**Лабораторная работа №2**

**Контейнеризация приложения**

### Шаг 0: Создание ветки

Перед внесением изменений в проект создайте отдельную ветку в вашем репозитории с названием feature/lab2\_docker:

git checkout -b feature/lab2\_docker

### Шаг 1: Установка Docker

Если у вас еще не установлен Docker, скачайте и установите Docker Desktop с официального сайта [Docker](https://www.docker.com/).

### Шаг 2: Использование вашего Node.js проекта

Вы должны использовать Node.js проект, который вы разработали на предыдущей лабораторной работе.

### Шаг 3: Создание Dockerfile

В корневой директории вашего проекта создайте файл с именем Dockerfile (без расширения) со следующим содержанием:

*# Используем официальный образ Node.js как базовый*

FROM node:14

*# Устанавливаем рабочую директорию*

WORKDIR /usr/src/app

*# Копируем package.json и package-lock.json*

COPY package\*.json ./

*# Устанавливаем зависимости*

RUN npm install

*# Копируем весь код проекта в контейнер*

COPY . .

*# Открываем порт, на котором будет работать приложение*

EXPOSE 3000

*# Команда для запуска приложения*

CMD ["node", "app.js"]

### Шаг 4: Создание .dockerignore

Чтобы исключить ненужные файлы из контейнера, создайте файл .dockerignore в корневой директории:

node\_modules

npm-debug.log

### Шаг 5: Сборка Docker-образа

В терминале (в директории вашего проекта) выполните следующую команду, чтобы построить образ:

docker build -t my-node-app .

### Шаг 6: Запуск контейнера

После успешной сборки образа, запустите контейнер с помощью следующей команды:

docker run -p 3000:3000 my-node-app

**Шаг 7: Проверка работы приложения с помощью Postman**

Откройте приложение Postman и создайте новый запрос.

**Контрольные вопросы:**

1. Что такое Docker и для чего он используется?
2. Объясните разницу между контейнерами и виртуальными машинами.
3. Что такое Dockerfile? Какие основные инструкции можно использовать в Dockerfile?
4. Каковы шаги для создания Docker-образа из Dockerfile?
5. Как можно передать переменные окружения в контейнер при его запуске?
6. Что такое Docker Compose и для чего он используется?
7. Как можно просмотреть список всех запущенных контейнеров?
8. Что такое .dockerignore и как он влияет на процесс сборки образа?
9. Как запустить контейнер в фоновом режиме и как его остановить?
10. Чем отличается команда docker run от docker build?
11. Как можно получить доступ к терминалу работающего контейнера?
12. Как можно удалить неиспользуемые образы и контейнеры, чтобы освободить место на диске?