



Http Core Module

#Node JS Notes

Web Based Hello World



Node.js Module:

- Module in Node.js is a simple or complex functionality organized in single or multiple JavaScript files which can be reused throughout the Node.js application.
- Each module in Node.js has its own context, so it cannot interfere with other modules or pollute global scope. Also, each module can be placed in a separate .js file under a separate folder.



Node.js Module Types

- Node.js includes three types of modules:
 1. Core Modules
 2. Local Modules
 3. Third Party Modules



Node.js Core Modules

- Node.js is a light weight framework.
- The core modules include bare minimum functionalities of Node.js.
- These core modules are compiled into its binary distribution and load automatically when Node.js process starts.
- However, you need to import the core module first in order to use it in your application.



Important core modules in Node.js.

Core Module	Description
http	http module includes classes, methods and events to create Node.js http server.
url	url module includes methods for URL resolution and parsing.
querystring	querystring module includes methods to deal with query string.
path	path module includes methods to deal with file paths.
fs	fs module includes classes, methods, and events to work with file I/O.
util	util module includes utility functions useful for programmers.

<https://nodejs.org/dist/latest-v11.x/docs/api/>



Loading Core Modules:

- In order to use Node.js core or NPM modules, you first need to import it using `require()` function as shown below.
- `var module = require('module_name');`
- `#include<conio.h>`



Example: Load and Use Core http Module

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

```
var server = http.createServer(function(req, res){  
    //write code here  
});
```

```
server.listen(5000);
```



Create Server

- First We need to required a Http Module
- Node.js has a built-in module called HTTP, which allows Node.js to transfer data over the Hyper Text Transfer Protocol (HTTP).
 - `var http = require("http");`
- Create Server using
 - `http.createServer()`




Server Demo

JS akash.js

```
1 var http = require("http");
2
3 http.createServer(function (request, response) {
4
5     // Send the HTTP header
6     // HTTP Status: 200 : OK
7     // Content Type: text/plain
8     response.writeHead(200, {'Content-Type': 'text/plain'});
9     // Send the response body as "Hello World"
10    response.end('Hello World in Node js\n');
11 }).listen(8081);
12
13 // Console will print the message
14 console.log('Server running at http://127.0.0.1:8081/');
15
```

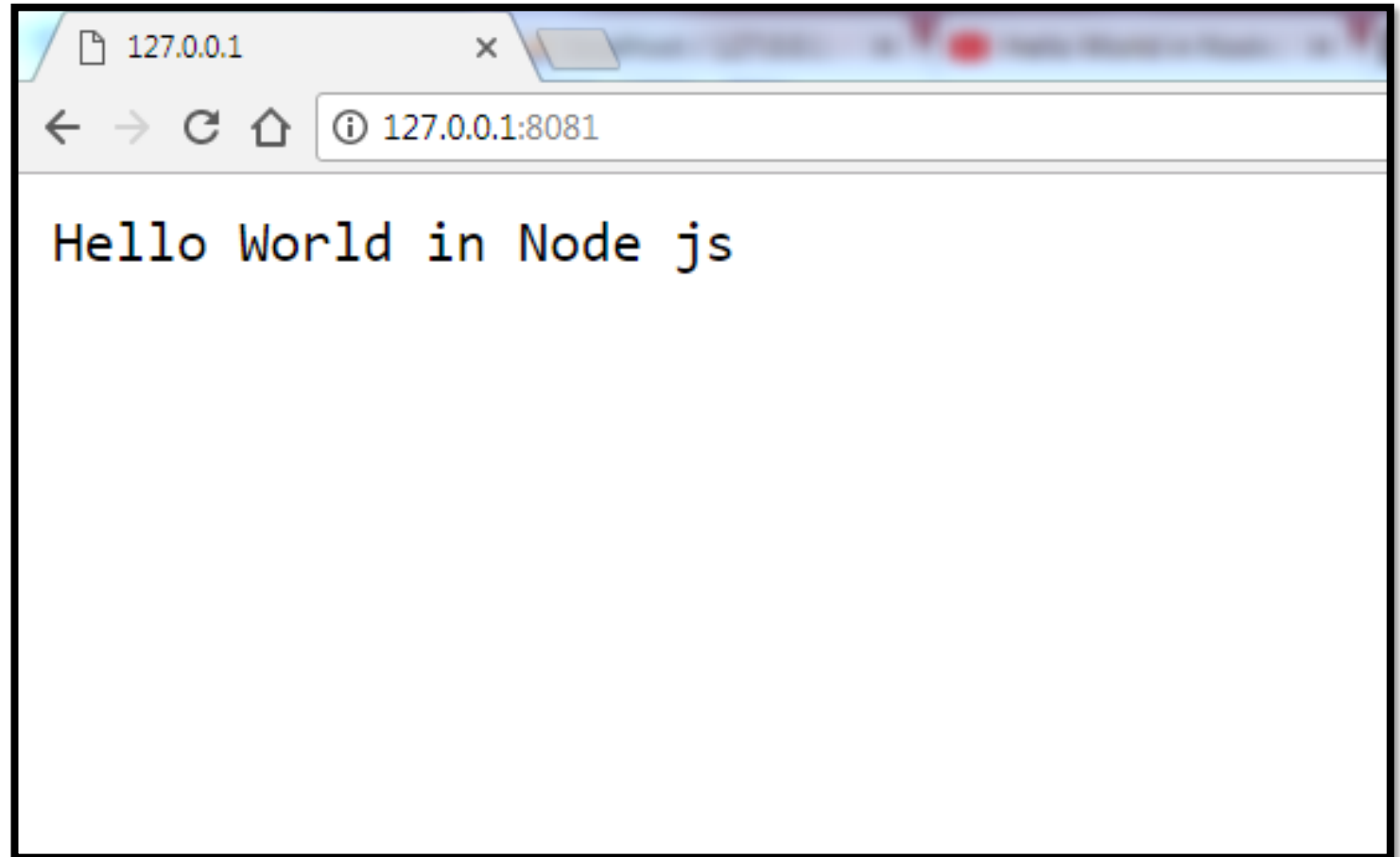


- 
- writeHead writes the HTTP header (status code 200)
 - end writes the body and closes the response.



Output

- Run the Code :
- Node filename.js
- Open the URL
- <http://127.0.0.1:8081>
- <http://localhost:8081/>



Port Number

- Best Port Number
- 8081 (React Native)
- 4200 (Angular)
- 3000 (Express / ReactJS)
- 5000 (Free)
- 4000



Get Exclusive Video Tutorials



www.apptutorials.com

<https://www.youtube.com/user/Akashtips>



Connect With Me



Akash Padhiyar
#AkashSir

www.akashsir.com
www.akashtechlabs.com
www.akashpadhiyar.com
www.apptutorials.com

Social Info



Akash.padhiyar



Akashpadhiyar



Akash_padhiyar



+91 99786-21654



#Akashpadhiyar
#apptutorials



Get More Details

www.akashsir.com



If You Liked It !

Rating Us Now



Just Dial

https://www.justdial.com/Ahmedabad/Akash-Technolabs-Navrangpura-Bus-Stop-Navrangpura/079PXX79-XX79-170615221520-S5C4_BZDET



Sulekha

<https://www.sulekha.com/akash-technolabs-navrangpura-ahmedabad-contact-address/ahmedabad>

