Node.js Application on Docker

- ## Steps to create

mkdir nodejs

cd odejs

ls

touch file index.js package.json Dockerfile

ls

```
Last login: Wed Dec 24 03:23:18 2025 from 18.206.107.28
ubuntu@ip-172-31-27-216:~$ mkdir nodejs
ubuntu@ip-172-31-27-216:~$ cd nodejs
ubuntu@ip-172-31-27-216:~/nodejs$ ls
ubuntu@ip-172-31-27-216:~/nodejs$ touch file index.js package.json Dockerfile
ubuntu@ip-172-31-27-216:~/nodejs$ ls
Dockerfile  file  index.js  package.json
```

vi index.js

```
const express = require("express");
const app = express();

app.get("/", (req, res) => {
  res.send("Hello from Dockerized Node.js 🚀");
});

const PORT = process.env.PORT || 3000;
app.listen(PORT, () => {
  console.log(`Server running on port ${PORT}`);
});

~
~
```

vi package.json

```
{
  "name": "node-docker-app",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "start": "node index.js"
  },
  "dependencies": {
    "express": "^4.18.2"
  }
}
```

vi Dockerfile

```
FROM node:20-alpine

WORKDIR /nodejs

COPY package*.json ./

RUN npm install --production

COPY . .

EXPOSE 3000

CMD ["npm", "start"]
~
~
~
```

sudo su

docker build .

```
ubuntu@ip-172-31-27-216:~/nodejs$ sudo su
root@ip-172-31-27-216:/home/ubuntu/nodejs# docker build .
DEPRECATED: The legacy builder is deprecated and will be removed in a future release.
            Install the buildx component to build images with BuildKit:
            https://docs.docker.com/go/buildx/

Sending build context to Docker daemon  4.608kB
Step 1/7 : FROM node:20-alpine
20-alpine: Pulling from library/node
1074353eec0d: Pulling fs layer
8d06ba6946d1: Pulling fs layer
cb3325e64457: Pulling fs layer
fd1849a5c548: Pulling fs layer
fd1849a5c548: Waiting
1074353eec0d: Verifying Checksum
1074353eec0d: Download complete
cb3325e64457: Verifying Checksum
cb3325e64457: Download complete
fd1849a5c548: Verifying Checksum
fd1849a5c548: Download complete
1074353eec0d: Pull complete
8d06ba6946d1: Verifying Checksum
8d06ba6946d1: Download complete
8d06ba6946d1: Pull complete
cb3325e64457: Pull complete
fd1849a5c548: Pull complete
Digest: sha256:658d0f63e501824d6c23e06d4bb95c71e7d704537c9d9272f488ac03a370d448

  i-0141cc92a35a88c78 (Docker)
```

docker images

```
root@ip-172-31-27-216:/home/ubuntu/nodejs# docker images
REPOSITORY          TAG          IMAGE ID       CREATED          SIZE
<none>              <none>       9a2c292bfcb3   8 seconds ago    142MB
<none>              <none>       430800742503   56 minutes ago   133MB
<none>              <none>       0fca0281cc0a   59 minutes ago   122MB
flask1              latest       cef2bc8e2d9b   18 hours ago     133MB
<none>              <none>       24d2bd61ce35   18 hours ago     133MB
<none>              <none>       a89a6427ef3e   18 hours ago     133MB
<none>              <none>       5626aa053d68   18 hours ago     122MB
<none>              <none>       548b8dbbd32f   18 hours ago     133MB
```

docker run -d -p 3000:3000 --name nodejs 9a2c292bfcb3

```
root@ip-172-31-27-216:/home/ubuntu/nodejs# docker run -d -p 3000:3000 --name nodejs 9a2c292bfcb3
1e0cd5e3673952c5a6961d13951704ae44ed4b9f12d4c3c67fb852b3ec4bfd93
```

docker ps -a

```
root@ip-172-31-27-216:/home/ubuntu/nodejs# docker ps -a
CONTAINER ID   IMAGE          COMMAND                  CREATED          STATUS
1e0cd5e36739   9a2c292bfcb3   "docker-entrypoint.s…"   7 seconds ago    Up 7 seconds
9172c02833b7   430800742503   "python file.py"         54 minutes ago   Exited (0) 40 minutes ago
b5e309f9d093   0fca0281cc0a   "/bin/sh -c 'pip ins…"   About an hour ago Exited (1) About an hour ago
```

docker exec -it nodejs sh

ls

exit

```
root@ip-172-31-27-216:/home/ubuntu/nodejs# docker exec -it nodejs sh
/nodejs # ls
Dockerfile          file                 index.js              node_modules
/nodejs # exit
```
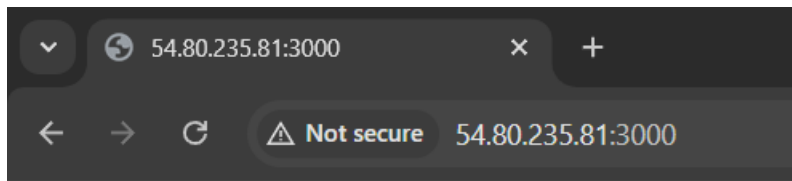
cat index.js

```
root@ip-172-31-27-216:/home/ubuntu/nodejs# cat index.js
const express = require("express");
const app = express();

app.get("/", (req, res) => {
  res.send("Hello from Dockerized Node.js 🚀");
});

const PORT = process.env.PORT || 3000;
app.listen(PORT, () => {
  console.log(`Server running on port ${PORT}`);
});
```

Go to instance copy public ip and paste on incognito mode public ip:3000(port no of node.js)

Hello from Dockerized Node.js 🚀