Lesson 05 Demo 04

Creating HTTP Requests and Callbacks

Objective: To create an HTTP request and a callback for sending messages to the server and handling HTTP requests effectively

Tools required: Node Package Manager and Visual Studio Code

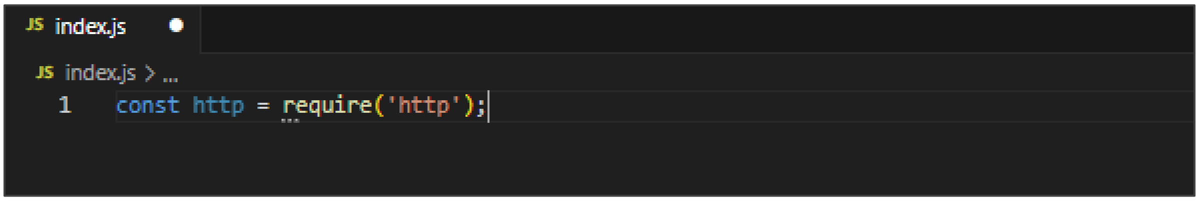
Prerequisites: Basic Linux Commands, NPM commands, JavaScript, and HTTP module

Steps to be followed:

1. Create a simple HTTP request in JavaScript
2. Create a callback for the HTTP request

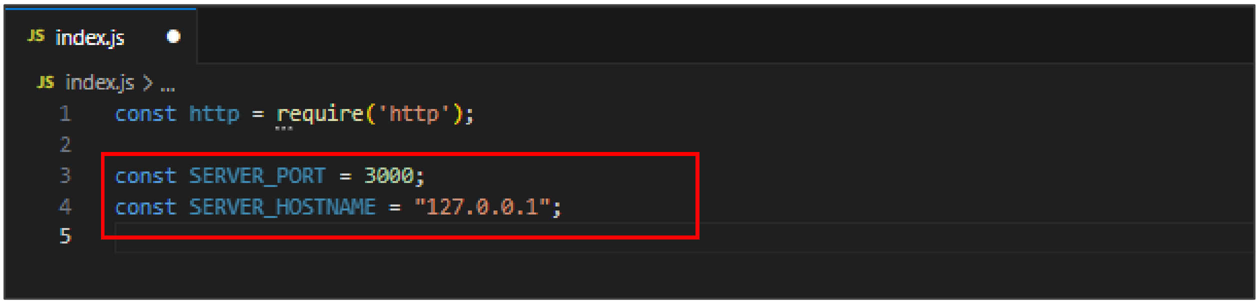
Step 1: Create a simple HTTP request in JavaScript

1.1 Use the following code statement to import the HTTP module:
const http = require('http');



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

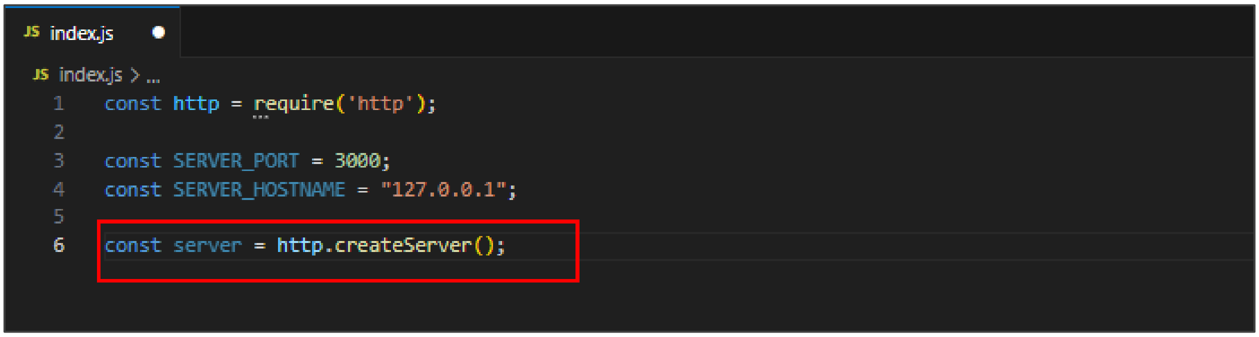
1.2 Declare the port number and hostname for the server setup:
const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";



