In the previous tutorial we went over Initializing a NodeJs Project. In this tutorial we’ll be going over how to start up a nodeJs HTTP Server.

* How to use the ‘http’ module

Node.js has a built in HTTP Module, which can be accessed with this method. This allows NodeJs to transfer data to the (HTTP) In order to include the HTTP module, use the require() method,

|  |
| --- |
| const http = require('http'); |

* How to create a listening server

Use the create server method to create an HTTP server that creates a listener on the specified port or path. When port 8080 gets access, it will write “Hello World!” as a response.

|  |
| --- |
| http.createServer(function (req, res) {    res.write('Hello World!'); //write response to client    res.end(); //ends the response  }).listen(8080); //the server listens on port 8080 |

\When the function passes the http.createServer() method, Then you’ll be able to access the port in server 8080.

* How to handle requests and responses?

In order to handle requests, the listen method needs to be called on the server object. When an HTTP request hits the server, node calls the request handler function with a few handy objects for dealing with the transaction, request and response

* How to send different responses based on the URL user is connecting to?

First, we check if the url if equal to the default home page using “(req.url == “/”)” It redirects the user to the file the code is written in.

|  |
| --- |
| const http = require('http');  http.createServer(function (req, res) {    if (req.url == "/") {      res.write('Hello World!'); //write response to client |

But if you want to redirect the user to different responses, you’ll be able to use the same with req.url.

|  |
| --- |
| } else if (req.url == “/request”) |

req.url represents the HTTP request. The equality operator (==) check whether its two operands are equal. This will send the user a different response when connecting to “/request”