Practical no 5

Aim : Install node js and create a server using express js and

Theory : Install Node JS

Node.js can be installed in different ways. This post highlights the most common and convenient ones.

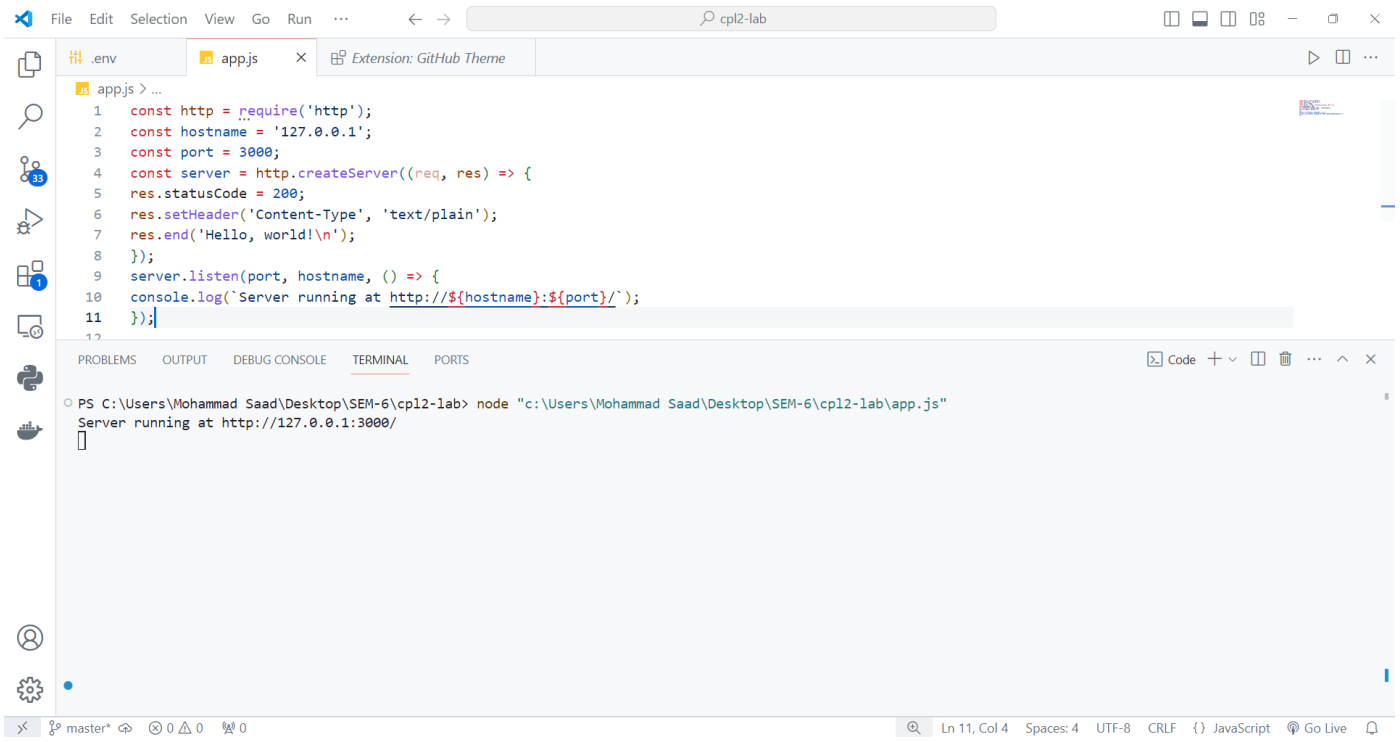
Official packages for all the major platforms are available at https://nodejs.org/download/.

