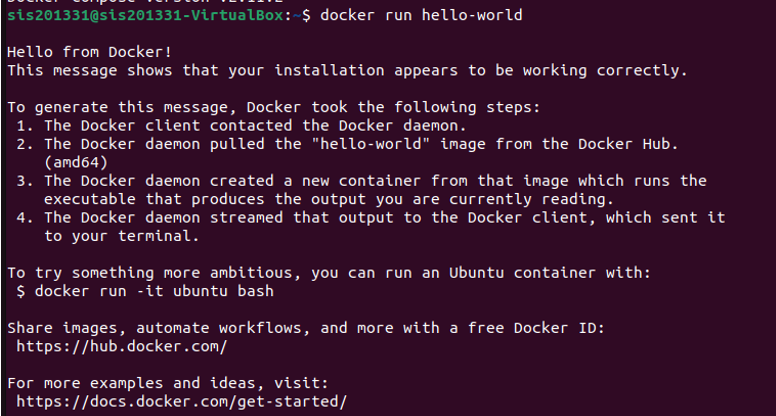
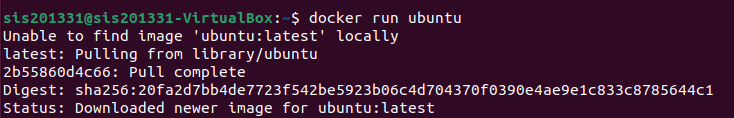
1. Установить Docker на всех развернутых в ходе ЛР2 виртуальных машинах



2. На усмотрение студента выбрать Docker-образ, который будет являться основой для разрабатываемого приложения



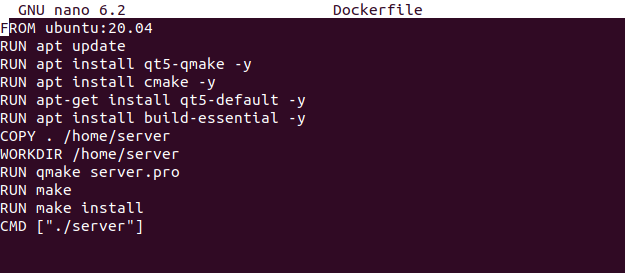
3. Получить образ на TEST-виртуальную машину



4. Добавить в корень проекта файл Dockerfile, внутри которого происходит сборка и запуск проекта из исходных файлов

