**Задание 1. Dockerfile для Jupyter Notebook**

Dockerfile:

# Базовый образ

FROM python:3.12.3

# Установим системные зависимости

RUN apt-get update && apt-get install -y build-essential libpq-dev

# Рабочая директория

WORKDIR /app

# Копируем файлы в контейнер

COPY ./requirements.txt ./

# Установка необходимых библиотек

RUN pip install --no-cache-dir -r requirements.txt

# Устанавливаем Jupyter Notebook

RUN pip install jupyterlab

# Открываем порт 8888 для доступа к Jupyter

EXPOSE 8888

# Команда запуска Jupyter Notebook

CMD ["jupyter", "lab", "--ip=0.0.0.0", "--allow-root", "--NotebookApp.token=''"]

Требования (`requirements.txt`):

numpy

pandas

scikit-learn

matplotlib

seaborn

Сборка образа:

docker build -t my-ml-notebook .

Запуск контейнера:

docker run -d -p 8888:8888 --name notebook\_container my-ml-notebook

После выполнения этих шагов Jupyter Notebook будет доступен по адресу: http://localhost:8888.

---------------------------------------------------------------------------------------------------------------------------------

**Задание 2. Использование Docker Compose**

Файл `docker-compose.yml`:

version: '3'

services:

notebook:

image: my-ml-notebook

ports:

- "8888:8888"

db:

image: postgres:13

environment:

POSTGRES\_USER: postgres

POSTGRES\_PASSWORD: postgres

POSTGRES\_DB: postgres

volumes:

- pgdata:/var/lib/postgresql/data

volumes:  
 pgdata:

Запуск сервисов:

docker-compose up -d

Проверяем работу сервисов командой:

docker-compose ps

или

docker ps

Оба контейнера будут запущены и доступны для взаимодействия.

-----------------------------------------------------------------------------------------------------

**Задание 3. Сохранение и передача Docker-образа/контейнера**

Команды bash:

Сохранение образа:

docker save -o my\_ml\_notebook.tar my-ml-notebook:latest

Опциональный компрессия:

docker save my-ml-notebook:latest | gzip > my\_ml\_notebook.tar.gz

Сохранение контейнера:

docker export -o notebook\_container.tar notebook\_container

Передача файлов:

- Скопировать на флешку или отправить по сети.

Импорт образа на другой машине:

docker load -i my\_ml\_notebook.tar

Импорт контейнера:

docker import notebook\_container.tar imported\_notebook:latest