**МИНИСТЕРСТВО ТРАНСПОРТА И КОММУНИКАЦИЙ РЕСПУБЛИКИ БЕЛАРУСЬ**

**Учреждение образования**

**«Белорусский государственный университет транспорта»**

**Кафедра «Управление эксплуатационной работой и охрана труда**

**Отчёт  
по лабораторным работам**

по дисциплине «Средства и технологии анализа и разработки информационных систем»

Выполнил Проверил

Студент группы ГИ-31 м.т.н., ст.пр.

Долинова В. А. Козлов В. Г.

Гомель 2024

# Лабораторная работа №2 ТЕМА: Установка среды разработки. Внедрение зависимостей

**Цель работы:** познакомиться с корпоративными средствами создания и сборки приложений. Создать веб-приложение посредством Maven и импортировать его под IntelliJ IDEA.

I. Настройка инструментариев и среды для разработки приложений на JAVA.

Для того чтобы начать работу со сборщиком приложения Maven проведена настройка среды. Для этого:

1. Установлена переменная окружения M2\_HOME.

2. Установлена переменная окружения PATH.

3. Установлена переменная окружения JAVA\_HOME, которая указывает путь к установленному JDK.

Результат представлен на рисунке 1.

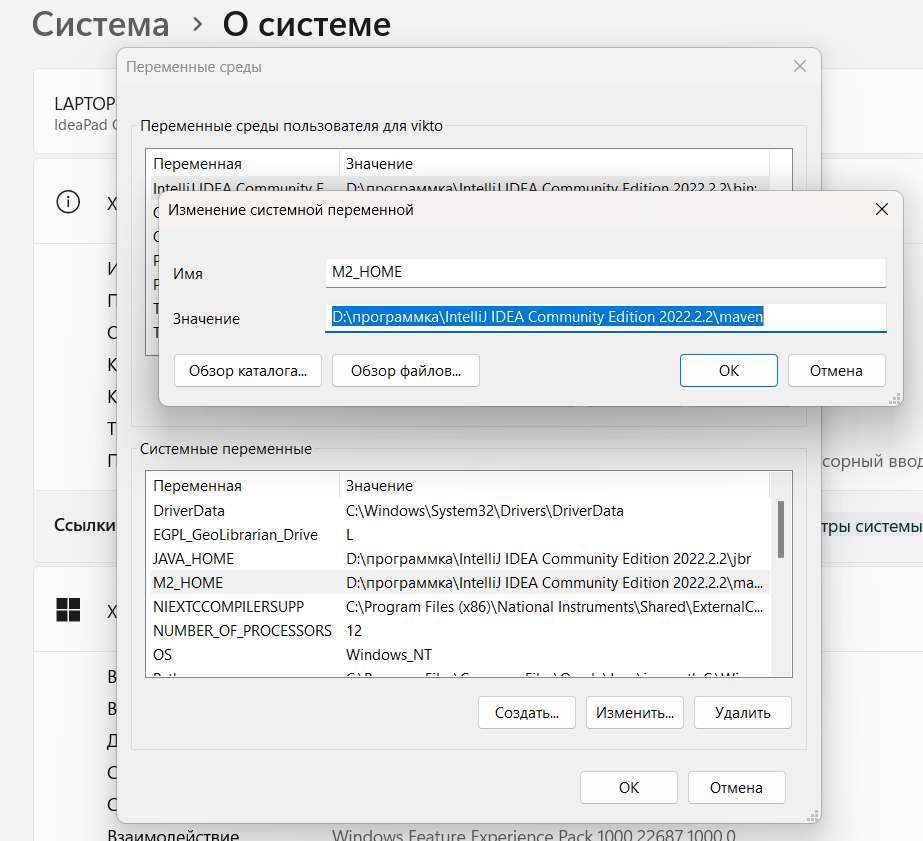


Рисунок 1 – Настройка среды

Проверка корректности установки показана рисунке 2.

