API using node js

What is an API?

- **API** stands for Application Programming Interface.
- It allows communication between two software systems.
- APIs define endpoints for sending and receiving data.
- **Example:** A weather app uses an API to get real-time weather data.

Why Use Node.js for APIs?

- Fast and non-blocking I/O — great for real-time and data-heavy apps.
- Uses JavaScript (same language frontend/backend).
- Rich ecosystem with npm packages.
- Works well with JSON, the standard for web APIs.

Types of APIs in Node.js

- **REST API** – Commonly used, uses HTTP methods (GET, POST, PUT, DELETE).
- **GraphQL** – Query-based API, more flexible than REST.
- **Real-time APIs** – Use WebSockets (e.g., with Socket.io).

Tools & Frameworks

- **Express.js** – Most popular framework for building APIs in Node.js.
- **Body-parser** – Parses incoming request bodies.
- **Cors** – Enables cross-origin requests.
- **Postman** – Tool for testing APIs.

Sample REST API Structure

- GET /users – Fetch all users
- GET /users/:id – Fetch single user
- POST /users – Create new user
- PUT /users/:id – Update user
- DELETE /users/:id – Delete user

Example Code Snippet

```
const express = require('express');
const app = express();
app.use(express.json());

app.get('/api', (req, res) => {
  res.send('Hello from API!');
});

app.listen(3000, () => console.log('Server running on port 3000'));
```

Summary & Next Steps

- Node.js + Express is a powerful stack for building APIs.
- Easy to get started, flexible for complex apps.

Next Steps:

- Try creating a simple CRUD API.
- Explore middleware and error handling.
- Learn about authentication (JWT, OAuth).