Node.js is an open-source web *server* environment that uses JavaScript and runs on various platforms (Windows, Linux, Unix, Mac OS X, etc.). In Module 10, you were introduced to Node.js and used it to make API calls. In this module, you will use Node.js to construct a web *server* that will produce and consume data from Kafka.

**Installing Node.js**

Open a new command prompt and run the following command to determine whether you already have Node.js installed on your machine:

| node --version |
| --- |

If the command is not recognized, then you will need to install Node.js:

Open your web browser and navigate to [Node.js](https://nodejs.org/en/)

[Links to an external site.](https://nodejs.org/en/)

to download and install the version of Node.js that is appropriate for your operating system (Windows or Mac).

**Creating a Hello World Node.js Web Application**

Now you are ready to create a Hello World Node.js web application by following the steps below:

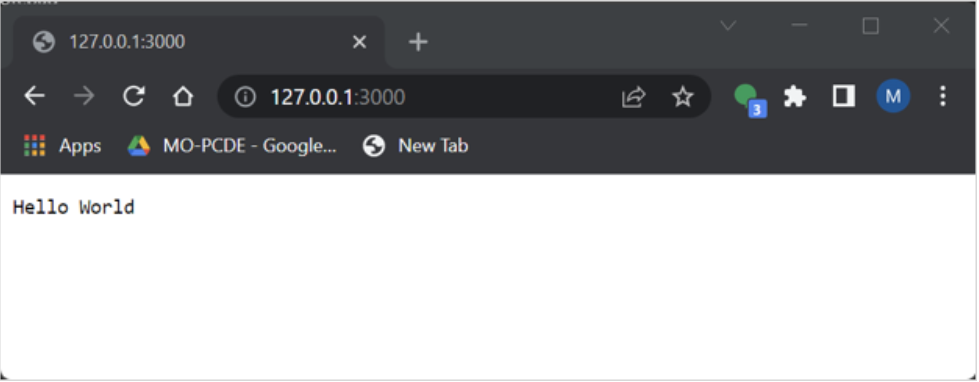
1. To create a Hello World web application, open VS Code and create a file titled index.js. Paste the following code into the file and save it:

| 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}/`)  }) |
| --- |

1. Open a command prompt and navigate to the folder where you saved the index.js file. Run the node index command below to start the web *server*:

| node index |
| --- |

1. Node.js will start the web *server* with the Hello World page shown below:

Next, open your web browser and navigate to http://127.0.0.1:3000:You are now ready to build web applications using the Node.js *server*.

**Project 24.3 Connections**

In [Project 24.3](https://classroom.emeritus.org/courses/10605/assignments/246369), you will use Node.js to run JavaScript code to consume the location data produced by the code that defines the *publisher*.

As a data engineer, you will often use Node.js to execute JavaScript code outside a web browser. This ensures better productivity during the development process because it’s fast and it allows for code sharing and reuse.