

Example Task: Build a Simple Notes API

Name: Amita Narayanan Kutty

Objective: Create a Node.js API using Express to store and retrieve notes with GET /notes (list notes) and POST /notes (add a note).

Requirements

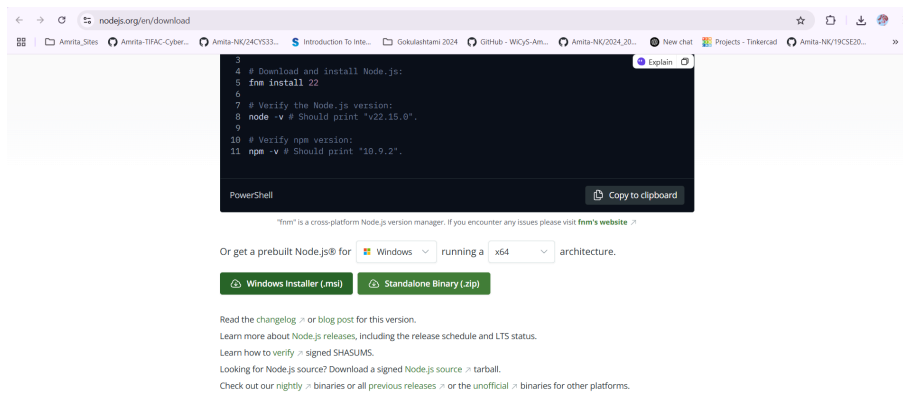
- **Node.js:** Installed from nodejs.org (LTS version, e.g., v20.x).
- **VS Code:** Installed from code.visualstudio.com.
- **Terminal:** Use VS Code's integrated terminal

Step-by-Step Instructions in VS Code:

Step 1: Set Up Node.js

1. Install Node.js:

- Download and install the LTS version from nodejs.org.



- Do the following

```
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> node -v
v20.9.0
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> npm -v
10.1.0
```

Step 2: Create Project Folder

1. **Open VS Code:**
 - Launch VS Code.
2. **Create a Folder:**
 - Click File > Open Folder and create a new folder named notes-api.
 - Open it in VS Code.
3. **Initialize Project:**
 - In the VS Code terminal, run:

```
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> npm init -y
Wrote to C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks\package.json:

{
  "name": "tasks",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "keywords": [],
  "author": "",
  "license": "ISC"
}
```

This creates a package.json file.

```
{ } package-lock.json
{ } package.json
```

4. Install Express:

- Run:

```
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> npm install express

added 66 packages, and audited 67 packages in 6s

14 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

This installs Express and creates a node_modules folder and package-lock.json.

```
JS index.js
{} package-lock.json
{} package.json
```

Step 3: Write the API Code

1. Create index.js:

- In VS Code, click File > New File, name it index.js, and save it in the notes-api folder.

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

let notes = [];

app.get('/notes', (req, res) => {
  res.json(notes);
});

app.post('/notes', (req, res) => {
  const note = { id: notes.length + 1, content: req.body.content };
  notes.push(note);
```

```

    res.status(201).json(note);
  });

app.listen(3000, () => {
  console.log('Server running at http://localhost:3000/');

  ○ });

```

2. Save the File:

- Press Ctrl + S or File > Save.

Step 4: Run the API

1. Start the Server:

- In the VS Code terminal, run:

```

Node.js v20.9.0
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> node index.js
Server running at http://localhost:3000/
█

```

2. Test the API:

- Open a new VS Code terminal tab (+ button in the terminal panel) and test:
- **GET** all notes:

```

PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> curl http://localhost:3000/notes

StatusCode      : 200
StatusDescription : OK
Content         : []
RawContent      : HTTP/1.1 200 OK
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 2
                  Content-Type: application/json; charset=utf-8
                  Date: Sat, 03 May 2025 18:30:59 GMT
                  ETag: W/"2-19Fw4VU07kr8Cv8lt4zaMC..."
Forms           : {}
Headers         : {[Connection, keep-alive], [Keep-Alive, timeout=5], [Content-Length, 2],
                  [Content-Type, application/json; charset=utf-8]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : mshtml.HTMLDocumentClass
RawContentLength : 2

```

- **POST** a note:

```
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> Invoke-WebRequest -Uri http://localhost:3000/notes -Method POST -Headers @{"Content-Type"="application/json"} -Body '{"content":"Buy milk"}'
```

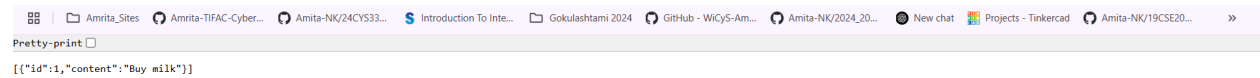
```
StatusCode      : 201
StatusDescription : Created
Content         : {"id":1,"content":"Buy milk"}
RawContent      : HTTP/1.1 201 Created
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 29
                  Content-Type: application/json; charset=utf-8
                  Date: Sat, 03 May 2025 18:32:10 GMT
                  ETag: W/"1d-CjJbajY2y1383N8..."
Forms           : {}
Headers         : {[Connection, keep-alive], [Keep-Alive, timeout=5], [Content-Length, 29],
                  [Content-Type, application/json; charset=utf-8]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : mshtml.HTMLDocumentClass
RawContentLength : 29
```

- **GET** again:

```
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> curl http://localhost:3000/notes
```

```
StatusCode      : 200
StatusDescription : OK
Content         : [{"id":1,"content":"Buy milk"}]
RawContent      : HTTP/1.1 200 OK
                  Connection: keep-alive
                  Keep-Alive: timeout=5
                  Content-Length: 31
                  Content-Type: application/json; charset=utf-8
                  Date: Sat, 03 May 2025 18:32:33 GMT
                  ETag: W/"1f-awz6u8dt4YJbTrpCmlIL..."
Forms           : {}
Headers         : {[Connection, keep-alive], [Keep-Alive, timeout=5], [Content-Length, 31],
                  [Content-Type, application/json; charset=utf-8]...}
Images          : {}
InputFields     : {}
Links           : {}
ParsedHtml      : mshtml.HTMLDocumentClass
RawContentLength : 31
```

Step 5: Server Page Rendering



Step 6: Stop the Server

- In the VS Code terminal running the server, press Ctrl + C to stop it.

```
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> node index.js
Server running at http://localhost:3000/
PS C:\Users\Amita Narayanan\OneDrive\Desktop\Tasks> █
```