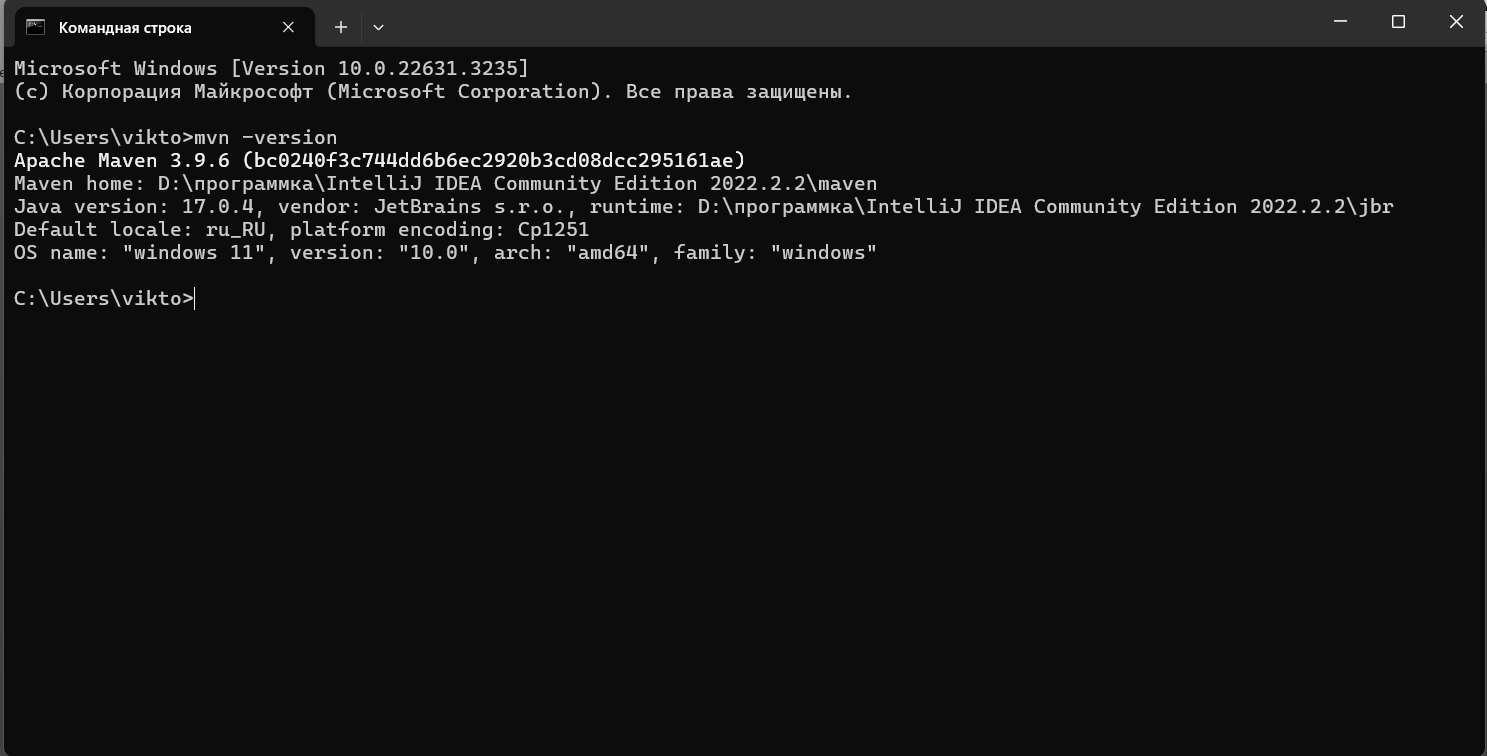


Рисунок 2 – Проверка работы Maven

II. Создание веб-приложения.

Выполнена команда для генерации шаблона приложения и дальнейшей разработки. Результат приведен на рисунке 3.