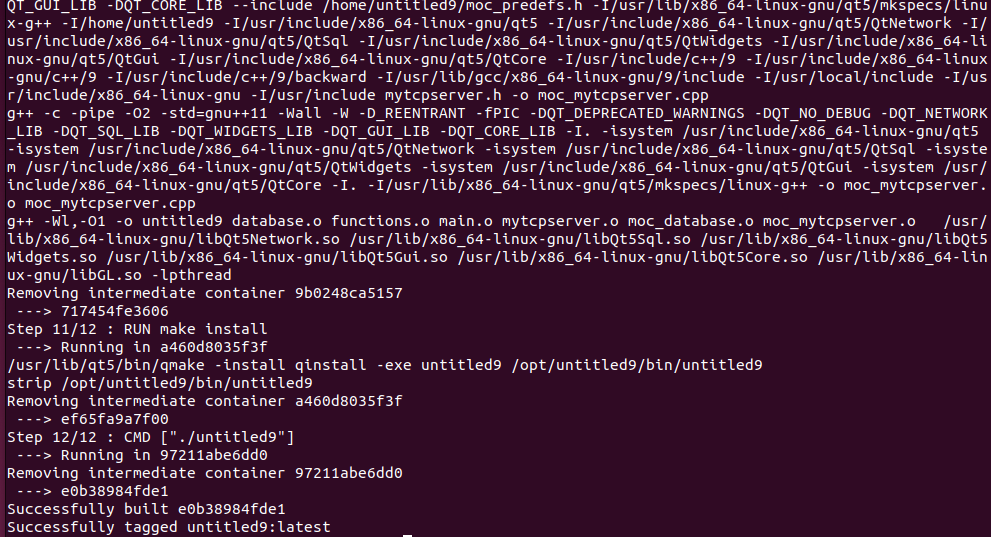




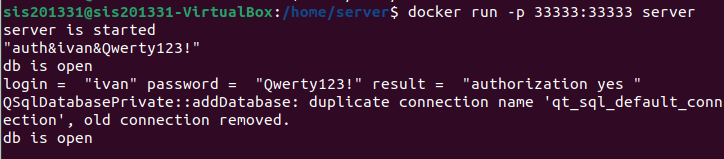


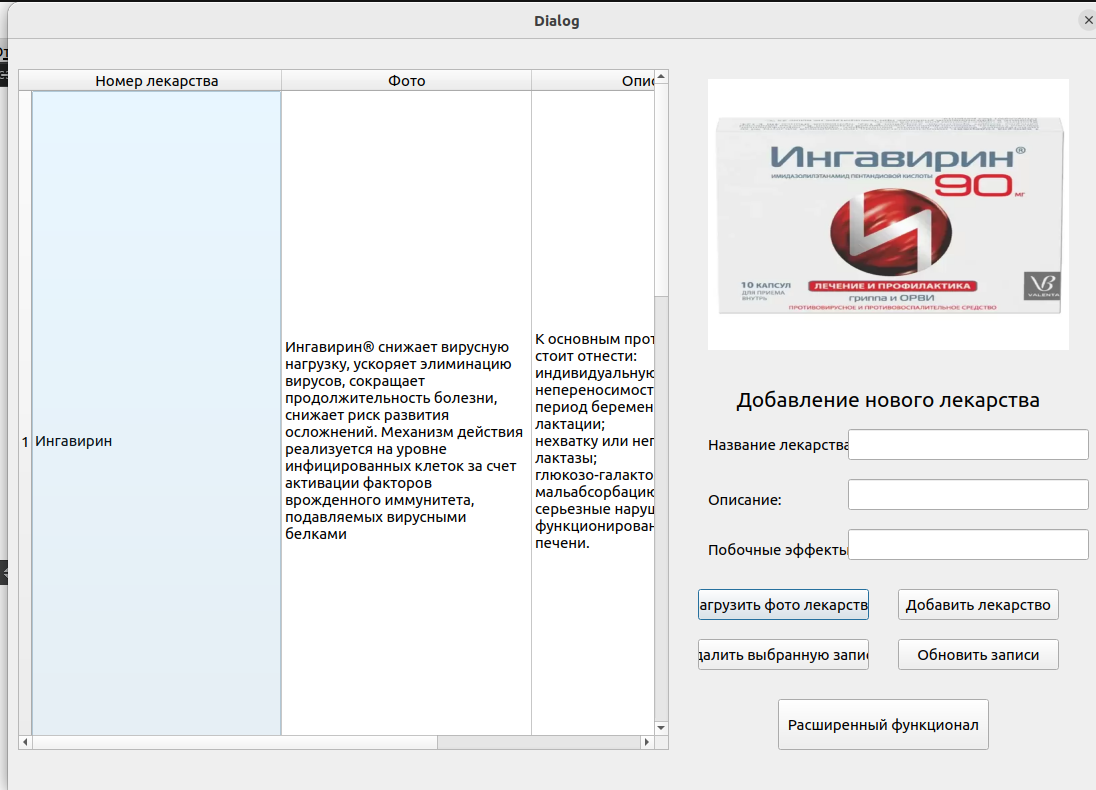
5. Собрать приложение внутри Docker, запустить контейнер из полученного образа







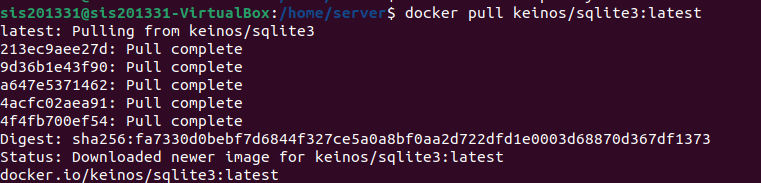




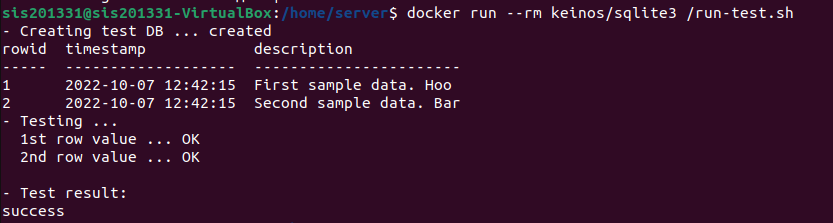
1. Установить на все ВМ утилиту docker-compose



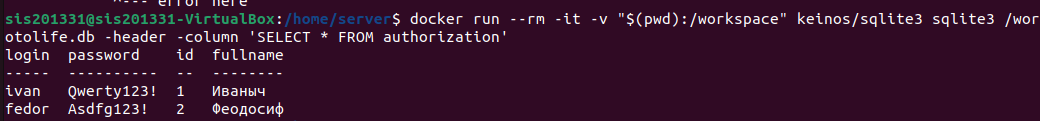
2. На выбор студента подобрать подходящую для приложения СУБД и получить с Dockerhub ее официальный (либо сторонний) образ. В случае, если приложение не использует БД, выбрать любой сторонний внешний сервис, используемый приложением (Redis, Minio и др.)