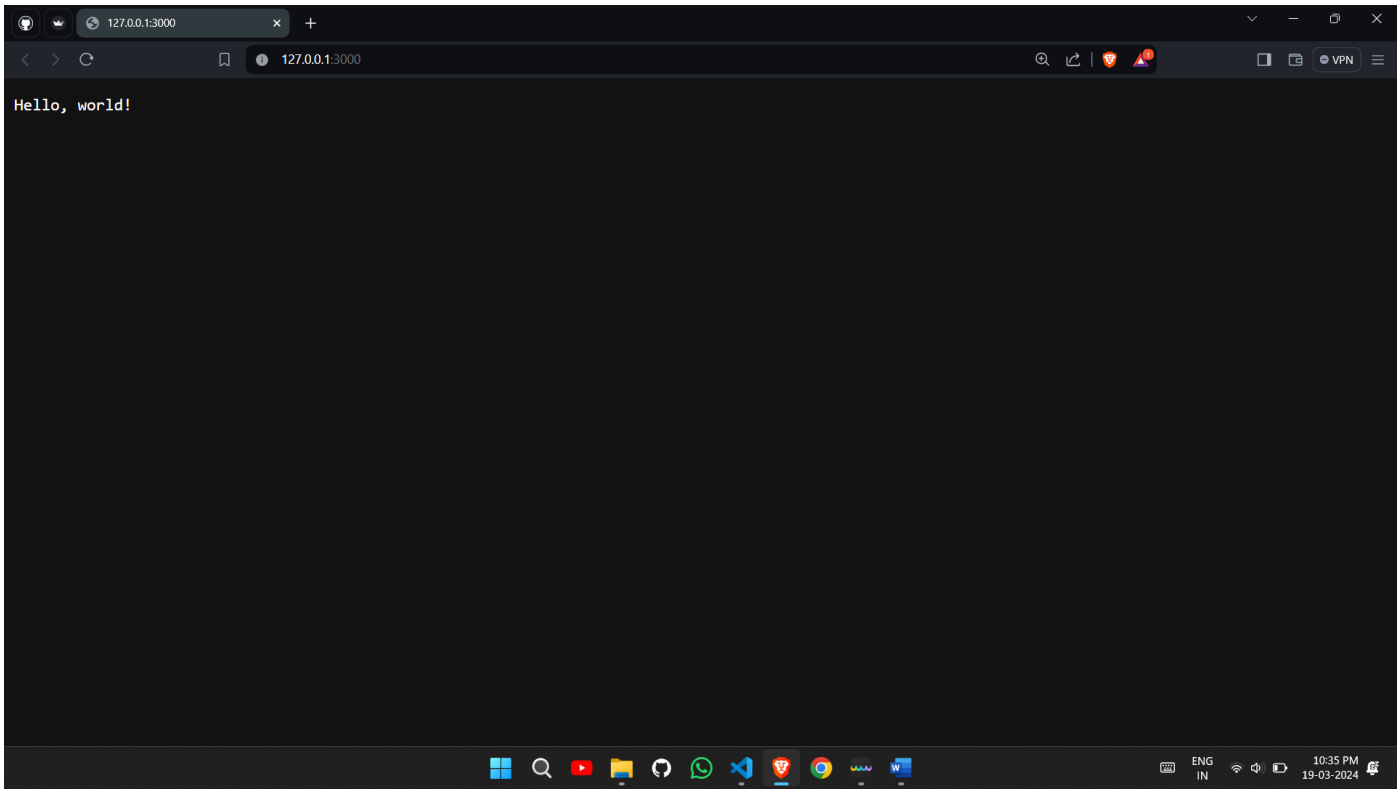
nvm is a popular way to run Node.js. It allows you to easily switch the Node.js version, and install new