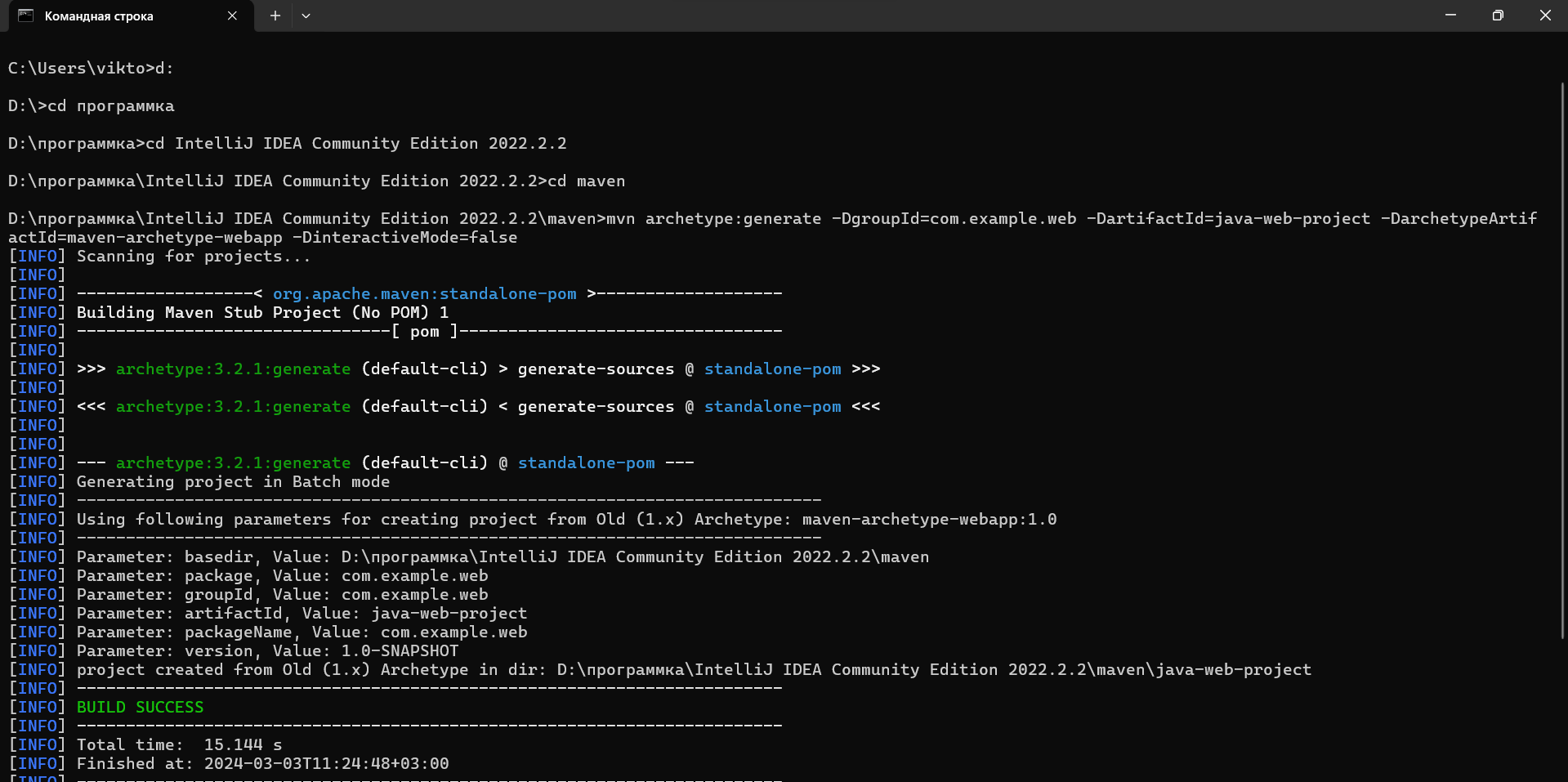


Рисунок 3 – Генерация шаблона приложения

Изучение структуры шаблона приложения и создание папки java показано на рисунке 4.

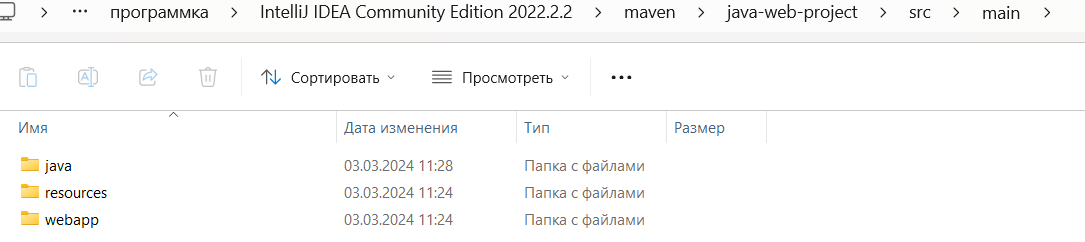
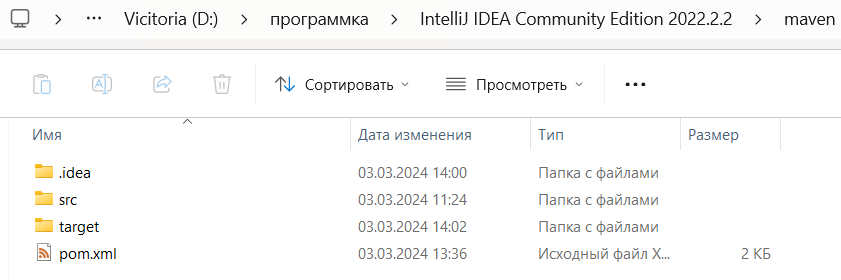


Рисунок 4 – Структура шаблона приложения

Сборка созданного проекта и проверка его корректности сборки, вызывав команду mvn clean install, отображена на рисунке 5 со статусом BUILD SUCCESS. Это означает, что приложение правильно создано и успешно компилируется в готовую сборку, которая является конечным этапом разработки приложения.