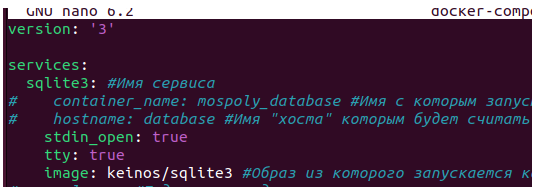


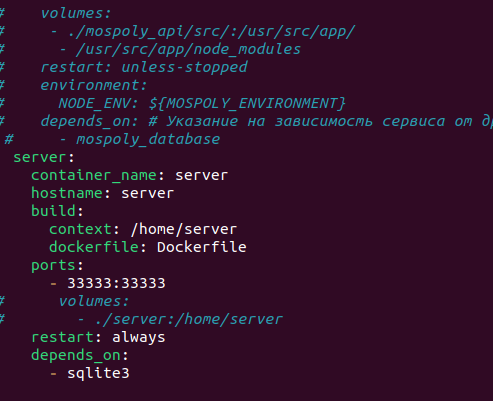




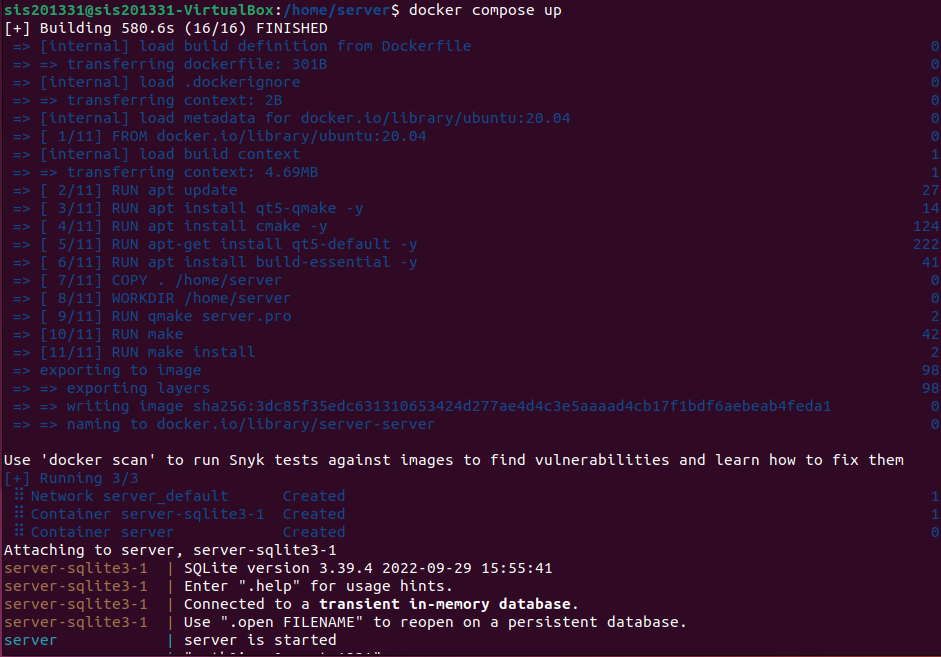


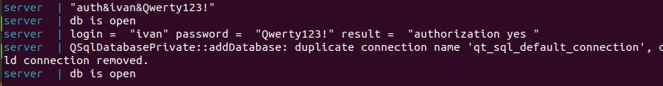
3. В корне проекта создать файл docker-compose.yaml, в котором описать параметры для сборки и запуска разрабатываемого приложения а также запуска стороннего сервиса, выбранного студентом. В конфигурации указать подключаемые директории (volume), в которых будут храниться файлы запущенных приложений (логи, файлы БД и другие).



