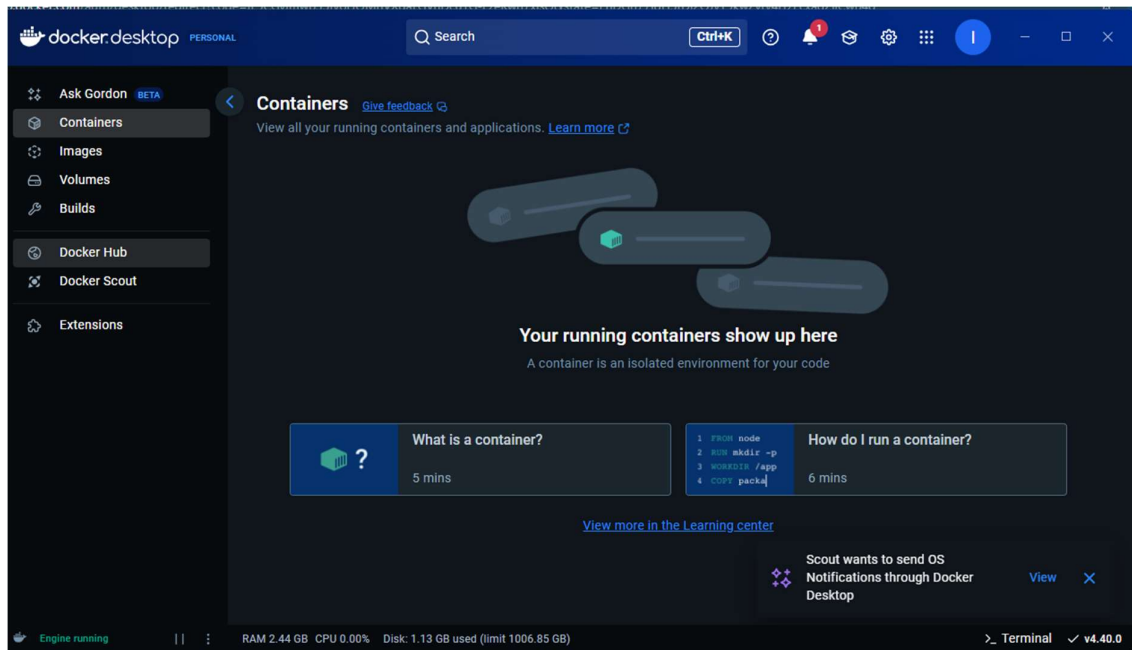


LAB 6

Step 1: Download and install DOCKER desktop



Step 2: RUN some basic Docker commands

```
Login Succeeded
PS C:\Users\CSE> docker images
REPOSITORY TAG          IMAGE ID    CREATED    SIZE
PS C:\Users\CSE> docker pull nginx
Using default tag: latest
latest: Pulling from library/nginx
d3dc5ec71e9d: Pull complete
3e544d53ce49: Pull complete
254e724d7786: Pull complete
4f21ed9ac0c0: Pull complete
40a6e9f4e456: Pull complete
913115292750: Pull complete
d38f2ef2d6f2: Pull complete
Digest: sha256:c15da6c91de8d2f436196f3a768483ad32c258ed4e1beb3d367a27ed67253e66
Status: Downloaded newer image for nginx:latest
docker.io/library/nginx:latest

PS C:\Users\CSE> docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
2dd9859d2be2   myapp:1.0 "docker-entrypoint.s..." 22 minutes ago Up 22 minutes 0.0.0.0:3000->3000/tcp    gallant_hermann

PS C:\Users\CSE> docker ps -a
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                    NAMES
2dd9859d2be2   myapp:1.0 "docker-entrypoint.s..." 22 minutes ago Up 22 minutes 0.0.0.0:3000->3000/tcp    gallant_hermann
750efd3cc97d   myapp:1.0 "docker-entrypoint.s..." 23 minutes ago Exited (0) 22 minutes ago nice_nobel
```

LAB 6

```
C:\Users\201LAB>docker images
REPOSITORY    TAG       IMAGE ID      CREATED       SIZE
nginx         latest    c15da6c91de8  2 weeks ago   279MB

C:\Users\201LAB>docker rmi nginx:latest
Error response from daemon: No such image: nginx:latest

C:\Users\201LAB>docker rmi nginx:latest
Untagged: nginx:latest
Deleted: sha256:c15da6c91de8d2f436196f3a768483ad32c258ed4e1beb3d367a27ed67253e66

C:\Users\201LAB>docker images
REPOSITORY    TAG       IMAGE ID      CREATED       SIZE
```

Step 3: Create a project on VSCode

Inside this folder, create two files:

- `app.js`
- `package.json`

Step 2: Add Code to package.json

In package.json, add the following content:

```
{
  "name": "hello-world-app",
  "version": "1.0.0",
  "main": "app.js",
  "scripts": {
    "start": "node app.js"
  },
  "dependencies": {
    "express": "^4.17.1"
  }
}
```

App.js

```
const express = require('express');

const app = express();

const PORT = 3000;
```

LAB 6

```
app.get('/', (req, res) => {  
  res.send('Hello, World!');  
});
```

```
app.listen(PORT, () => {  
  console.log(`Server is running on http://localhost:${PORT}`);  
});
```

In terminal run : npm install

Step 4: Create a Dockerfile in the same project folder

```
# Use official Node.js runtime as a parent image  
FROM node:14  
  
# Set working directory in the container  
WORKDIR /usr/src/app  
  
# Copy package.json and package-lock.json  
COPY package*.json ./  
  
# Install dependencies  
RUN npm install  
  
# Copy the rest of the application source code  
COPY . .  
  
# Expose port 3000  
EXPOSE 3000  
  
# Run the app when the container launches  
CMD ["npm", "start"]
```

Step 5 : Build a docker image

```
PS C:\Users\CSE\Desktop\1ds23cs103> docker build -t myapp:1.0 .  
[+] Building 30.3s (11/11) FINISHED          docker:desktop-linux  
=> [internal] load build definition from Dockerfile 0.0s  
=> => transferring dockerfile: 440B 0.0s  
=> [internal] load metadata for docker.io/library/node:14 3.0s  
=> [auth] library/node:pull token for registry-1.docker.io 0.0s  
=> [internal] load .dockerignore 0.1s  
=> => transferring context: 2B 0.0s  
=> [1/5] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa 16.7s  
=> => resolve docker.io/library/node:14@sha256:a158d3b9b4e3fa8 0.1s
```

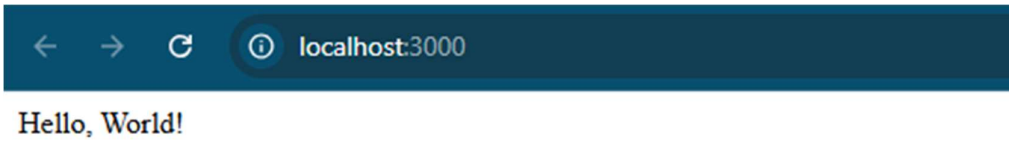
Step 6 : Run the Docker Container

```
PS C:\Users\CSE\Desktop\1ds23cs103> docker run -p 3000:3000 myapp:1.0  
  
> hello-world-app@1.0.0 start /usr/src/app  
> node app.js  
  
Server is running on http://localhost:3000
```

Step 7 : Access the Application by visiting

<http://localhost:3000>

Step 8 : The output should be

A screenshot of a web browser window. The address bar shows 'localhost:3000' with a blue information icon to its left. Below the address bar, the text 'Hello, World!' is displayed in a simple black font.

← → ↻ ⓘ localhost:3000

Hello, World!