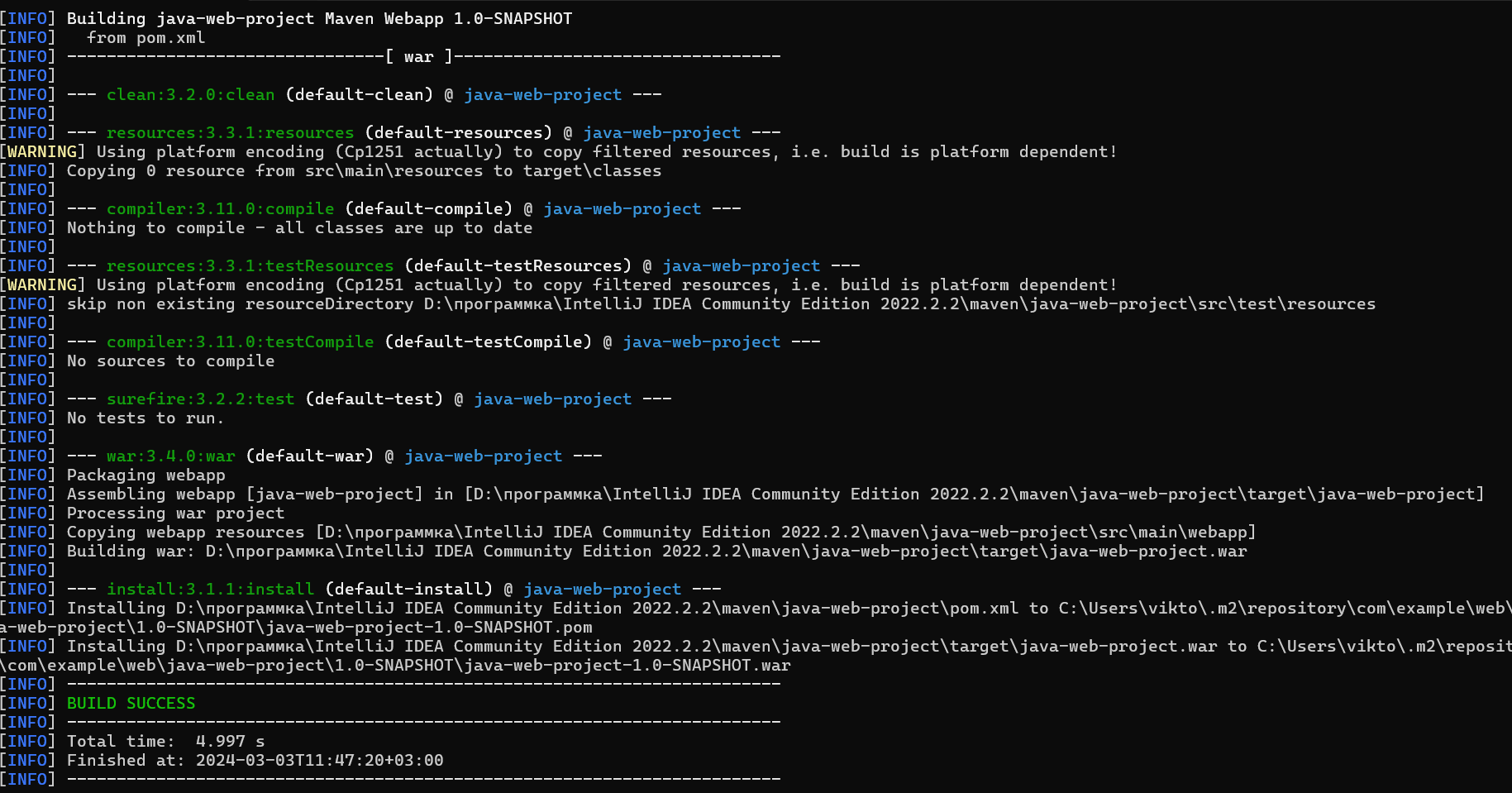


Рисунок 5 – Сборка maven-проекта

Файлы созданного веб-приложения показаны на рисунке 6.