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

const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";

```

1.3 Use the **createServer** function to create the HTTP server:

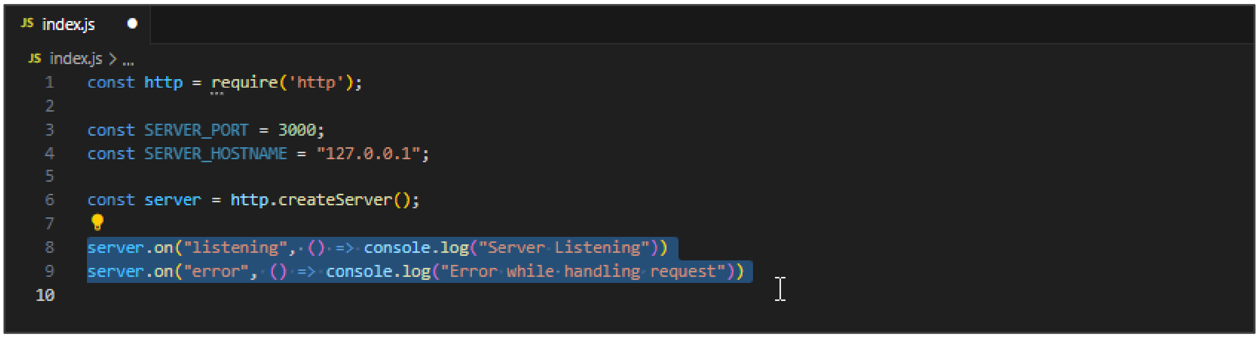
const server = http.createServer();



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

const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";

const server = http.createServer();
```

1.4 Set the server to listen on **listening** and **error** events:

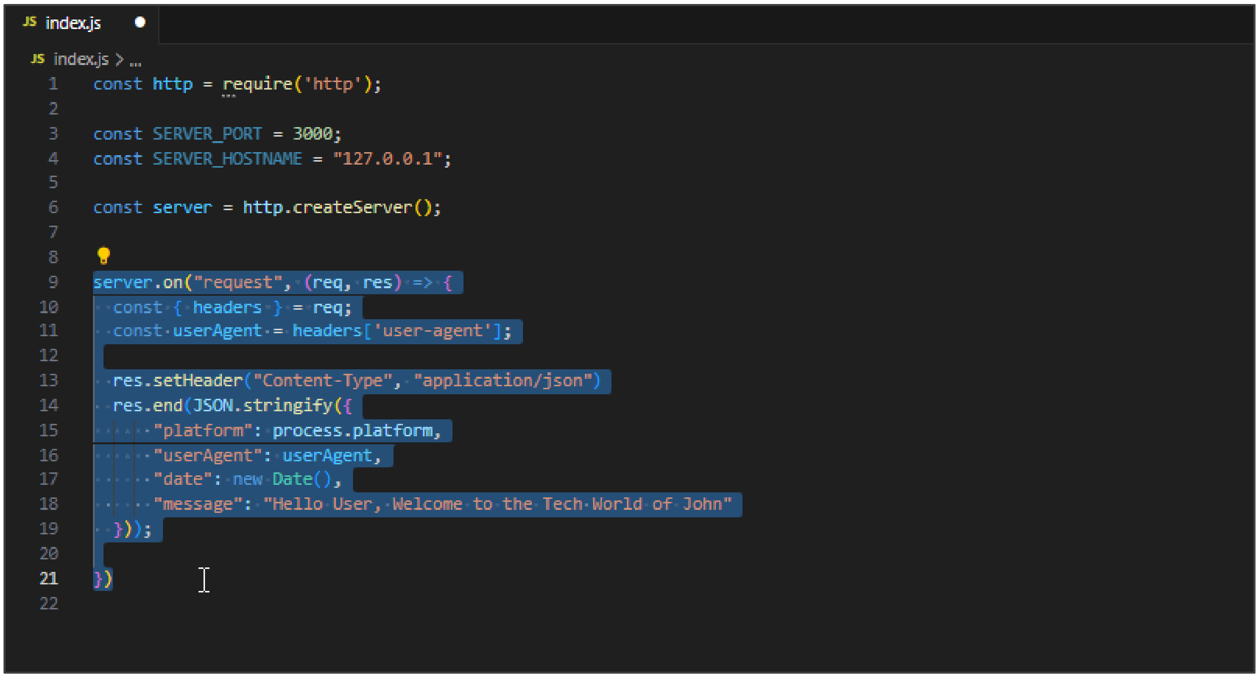
server.on("listening", () => console.log("Server Listening"))
server.on("error", () => console.log("Error while handling request"))