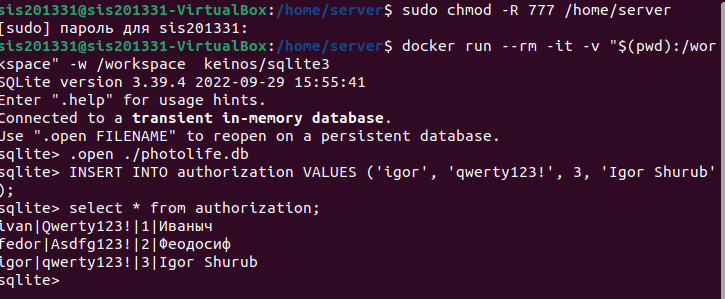


4. Запустить приложение и продемонстрировать его работу со сторонним сервисом.

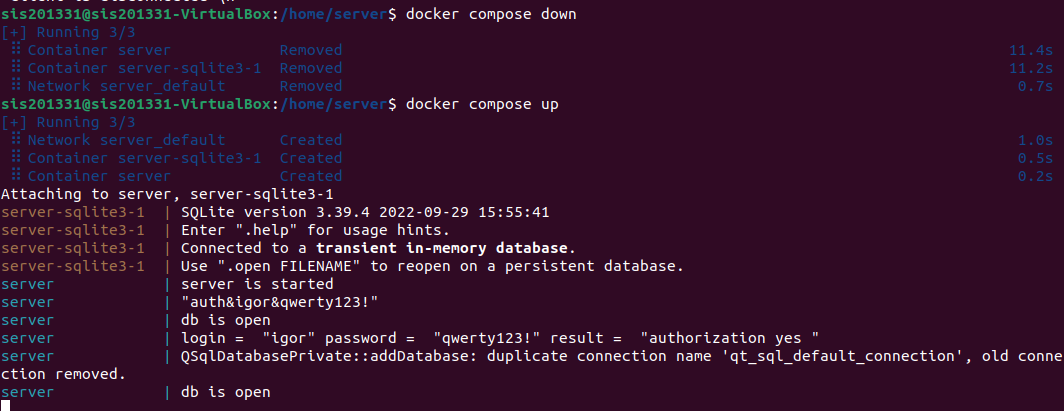


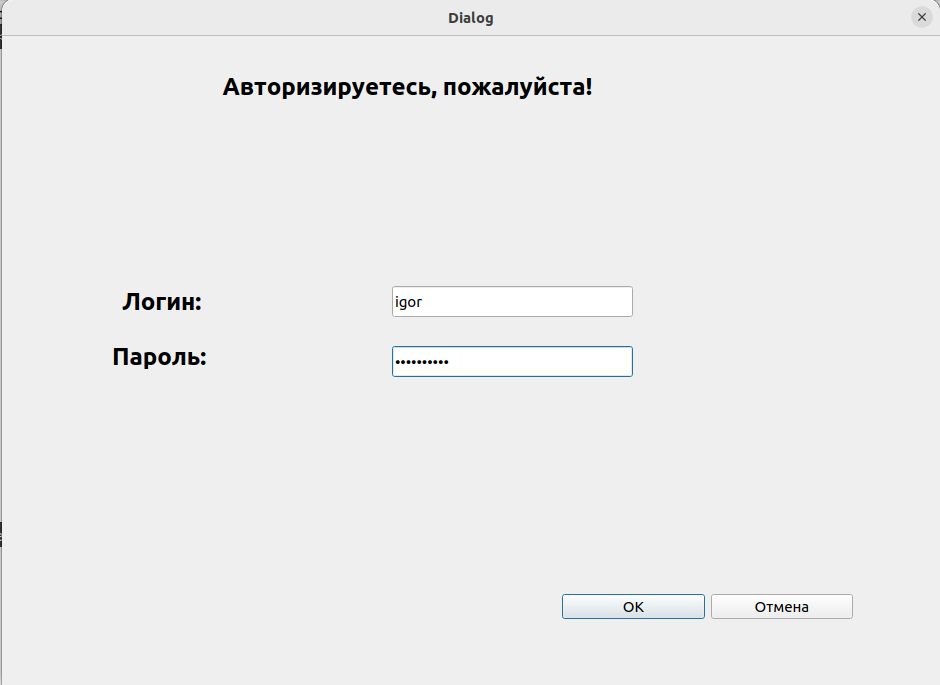


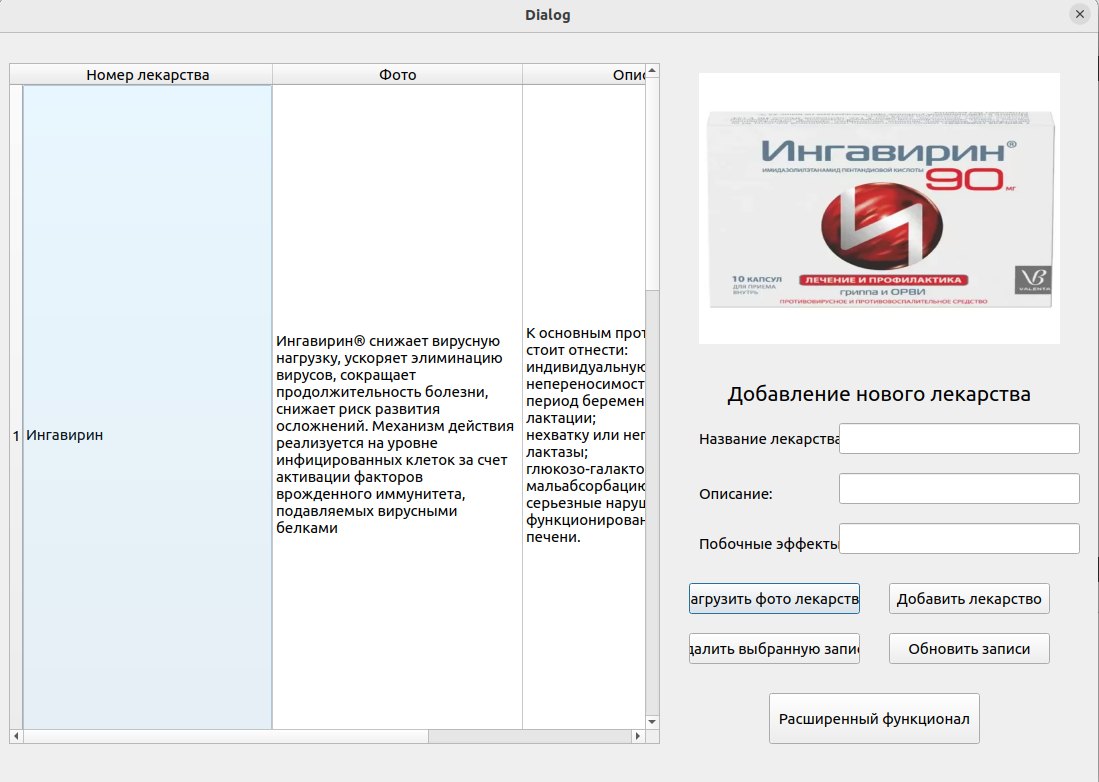
5. Внести изменения, требующие записи информации в файлы приложения (добавить записи в базу данных через приложение, создать файл и др.)



6. Остановить и запустить приложение еще раз, продемонстрировать, что изменения, внесенные в ходе работы, были сохранены.







1. Каким образом docker обеспечивает изоляцию контейнеров от хостовой ОС и друг от друга?

Приложение, запущенное в контейнере думает, что оно одно во всей ОС. Изоляция достигается за счет использования таких Linux-механизмов, как namespaces и control groups. Если говорить просто, то namespaces обеспечивают изоляцию в рамках ОС, а control groups устанавливают лимиты на потребление контейнером ресурсов хоста, чтобы сбалансировать распределение ресурсов между запущенными контейнерами.

2. Какие параметры требуется передать контейнеру docker при запуске, чтобы его сетевой адаптер был тождественен сетевому адаптеру хостовой ОС и почему?

docker run **-p 127.0.0.1**:80:80

Это эквивалентно режиму моста в VMware. Он находится в той же сети, что и хост-компьютер, но не имеет отдельного IP-адреса.Docker использует технологию пространства имен Linux для изоляции ресурсов, таких как процесс изоляции простраства имен PID, файловая система изоляции пространства имен монтирования, сеть изоляции пространства имен сети и т.д. Пространство сетевых имен обеспечивает независимую сетевую среду, включая сетевые карты, маршрутизацию, правила iptable и т.д. Изолированные от других пространств сетевых имен. Контейнеру Docker обычно назначается независимое сетевое пространство имен. Но если при запуске контейнера используется режим хоста, контейнер не получит независимого сетевого пространства имен, но будет совместно использовать сетевое пространство имен с хостам. Контейнер не будет виртуализировать свою сетевую карту, настроить собвственный айпи адрес и т.д. Но будет использовать айпи адрес и порт хоста.

3. Какую последовательность команд требуется ввести, чтобы при помощи docker-compose сначала собрать приложение, затем запустить его в фоновом режиме и затем подключиться к его выводу stdout и stderr?

