**AIM:-**  
To install and understand the working of **Node.js**, **Express.js**, and **Docker**, and to perform the necessary environment setup to build and run backend web applications using containers.

**PROCEDURE : -**

**1. Node.js Installation**

**Node.js** is an open-source, cross-platform JavaScript runtime environment that executes JavaScript code outside of a browser. It is widely used for building backend services such as REST APIs, web servers, and real-time applications.

**Steps to Install Node.js:**

1. **Download Node.js**:

* Go to the official Node.js website: https://nodejs.org/
* Download the **LTS (Long Term Support)** version suitable for your OS (Windows, Linux, or macOS).

1. **Install Node.js**:

* Run the downloaded installer and follow the on-screen instructions.
* Ensure that npm (Node Package Manager) is included in the installation.

1. **Verify Installation**:  
    Open a terminal/command prompt and type:

* node -v
* npm -v

This will display the installed versions of Node.js and npm, confirming successful installation.

**Why Node.js?**

* Non-blocking I/O model
* Fast execution with V8 engine
* Perfect for scalable server-side development

**2. Express.js Installation**

**Express.js** is a fast, unopinionated, and minimalist web framework for Node.js used for building RESTful APIs and web applications.

**Steps to Install Express.js:**

1. **Create a Project Directory**:

* mkdir express-app
* cd express-app

1. **Initialize the Project**:  
   Generate a package.json file using:

* npm init -y

1. **Install Express Framework**:  
   Install Express locally in your project:

* npm install express

1. **Install Nodemon:**  
   Nodemon automatically restarts your server on file changes, useful for development:

* npm install -g nodemon

1. **Create a Sample Express Server**:  
   Create a file named app.js and write the following code:

const express = require('express');

const app = express();

const PORT = 3000;

app.get('/', (req, res) => {

res.send('Hello, Express.js!');

});

app.listen(PORT, () => {

console.log(`Server is running on http://localhost:${PORT}`);

});

1. **Run the Server**:

* node app.js

**OR (if using nodemon) : -**

* nodemon app.js

**Why Express.js?**

* Simplifies routing and middleware handling
* Lightweight and fast
* Large ecosystem of middleware packages

**3. Docker Installation and Setup**

**Docker** is an open-source platform used to develop, ship, and run applications in containers. Containers are lightweight, executable software packages that include all dependencies.