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

const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";

const server = http.createServer();

server.on("listening", () => console.log("Server Listening"))
server.on("error", () => console.log("Error while handling request"))
```

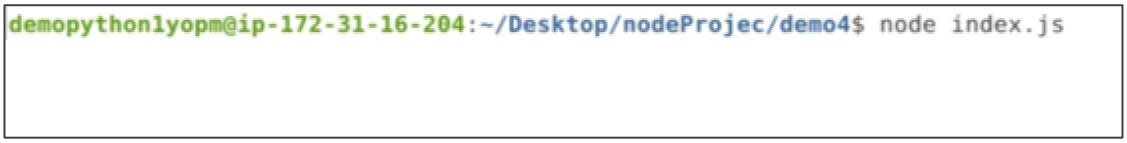
1.5 Send a JSON response on receiving an HTTP request:

```
server.on("request", (req, res) => {
  const { headers } = req;
  const userAgent = headers['user-agent'];

  res.setHeader("Content-Type", "application/json")
  res.end(JSON.stringify({
    "platform": process.platform,
    "userAgent": userAgent,
    "date": new Date(),
    "message": "Hello User, Welcome to the Tech World of John"
  }));

})
```



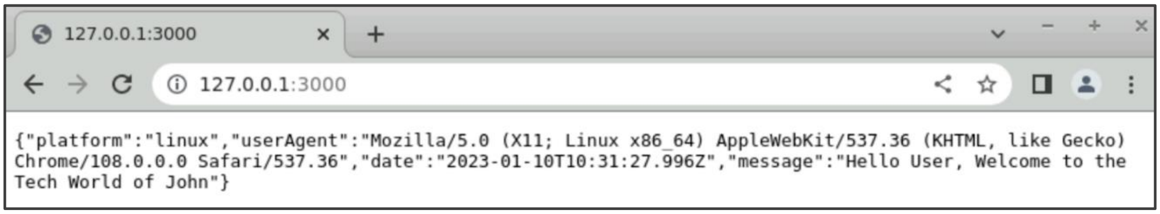
```
const http = require('http');

const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";

const server = http.createServer();


server.on("request", (req, res) => {
  const { headers } = req;
  const userAgent = headers['user-agent'];

  res.setHeader("Content-Type", "application/json")
  res.end(JSON.stringify({
      "platform": process.platform,
      "userAgent": userAgent,
      "date": new Date(),
      "message": "Hello User, Welcome to the Tech World of John"
  }));

})

```

1.6 Execute the command **node index.js** in the terminal to view the output



