**Node.js**

**What is Node.js?**

Node.js is a runtime environment for building fast and scalable network applications easily with JavaScript. Node.js uses an event driven, non blocking I/O model that makes it lightweight and efficient, perfect for data intensive real time applications that run across distributed devices. It  provides a rich library of various JavaScript modules which simplifies the development of web applications using Node.js to a great extent.

**Features**

Asynchronous and Event driven

The APIs of Node.js is non-blocking. A Node.js server based will never wait for any of the API to return certain data. The server moves to the next API after calling it and a notification mechanism of Events of Node.js helps the server to get a response from the previous API call.

Very Fast

Node.js library is fast in code execution.

Single threaded but highly Scalable

Node.js uses a single threaded program and the same program can provide service to a much larger number of requests than traditional servers like Apache HTTP Server.

No Buffering

The Node.js applications does not buffer any data and it just give the output of the data.

License

Node.js is under the MIT license.

**Environment Setup**

Local Environment Setup

You need the Text Editor and Node.js binary installables softwares for you to setup the environment for Node.js.

Text Editor

This is where you will type your program for the Node.js.  Examples of few editors include Windows Notepad, OS Edit command, Brief, Epsilon, EMACS, and vim or vi.   
The files you create with your editor are called source files and contain program source code. The source files for Node.js programs are typically named with the extension ".js".

Executing a File

After you have installed the Node.js, create a file that is named with the extension ".js". For example, in a file named index.js type the code : console.log("Hello Node.js")   
Now if you run the file using Node.js you will be able to see the output with the code : $ node index.js

Creating your First Application

The Node.js has three main components for you to actually create an application. We use the require directive to load Node.js modules, Apache HTTP Server for creating servers, and read request and return request that will read the HTTP request made by the client which can be a browser or a console and return the response.

Step 1

Import Required Module - we use the require directive   
var http = require("http");

Step 2

Create Server - we use the created http instance from the first step and call the http.createServer() method and bind it with a port like at port 8081 using the listen method.   
http.createServer(function (request, response) {   
// Send the HTTP header       
// HTTP Status: 200 : OK   
  // Content Type: text/plain   
    response.writeHead(200, {'Content‐Type': 'text/plain'});   
        // Send the response body as "Hello NodeJS"   
   response.end('Hello NodeJS\n');  }).listen(8081);   
// Console will print the message   
console.log('Server running at http://localhost:8081/');

Step 3

Testing Request and Response - put the import required module and created server in a file named index.js then run the code in your Node.js command prompt : node main.js

Now you will be able to create an application were you will be able to see the result in the web browser with the your localhost and the port you have provided in the code of your created server.