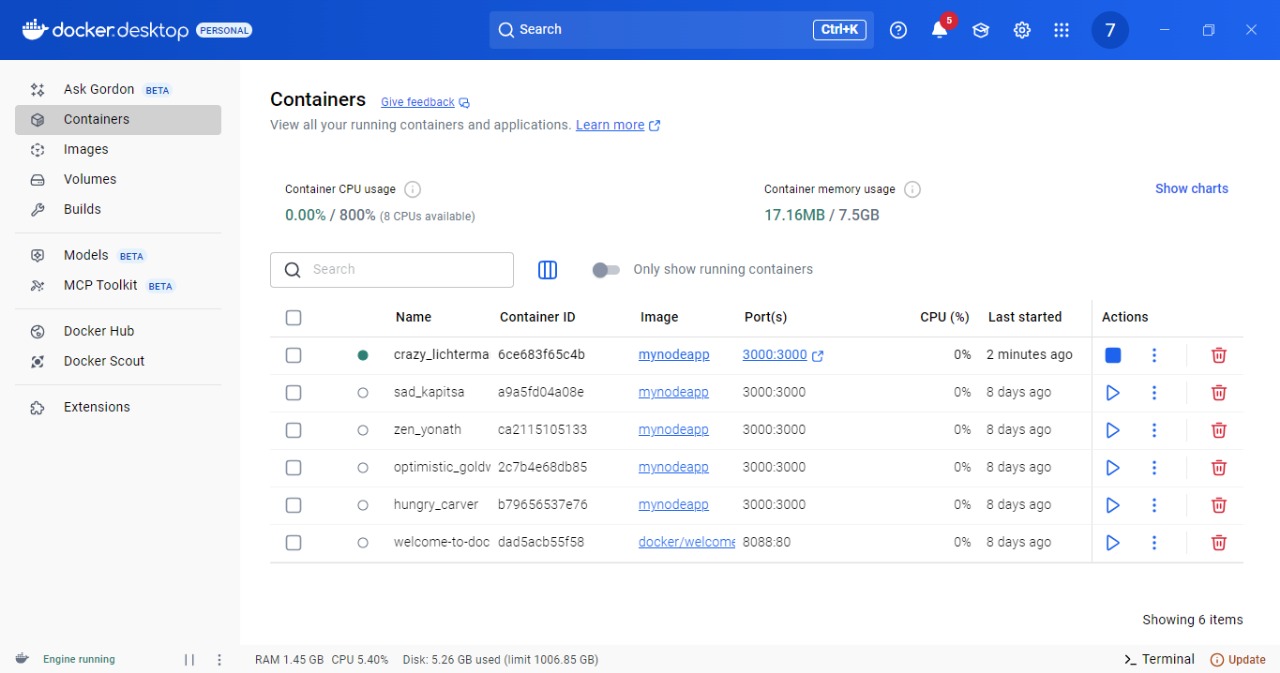
**Steps to Install Docker:**

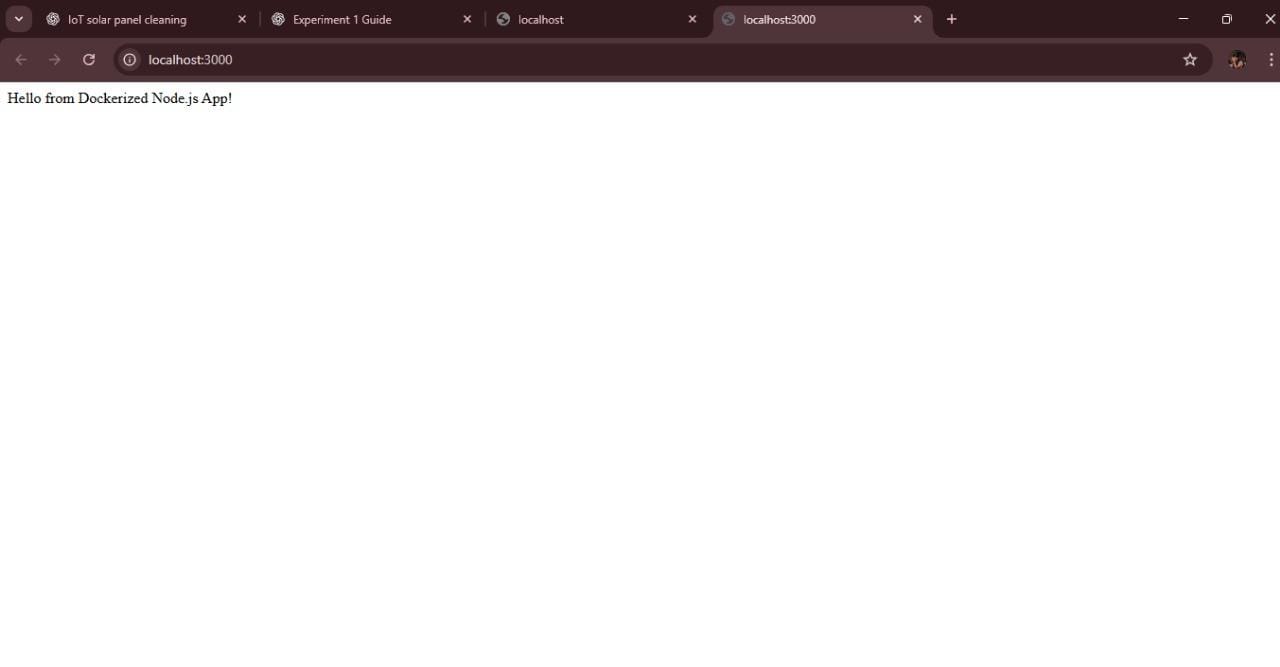
1. **Download Docker**:
   * Visit: https://www.docker.com/products/docker-desktop
   * Choose the installer for your operating system.
2. **Install Docker Desktop**:
   * Run the installer and follow the steps.
   * After installation, restart your system if required.
3. **Verify Docker Installation**:  
   Open your terminal and run:

* docker --version

This should print the installed Docker version.

**OUTPUT : -**





**RESULT : -**

The installation and setup of **Node.js**, **Express.js**, and **Docker** were successfully completed.