🐳 Docker Cheatsheet for Node.js App

📦 Build Docker Image

```
# 🔨 Build the Docker image and tag it as 'my-node-app'
docker build -t my-node-app .
```

🚀 Run the Container

```
# 🔨 Run the container and expose it on localhost:8000
docker run -p 8000:8000 my-node-app
```

```
# 🔨 Run the container in detached/background mode
docker run -d -p 8000:8000 my-node-app
```

```
# 🔨 Run an interactive Ubuntu container (good for testing)
docker run -it ubuntu
```

📋 Inspect Containers and Images

```
# 🔨 List running containers
docker ps
```

```
# 🔨 List all containers (running + stopped)
docker ps -a
```

```
# 🔨 List all Docker images available locally
docker images
```

🛑 Stop and Remove Containers

```
# 🩹 Stop a running container
docker stop <container_id_or_name>
```

```
# 🩹 Remove a stopped container
docker rm <container_id_or_name>
```

```
# 🩹 Remove a Docker image by name or ID
docker rmi my-node-app
```

🔍 View Logs and Inspect Details

```
# 🩹 View logs from a container (stdout/stderr)
docker logs <container_id>
```

```
# 🩹 Inspect detailed info of a container
docker inspect <container_id>
```

🧼 Clean Up Unused Resources

```
# 🩹 Remove all stopped containers
docker container prune
```

```
# 🩹 Remove all unused Docker images
docker image prune
```

```
# 🩹 Remove all unused data (containers, networks, images, etc.)
docker system prune
```

📓 Bonus: Access Bash in a Running Container

```

```

```
# 🩹 Access container shell (if bash is installed inside)
docker exec -it <container_id> /bin/bash
```