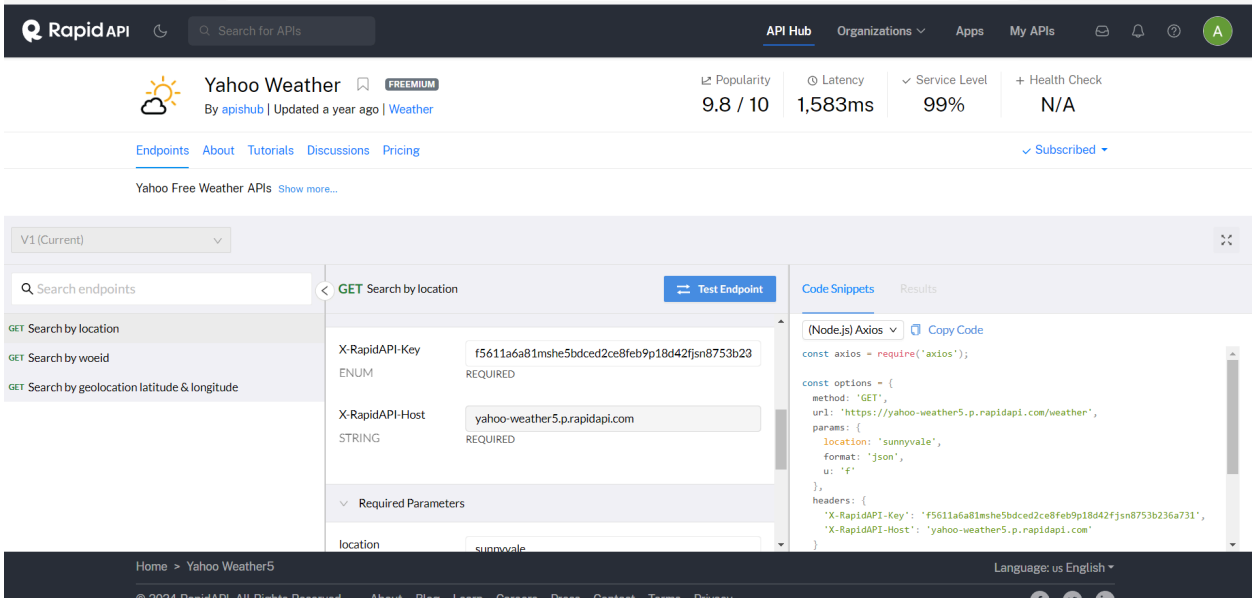
versions to try and easily rollback if something breaks. It is also very useful to test your code with old Node.js versions.

Code:

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

const hostname = '127.0.0.1';

const port = 3000;

const server = http.createServer((req, res) => {

res.statusCode = 200;

res.setHeader('Content-Type', 'text/plain');

res.end('Hello, world!\n');

});

server.listen(port, hostname, () => {

console.log(`Server running at http://${hostname}:${port}/`);