```
demopython1yopm@ip-172-31-16-204:~/Desktop/nodeProjec/demo4$ node index.js
```

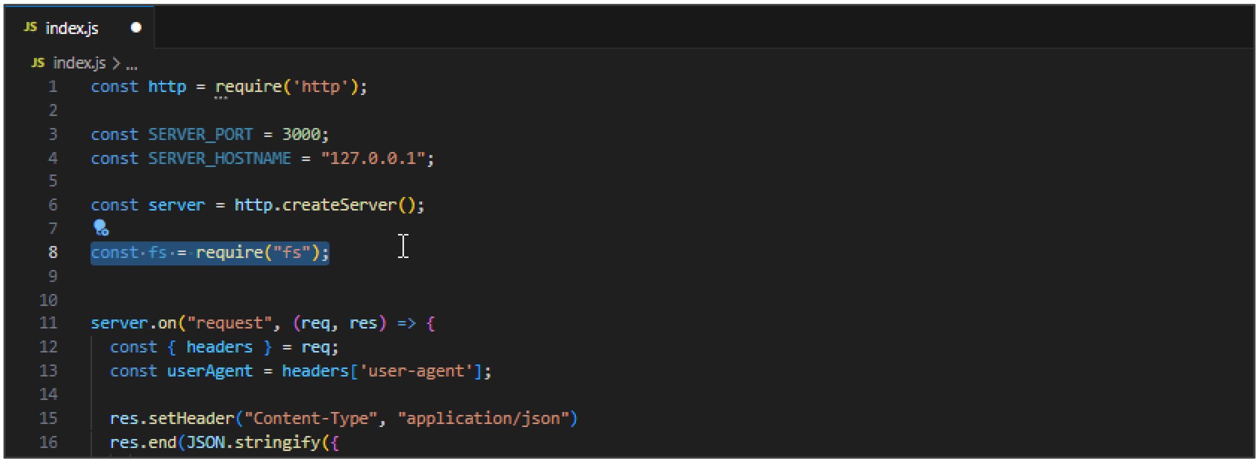
1.7 Navigate to the Chrome browser and type the URL **127.0.0.1:3000**



```
{"platform":"linux","userAgent":"Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/108.0.0.0 Safari/537.36","date":"2023-01-10T10:31:27.996Z","message":"Hello User, Welcome to the
Tech World of John"}
```

Step 2: Create a callback for the HTTP request

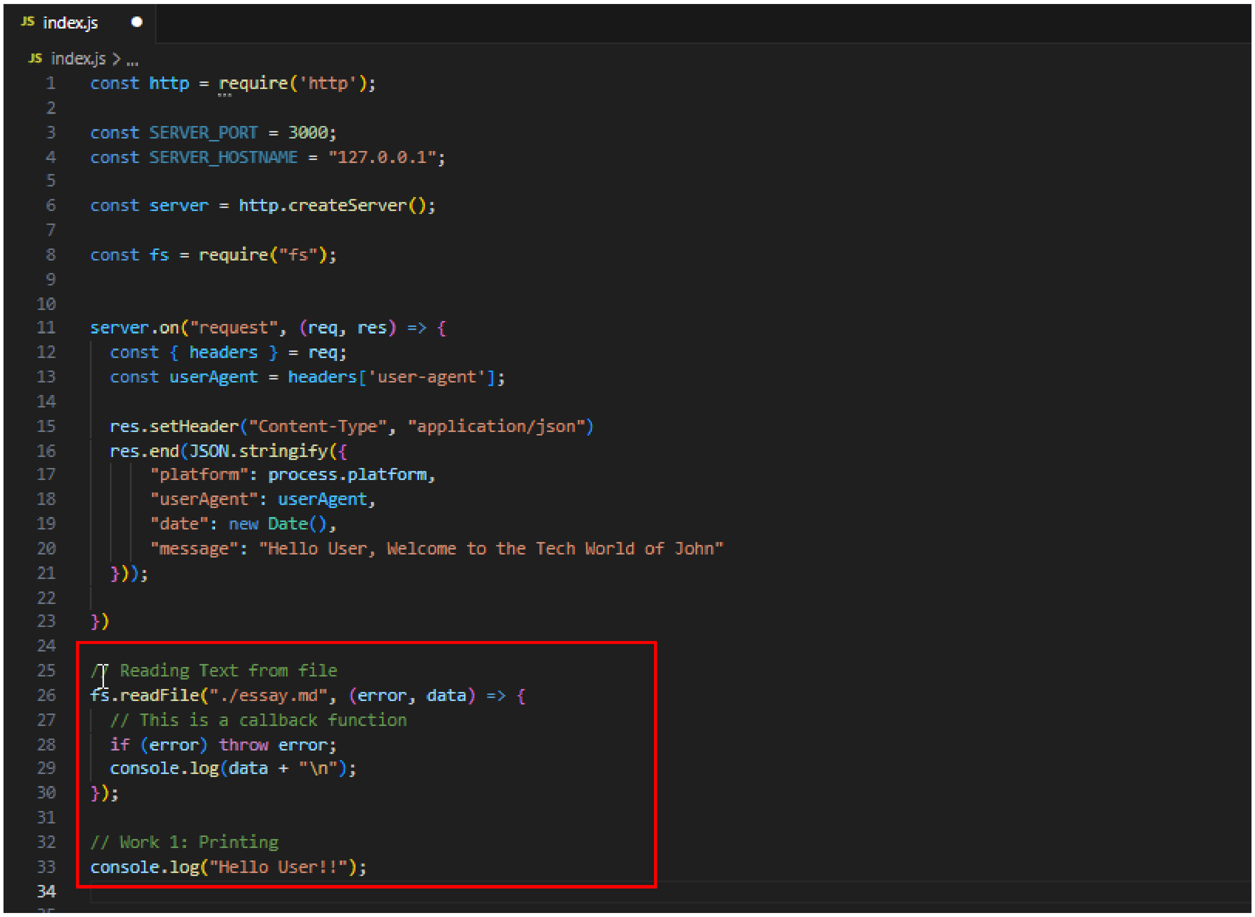
2.1 Import the file system module:
const fs = require("fs");



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

const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";

const server = http.createServer();

const fs = require("fs");


server.on("request", (req, res) => {
  const { headers } = req;
  const userAgent = headers['user-agent'];

  res.setHeader("Content-Type", "application/json")
  res.end(JSON.stringify({
```

