1️⃣ Difference Between Node.js and Express.js

✅ What is Node.js?

Node.js is a **JavaScript runtime environment**.

It allows JavaScript to run **outside the browser** (on the server).

📌 Example:

```
console.log("Hello from server");
```

Before Node → JS only worked inside browser
After Node → JS works in backend/server

✅ What is Express.js?

Express.js is a **framework built on top of Node.js**.

It helps us create:

- APIs
- Routes
- Middleware
- Server easily

🔥 Difference

Node.js	Express.js
Runtime environment	Framework
Runs JS on server	Makes server development easier
Low-level	High-level

🎯 Why We Use Express Instead of Only Node?

Without Express:

```
const http = require("http");
```

Everything manual.

With Express:

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

const app = express();
```

Much easier, cleaner, faster development.

2️⃣ What is .env ?

✅ What is it?

.env is a file used to store:

- Secret keys
- Port number
- Database URL
- JWT secret
- API keys

🎯 Why we use it?

Security 🔐

Instead of:

```
const password = "mysecret123";
```

We use:

```
DB_PASSWORD=mysecret123
```

Then in code:

```
require("dotenv").config();

process.env.DB_PASSWORD
```

💻 Developer Purpose

- Keeps sensitive data safe
- Different values for dev & production

- Easy configuration management

3️⃣ What is listen() ?

✅ What is it?

app.listen() starts the server.

```
app.listen(5000, () => {

  console.log("Server running");

});
```

🎯 Why we use it?

Without listen():
Server won’t start.

It tells Node:
👉 “Start listening to requests on this port”

4️⃣ What is Express?

✅ Definition:

Express is a **minimal and flexible web framework for Node.js**

It handles:

- Routing
- Middleware
- HTTP requests
- APIs

Official website: https://expressjs.com

5️⃣ Why We Use Express?

Because it:

- Simplifies server creation

- Makes REST API easy
- Supports middleware
- Fast & lightweight
- Huge community support

Example:

```
app.get("/users", (req, res) => {
  res.send("All users");
});
```

Very simple 👍

6️⃣ What is Nodemon?

✅ What is it?

nodemon automatically restarts server when you change file.

Without nodemon:
Every time:

```
CTRL + C
node index.js
```

With nodemon:

```
nodemon index.js
```

It auto restarts 🔥

🎯 Why we use it?

For developer productivity.

It saves time during development.

Install:

```
npm install -g nodemon
```

7️⃣ What is Port Number and Localhost?

✅ What is Port?

A port is like a **door number for your server**.

Example:

http://localhost:5000

5000 → port number

✅ What is localhost?

localhost means:
👉 Your own computer

IP address:

127.0.0.1

So:

http://localhost:5000

Means:
👉 My computer server running on port 5000

8️⃣ What is API?

✅ Full Form:

Application Programming Interface

Simple meaning:

API = A bridge between frontend and backend

Example Flow:

Frontend:

```
fetch("/users")
```

Backend:

```
app.get("/users", (req, res) => {

  res.json(users);
```

});

API connects both.

🎯 Why we use API?

- Send data
- Receive data
- Login
- Register
- Fetch products
- Payment

All done using APIs.

9️⃣ Flow of Express + Frontend Process

Let’s understand full real developer flow 🔥

Step 1:

User clicks button in React

Step 2:

React sends request

fetch("http://localhost:5000/api/login")

Step 3:

Express receives request

app.post("/api/login")

Step 4:

Backend checks DB

Step 5:

Backend sends response

res.json({ success: true });

Step 6:

Frontend receives response

Full Flow Diagram:

User → Frontend → API Request → Express → Database → Response → Frontend → User

---

🔟 What is package.json?

✅ What is it?

package.json is the heart of your Node project.

It contains:

- Project name
- Version
- Dependencies
- Scripts
- Author

Example:

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

🎯 Why we use it?

- Manage packages
- Install dependencies
- Run scripts
- Share project with team

When someone runs:

```
npm install
```

It installs all dependencies from package.json.