Screenshots:

App.js

const express = require('express');

const axios = require('axios');

const { spawn } = require('child\_process');

const app = express();

const PORT = 8008;

app.use(express.urlencoded({ extended: true }));

app.use(express.json());

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

  const urls = req.query.url || [];

  const fetchNumbers = (url) =>

    new Promise((resolve) => {

      const process = spawn('node', ['fetch.js', url]);

      let data = '';

      process.stdout.on('data', (chunk) => {

        data += chunk.toString();

      });

      process.stdout.on('end', () => {

        resolve(JSON.parse(data));

      });

    });

  const fetchPromises = urls.map(fetchNumbers);

  try {

    const results = await Promise.all(fetchPromises);

    const mergedNumbers = new Set();

    results.forEach((result) => {

      if (Array.isArray(result.numbers)) {

        result.numbers.forEach((number) => mergedNumbers.add(number));

      }

    });

    const sortedNumbers = Array.from(mergedNumbers).sort((a, b) => a - b);

    res.json({ numbers: sortedNumbers });

  } catch (error) {

    console.error(error);

    res.status(500).json({ error: 'An error occurred' });

  }

});

app.listen(PORT, () => {

  console.log(`Server is running on port ${PORT}`);

});

Fetch.js

const axios = require('axios');

const url = process.argv[2];

axios

  .get(url, { timeout: 500 })

  .then((response) => {

    const data = response.data;

    process.stdout.write(JSON.stringify(data));

  })

  .catch(() => {

    process.stdout.write(JSON.stringify({ numbers: [] }));

  });

Output Screenshots:



