Exp 5.1  
Dockerize a React Application with Multi-Stage Build.

FROM node:18-alpine AS build

# Set working directory

WORKDIR /app

# Install dependencies separately to leverage caching

COPY package\*.json ./

RUN npm install --legacy-peer-deps

# Copy rest of the application

COPY . .

# Build React app

RUN npm run build

# ---------- Stage 2: Production ----------

FROM nginx:stable-alpine

# Copy built React app to Nginx html folder

COPY --from=build /app/build /usr/share/nginx/html

# Copy custom Nginx config

COPY nginx.conf /etc/nginx/conf.d/default.conf

# Expose port 80

EXPOSE 80

# Start Nginx

CMD ["nginx", "-g", "daemon off;"]  
  
server {

listen 80;

location / {

root /usr/share/nginx/html;

index index.html index.htm;

try\_files $uri /index.html;

}

error\_page 500 502 503 504 /index.html;

}  
  
  
# Build image

docker build -t react-app .

# Run container

docker run -p 3000:80 react-app

output:

$ docker build -t react-app .

**Sample Output**

[+] Building 33.7s (12/12) FINISHED

=> [internal] load build definition from Dockerfile 0.1s

=> [internal] load .dockerignore 0.0s

=> [internal] load metadata for docker.io/library/node:18-alpine 1.2s

=> [internal] load metadata for docker.io/library/nginx:stable-alpine 1.1s

=> [build 1/6] FROM docker.io/library/node:18-alpine@sha256:... 0.0s

=> [build 2/6] WORKDIR /app 0.0s

=> [build 3/6] COPY package\*.json ./ 0.0s

=> [build 4/6] RUN npm install --legacy-peer-deps 10.5s

=> [build 5/6] COPY . . 0.2s

=> [build 6/6] RUN npm run build 17.0s

=> [stage-1 1/3] FROM docker.io/library/nginx:stable-alpine@sha256:... 0.0s

=> [stage-1 2/3] COPY --from=build /app/build /usr/share/nginx/html 0.1s

=> [stage-1 3/3] COPY nginx.conf /etc/nginx/conf.d/default.conf 0.0s

=> exporting to image 3.5s

=> => exporting layers 3.4s

=> => writing image sha256:abcd1234... 0.0s

=> => naming to docker.io/library/react-app 0.0s

Successfully built react-app

**▶️ Run Container**

$ docker run -p 3000:80 react-app

**Sample Output**

/docker-entrypoint.sh: Configuration complete; ready for start up

2025/09/26 12:10:10 [notice] 1#1: using the "epoll" event method

2025/09/26 12:10:10 [notice] 1#1: nginx/1.25.3

2025/09/26 12:10:10 [notice] 1#1: built by gcc 12.2.1

2025/09/26 12:10:10 [notice] 1#1: OS: Linux 5.15.0

2025/09/26 12:10:10 [notice] 1#1: start worker processes