МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОРОНЕЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «ВГТУ», ВГТУ)

Факультет информационных технологий и компьютерной безопасности

Кафедра компьютерных интеллектуальных технологий проектирования

ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

По дисциплине: «Системы хранения и обработки данных»

Тема: «Развёртывание СУБД Postgres с использованием средств автоматизации развёртывания и управления приложениями»

Выполнил работу студент группы мИИВТ-241: Ульянов А.Г.

подпись, дата

Принял: Короленко В.В.

подпись, дата

Воронеж 2024

Задачи лабораторной работы:

• установка приложения для работы с Docker-контейнерами;

• установка программного средства для работы с СУБД DBeaver;

• скачивание docker-образа с СУБД postgres;

• создание Dockerfile;

• создание образа на основе Dockerfile;

• запуск контейнера для развёртывания СУБД Postgres;

• подключение к работающему контейнеру и запуск интерфейса psql;

• создание контейнера с томом (volume);

• создание контейнера с использованием файла docker-compose.yml

1. Установка и настройка WSL

Переходим на страницу установки WSL. В разделе «Step 4 - Download the Linux kernel update package» кликните на ссылку «WSL2 Linux kernel update package for x64 machines», чтобы загрузить файл «wsl\_update\_x64.msi».

После этого установите загруженное обновление. Затем в Windows PowerShell введите команду:

powershell

wsl --set-default-version 2

для установки WSL по умолчанию на версию 2.

2. Создание Dockerfile

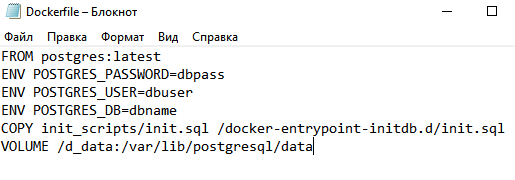


Рисунок 1 – Docker file

Содержимое Dockerfile:

dockerfile

FROM postgres:latest

ENV POSTGRES\_PASSWORD=dbpass

ENV POSTGRES\_USER=dbuser

ENV POSTGRES\_DB=dbname

COPY init\_scripts/init.sql /docker-entrypoint-initdb.d/init.sql

1. FROM postgres:latest — задаёт базовый образ для нового Docker-образа, последняя версия PostgreSQL.

2. ENV POSTGRES\_PASSWORD=dbpass — устанавливает пароль для пользователя PostgreSQL.

3.ENV POSTGRES\_USER=dbuser — задаёт имя пользователя PostgreSQL.

4. ENV POSTGRES\_DB=dbname — задаёт имя создаваемой базы данных.

5. COPY init\_scripts/init.sql /docker-entrypoint-initdb.d/init.sql — копирует файл init.sql в образ, который выполняется при старте контейнера.

После этого перезагружаем компьютер и вновь запускаем Docker Desktop. Создайте образ с помощью команды:

bash

docker build -t d\_img:latest .

Эту команду нужно выполнять из каталога, где находится Dockerfile. В результате будет создан Docker-образ «d\_img».

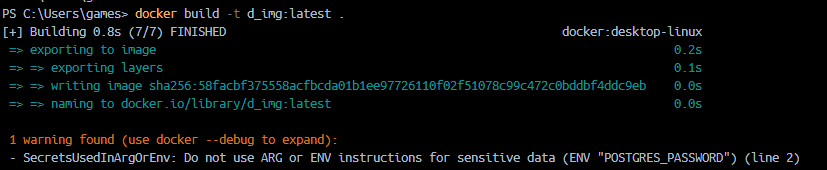


Рисунок 2 – Образ docker

3. Создание Docker-контейнера

Для создания контейнера вводим в консоли:

bash

docker run -d -p 5432:5432 --name d\_cont d\_img:latest

- docker run — запускает контейнер на основе указанного образа.

- -d — запускает контейнер в фоновом режиме.

- -p 5432:5432 — перенаправляет порт 5432 локальной машины на порт 5432 контейнера.

- --name d\_cont — задаёт имя контейнера.

- d\_img:latest — имя и тег образа, на основе которого создаётся контейнер.

После создания контейнера в терминале будет указан его номер.

4. Проверка базы данных через DBeaver

Подключение:

- Имя пользователя: dbuser

- Пароль: dbpass

- База данных: dbname

- Порт: 5432

5. Подключение к контейнеру

Для подключения к контейнеру выполним:

bash

docker exec -it d\_cont psql -d dbname -U dbuser

- docker exec — выполняет команду в работающем контейнере.

- -it — включает интерактивный режим.

- d\_cont — имя контейнера.

- psql — запускает клиент PostgreSQL.

- -d dbname — подключается к базе данных `dbname`.

- -U dbuser — указывает пользователя PostgreSQL.

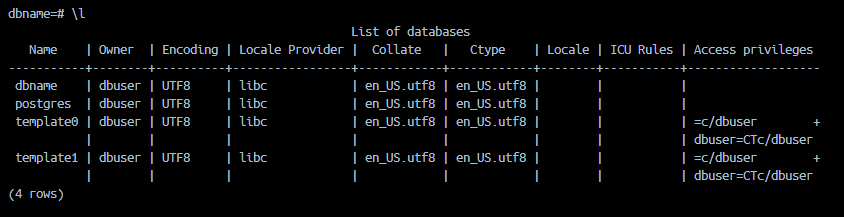


Рисунок 3 – Проверка Docker

Команда для внесения изменений

Для подключения к работающему контейнеру и запуска интерфейса psql используем:

bash

docker exec -it d\_cont psql -d dbname -U dbuser

6. Удаление и создание нового контейнера

Для удаления и создания нового контейнера необходимо использовать:

bash

docker run -d -p 5432:5432 --name d\_cont d\_img:latest -v /d\_data:/var/lib/postgresql/data

7. Создание файла с командами

Создадим текстовый файл «cmd.txt», в котором запишем все команды, необходимые для развертывания базы данных PostgreSQL с помощью Dockerfile.

docker-compose.yml:

yaml

version: "3.9"

services:

dbpost:

image: postgres:latest

environment:

POSTGRES\_DB: dbcompose

POSTGRES\_USER: usrcompose

POSTGRES\_PASSWORD: passcompose

volumes:

- ./init\_scripts/init.sql:/docker-entrypoint-initdb.d/init.sql

ports:

- 5432:5432

1. version: "3.9" — указывает версию синтаксиса Docker Compose.

2. services: — раздел описания сервисов (контейнеров).

3. dbpost: — имя сервиса.

4. image: postgres:latest — используется официальный образ Postgres.

5. environment: — определяет переменные среды для настройки контейнера.

6. volumes: — указывает монтирование томов.

7. ports: — это процесс, который позволяет направлять сетевой трафик с одного порта на другой. В контексте Docker это означает, что вы можете перенаправить порты вашего локального ПК на порты, используемые внутри контейнера.