});
```


File Edit Selection View Go Run
cpl2-lab
.env
app.js
Extension: GitHub Theme
app.js > ...
1 const http = require('http');
2 const hostname = '127.0.0.1';
3 const port = 3000;
4 const server = http.createServer((req, res) => {
5 res.statusCode = 200;
6 res.setHeader('Content-Type', 'text/plain');
7 res.end('Hello, world!\n');
8 });
9 server.listen(port, hostname, () => {
10 console.log(`Server running at http://${hostname}:${port}/`);
11 });
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Code
PS C:\Users\Mohammad Saad\Desktop\SEM-6\cpl2-lab> node "c:\Users\Mohammad Saad\Desktop\SEM-6\cpl2-lab\app.js"
Server running at http://127.0.0.1:3000/
master*
Ln 11, Col 4
Spaces: 4
UTF-8
CRLF
JavaScript
Go Live


127.0.0.1:3000
127.0.0.1:3000
VPN
Hello, world!
ENG
IN
10:35 PM
19-03-2024

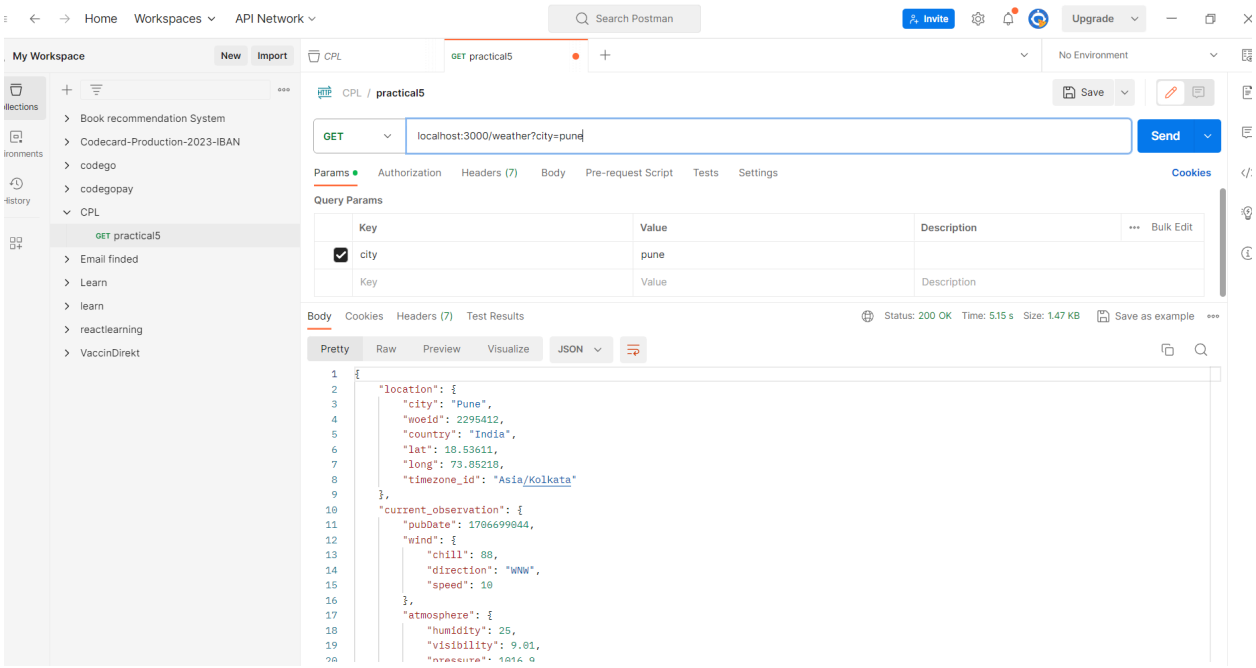


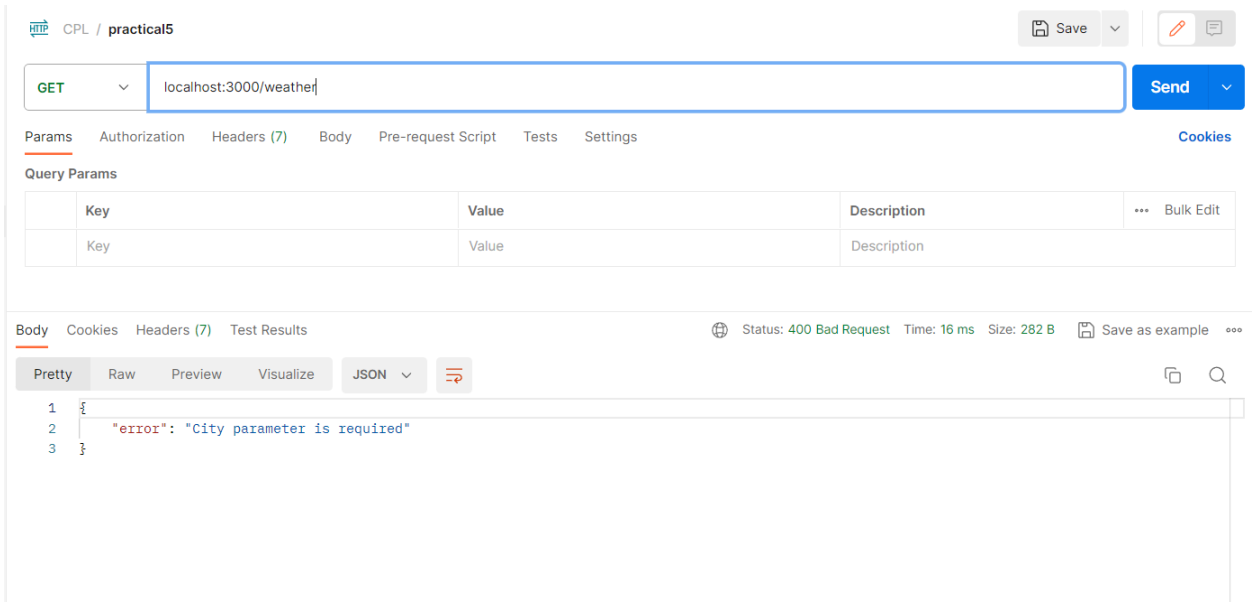
Code :

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

const axios = require('axios');

const app = express();

const port = 3000;

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

  try {

    const city = req.query.city;

    if (!city) {

      return res.status(400).json({ error: 'City parameter is required' });

    }

const options = {

  method: 'GET',

  url: 'https://yahoo-weather5.p.rapidapi.com/weather',

  params: {

    location: city,
```

```
    format: 'json',
    u: 'f'
  },
  headers: {
    'X-RapidAPI-Key': 'f5611a6a81mshe5bdced2ce8feb9p18d42fjsn8753b236a731',
    'X-RapidAPI-Host': 'yahoo-weather5.p.rapidapi.com'
  }
};
const response = await axios.request(options);
            console.log("got the data");
    res.json(response.data);
  } catch (error) {
    console.error('Error fetching weather data:', error);
    res.status(500).json({ error: 'An error occurred while fetching weather data' });
  }
});
app.listen(port, () => {
  console.log(`Server is running on http://localhost:${port}`);
});
```