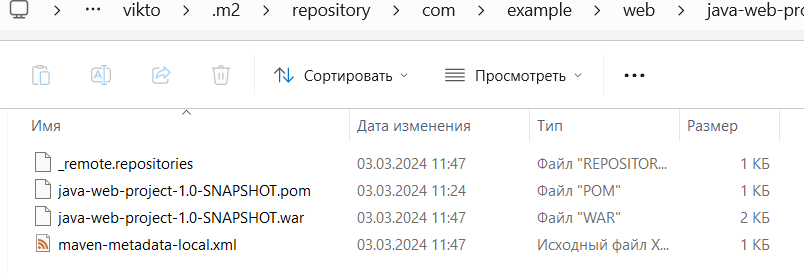


Рисунок 6 – Структура репозитория

III. Разработка и наполнение проекта.

Подключен дополнительный плагин для запуска веб-сервера jetty с помощью maven. Для этого открыт файл pom.xml и в раздел <build>… </build> добавлен плагин jetty.

<plugins>  
 <plugin>  
 <groupId>org.eclipse.jetty</groupId>  
 <artifactId>jetty-maven-plugin</artifactId>  
 <version>9.4.54.v20240208</version>  
 <configuration>  
 <scanIntervalSeconds>10</scanIntervalSeconds>  
 <connectors>  
 <connector  
 implementation="org.mortbay.jetty.nio.SelectChannelConnector">  
 <port>8080</port>  
 <maxIdleTime>60000</maxIdleTime>  
 </connector>  
 </connectors>  
 <stopKey/>  
 <stopPort/>  
 </configuration>  
 </plugin>  
</plugins>

Запущен веб-сервер jetty. Для этого используется команда mvn jetty:run в терминале. По умолчанию сервер запускается по порту 8080. По успешному запуску из командной строки, выведется сообщение об успешном запуске сервера. Результат показан на рисунке 7.

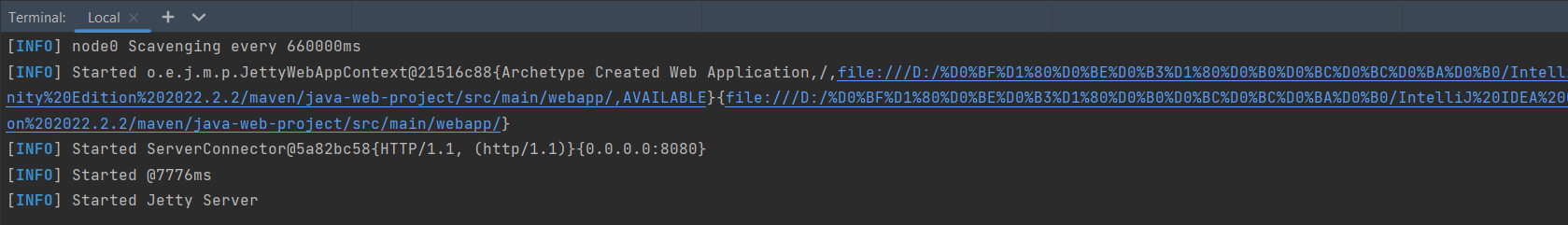


Рисунок 7 – Сообщение об успешном запуске сервера

Далее запущен веб-браузер, открыта ссылка http://localhost:8080. Таким образом, успешно отобразилась отработанная страница веб-сервера, как видно на рисунке 8.

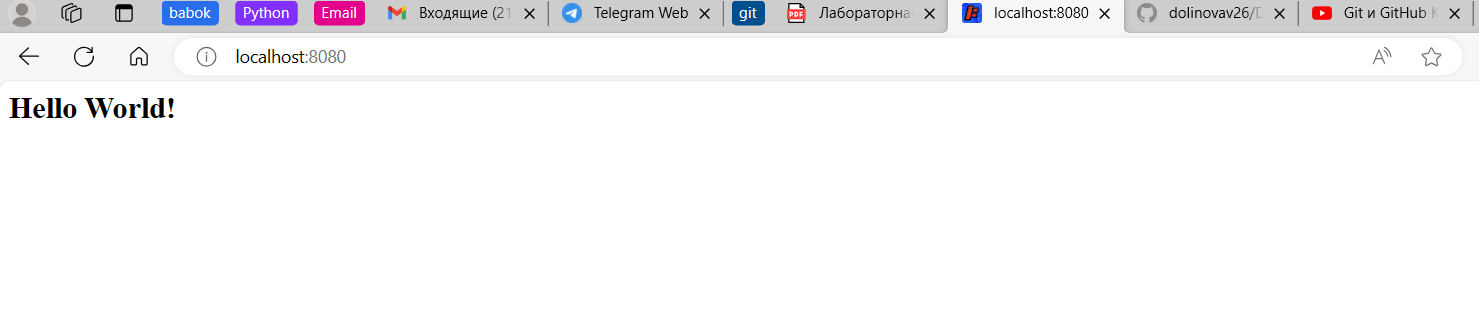


Рисунок 8 – Отображение страницы в веб-браузере