

ASSIGNMENT:1


Create a simple Docker container for a Node.js web application.


Step1: Create a Docker Hub Account:

If you don't already have a Docker Hub account, you need to create one. Go to the Docker Hub website (<https://hub.docker.com/>) and sign up for a new account.



Create your account

 Continue with Google

 Continue with GitHub

OR

Email
sunnybhushan9185@gmail.com

Username
sunny2883

Password
..... 

☐ Send me occasional product updates and announcements.

This site is protected by reCAPTCHA and the Google [Privacy Policy](#) and [Terms of Service](#) apply.

Sign up

Step2: Prepare Your Node.js Web Application:

Ensure your Node.js web application is ready and functional. It should have a package.json file containing all dependencies and a script to start the server.

Make sure your application listens on a port that can be exposed to the host. Typically, this would be port 3000 for Node.js applications.

Step3:Create a Dockerfile:

In the root directory of your Node.js application, create a file named Dockerfile (without any file extension).

Define the necessary instructions in the Dockerfile to build your Docker image. Here's a basic example to get you started:

FROM node:14-alpine

WORKDIR /app

Copy package.json and package-lock.json to the working directory

COPY package*.json ./

Install dependencies

RUN npm install

Copy the rest of the application code

COPY . .

Expose the port on which the app will run

EXPOSE 3000

Command to run the application

CMD ["node", "server.js"]

Step:4 Build the Docker Image:

Open a terminal or command prompt.

Navigate to the directory containing your Dockerfile and the Node.js application.

Run the following command to build the Docker image:

docker build -t sunny2883/nodejsapp:v2 .

```
-----  
PS C:\Users\91725\Desktop\views> docker build -t sunny2883/nodejsapp:v2 .  
[+] Building 2.1s (11/11) FINISHED                                docker:default
```

Step:5 Log in to Docker Hub:

In your terminal or command prompt, log in to Docker Hub using the docker login command:

docker login

Enter your Docker Hub username and password when prompted.

```
-----  
PS C:\Users\91725\Desktop\views> docker login  
Log in with your Docker ID or email address to push and pull images from Docker Hub. If  
you don't have a Docker ID, head over to https://hub.docker.com/ to create one.  
You can log in with your password or a Personal Access Token (PAT). Using a limited-scope  
PAT grants better security and is required for organizations using SSO. Learn more at  
https://docs.docker.com/go/access-tokens/
```

```
Username: sunny2883
```

```
Password:
```

```
Login Succeeded
```

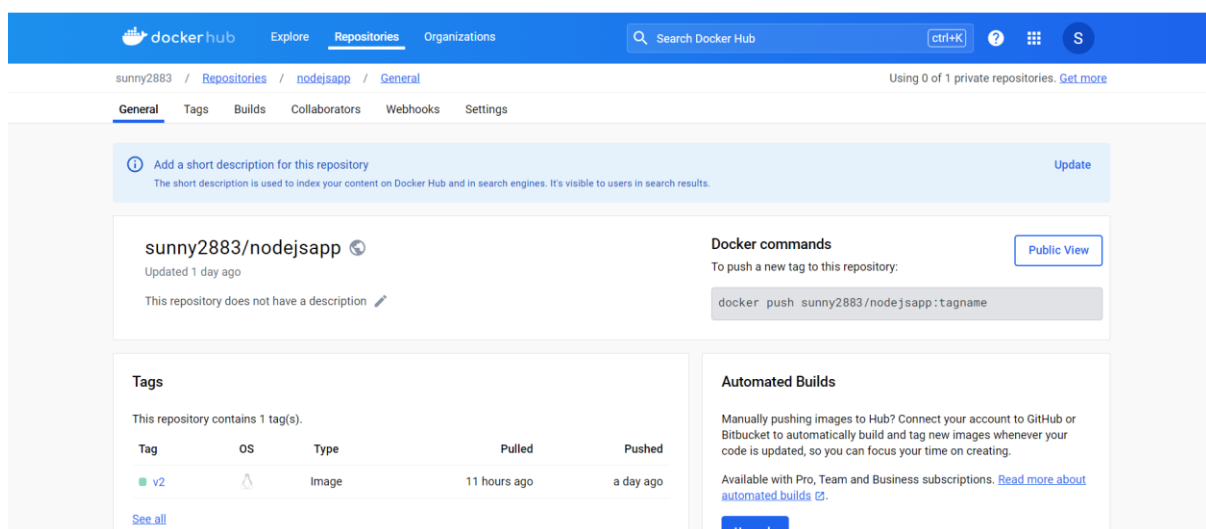
```
PS C:\Users\91725\Desktop\views> █
```

Step6: Push the Docker Image to Docker Hub:

Push your Docker image to Docker Hub using the docker push command:

docker push sunny2883/nodejsapp:v2

```
PS C:\Users\91725\Desktop\views> docker push sunny2883/nodejsapp:v2
The push refers to repository [docker.io/sunny2883/nodejsapp]
9ccbda9d5dad: Pushed
5f1aa029de8c: Pushed
7f642d591162: Pushed
0eac72462b23: Pushed
31f710dc178f: Pushed
a599bf3e59b8: Pushed
e67e8085abae: Pushed
f1417ff83b31: Pushed
v2: digest: sha256:63f49ea84ed848e393b67adcb9fec5d8a89405570621943583655b656b
435bee size: 1992
```



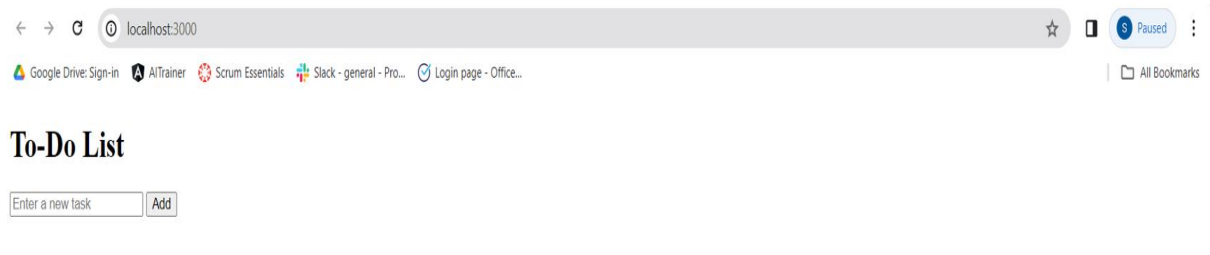
Step7: Run a container based on the pulled image:

```
docker run -d -p 3000:3000 sunny2883/nodejsapp:v2
```

```
PS C:\Users\91725\Desktop\views> docker run -d -p 3000:3000 sunny2883/nodejsa
pp:v2
5994a5e666d4ce871d1a96d82e8198914440b013b3d113eb9cc1ebe880f28e96
```

Step8: Access Your Application from Any Browser:

Open a web browser on that machine and navigate to `http://<machine_ip>:3000` to access your Node.js application running inside the Docker container.



Command to push the image:

[docker push sunny2883/nodejsapp:v2](#)