OUTPUT:





API Call : localhost:3000/weather?city=pune

Response JSON :

{

"location": {

"city": "Pune",

"woeid": 2295412,

```
    "country": "India",
    "lat": 18.53611,
    "long": 73.85218,
    "timezone_id": "Asia/Kolkata"
},
"current_observation": {
    "pubDate": 1706699246,
    "wind": {
        "chill": 88,
        "direction": "WNW",
        "speed": 10
    },
    "atmosphere": {
        "humidity": 25,
        "visibility": 9.01,
        "pressure": 1016.9
    },
    "astronomy": {
        "sunrise": "7:08 AM",
        "sunset": "6:28 PM"
    },
    "condition": {
        "temperature": 90,
        "text": "Sunny",
        "code": 32
    }
```

```
},
"forecasts": [
  {
    "day": "Wed",
    "date": 1706716800,
    "high": 90,
    "low": 57,
    "text": "Sunny",
    "code": 32
  },
  {
    "day": "Thu",
    "date": 1706803200,
    "high": 90,
    "low": 57,
    "text": "Mostly Sunny",
    "code": 34
  },
  {
    "day": "Fri",
    "date": 1706889600,
    "high": 86,
    "low": 56,
    "text": "Mostly Cloudy",
    "code": 28
  },
```

```
{
    "day": "Sat",
    "date": 1706976000,
    "high": 91,
    "low": 59,
    "text": "Mostly Cloudy",
    "code": 28
},
{
    "day": "Sun",
    "date": 1707062400,
    "high": 93,
    "low": 60,
    "text": "Partly Cloudy",
    "code": 30
},
{
    "day": "Mon",
    "date": 1707148800,
    "high": 91,
    "low": 59,
    "text": "Sunny",
    "code": 32
},
{
    "day": "Tue",
```

```
    "date": 1707235200,
    "high": 79,
    "low": 65,
    "text": "Haze",
    "code": 21
},
{
    "day": "Wed",
    "date": 1707321600,
    "high": 81,
    "low": 64,
    "text": "Sunny",
    "code": 32
},
{
    "day": "Thu",
    "date": 1707408000,
    "high": 84,
    "low": 65,
    "text": "Sunny",
    "code": 32
},
{
    "day": "Fri",
    "date": 1707494400,
    "high": 89,
```

```
            "low": 57,
            "text": "Sunny",
            "code": 32
        },
        {
            "day": "Sat",
            "date": 1707580800,
            "high": 85,
            "low": 55,
            "text": "Sunny",
            "code": 32
        }
    ]
}
```


Conclusion :

Hence we have used yahoo Free API for weather data via Rapid API platform and used postman to trigger the API request.