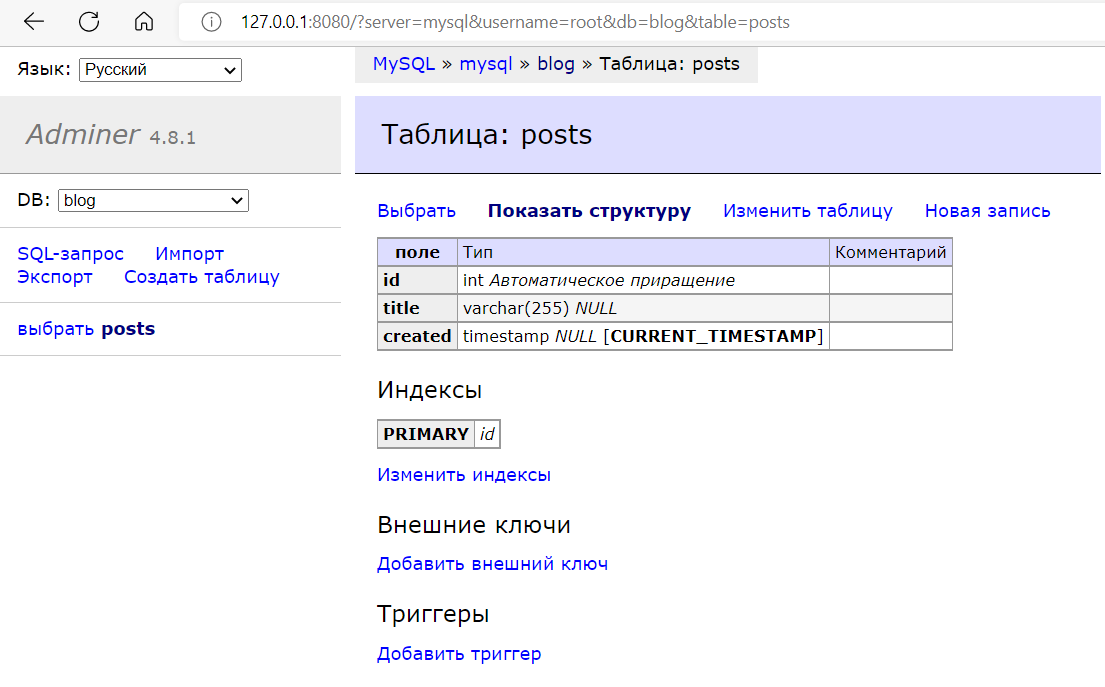


Запустим Adminer и подключимся к базе blog





Зайдём в таблицу posts



Добавим 3 записи

