EXPERIMENT-5.1

Name: Jayanaath S　　　　Subject: Full Stack

Section: 23BCC-1　　　　Subject Code:23CSP-339

UID: 23BCC70022　　　　Date: 8-10-2025

- **Aim:**

  To develop and deploy a production-ready React application using Docker with a **multi-stage build**, optimizing the image size by separating the **build environment (Node.js)** from the **runtime environment (Nginx)**.

- **Theory:**
  Docker enables applications to run in isolated containers that package code, dependencies, and environment settings together. In a multi-stage Docker build, multiple FROM statements are used to create lightweight, efficient images by performing the build and runtime processes in separate stages.
  For a React application, the first stage uses a Node.js base image to install dependencies and build static production files using npm run build. The second stage uses an Nginx image to serve these optimized static files. This approach minimizes image size, removes unnecessary build dependencies, and enhances deployment efficiency and scalability.

- **Code:**

1. Project Setup

```
npx create-react-app my-react-app
cd my-react-app
```

2. .dockerignore

```
node_modules
build
.dockerignore
Dockerfile
.git
.gitignore
README.md
npm-debug.log
```

3. Dockerfile

```
# ---------- Stage 1: Build the React app ----------
FROM node:18 AS build

WORKDIR /app

COPY package*.json ./
RUN npm install

COPY . .
RUN npm run build

# ---------- Stage 2: Serve app using Nginx ----------
FROM nginx:alpine

RUN rm -rf /usr/share/nginx/html/*
COPY --from=build /app/build /usr/share/nginx/html

EXPOSE 80
CMD ["nginx", "-g", "daemon off;"]
```

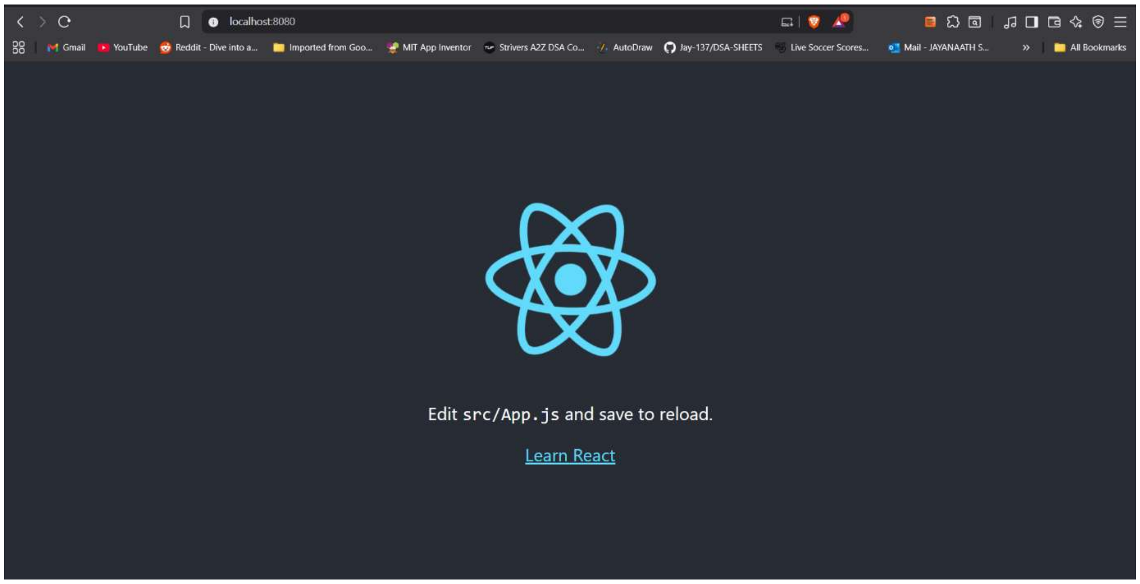
4. Build and Run Commands

```
# Build Docker image
docker build -t my-react-app .

# Run the container
docker run -d -p 8080:80 my-react-app
```

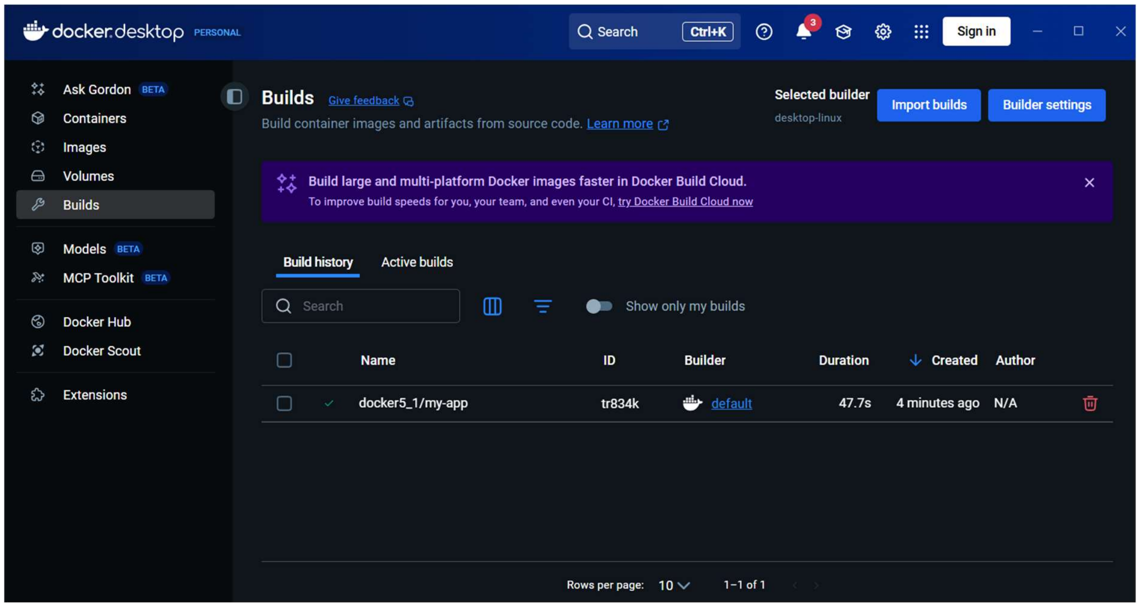
- **Result**
  - The React app is successfully served at http://localhost:8080.
  - The final Docker image size is significantly smaller compared to a single-stage build.
  - The build and runtime stages are clearly separated, ensuring an optimized production-ready image.

- **Output:**



Edit `src/App.js` and save to reload.

Learn React



Builds

Build container images and artifacts from source code. Learn more

Selected builder
desktop-linux

Build large and multi-platform Docker images faster in Docker Build Cloud.
To improve build speeds for you, your team, and even your CI, try Docker Build Cloud now

Name	ID	Builder	Duration	Created	Author
docker5_1/my-app	tr834k	default	47.7s	4 minutes ago	N/A

Rows per page: 10 1–1 of 1