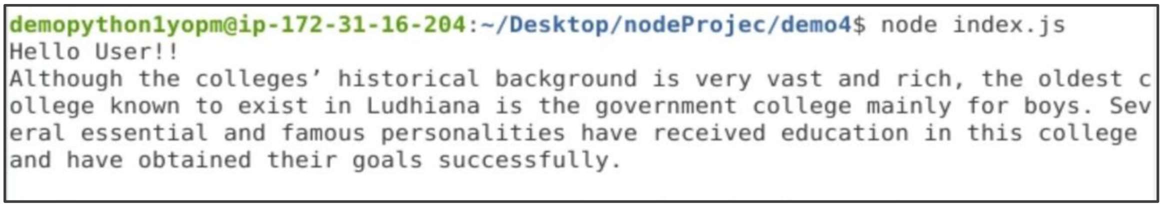
2.2 Use the **readFile** function to read the **essay.md** file, which implements a callback. If an error occurs, it throws the error else, prints the data.

```
// Reading Text from file
fs.readFile("./essay.md", (error, data) => {
 // This is a callback function
 if (error) throw error;
 console.log(data + "\n");
});

// Work 1: Printing
console.log("Hello User!!");
```



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

const SERVER_PORT = 3000;
const SERVER_HOSTNAME = "127.0.0.1";

const server = http.createServer();

const fs = require("fs");


server.on("request", (req, res) => {
  const { headers } = req;
  const userAgent = headers['user-agent'];

  res.setHeader("Content-Type", "application/json")
  res.end(JSON.stringify({
      "platform": process.platform,
      "userAgent": userAgent,
      "date": new Date(),
      "message": "Hello User, Welcome to the Tech World of John"
  }));

})

// Reading Text from file
fs.readFile("./essay.md", (error, data) => {
  // This is a callback function
  if (error) throw error;
  console.log(data + "\n");
});

// Work 1: Printing
console.log("Hello User!!");
```

2.3 Execute the ***node index.js*** command in the terminal to view the output



```
demopython1yopm@ip-172-31-16-204:~/Desktop/nodeProjec/demo4$ node index.js
Hello User!!
Although the colleges’ historical background is very vast and rich, the oldest c
ollege known to exist in Ludhiana is the government college mainly for boys. Sev
eral essential and famous personalities have received education in this college
and have obtained their goals successfully.
```

By following these steps, you have successfully created an HTTP request and a callback for sending messages to the server and handling HTTP requests effectively