**Aim: Study HTTP module and working of Node.js as a web server**

**Objective: To understand the working of node.js as a web server with built in module HTTP**

**Scope:** To be the most widely used web framework by developers globally

**Theory:-**

The HTTP module provides a way of making Node.js transfer data over HTTP (Hyper Text Transfer Protocol).

**3.1 Syntax**

The syntax for including the HTTP module in your application:

var http = require('http');

## 3.2 HTTP Properties and Methods

|  |  |
| --- | --- |
| createClient() | Deprecated. Creates a HTTP client |
| createServer() | Creates an HTTP server |
| get() | Sets the method to GET, and returns an object containing the user's request |
| globalAgent | Returns the HTTP Agent |
| request() | Returns an object containing the user's request |

The HTTP module can create an HTTP server that listens to server ports and gives a response back to the client.

Use the createServer() method to create an HTTP server:

var http = require('http');  
  
//create a server object:  
http.createServer(function (req, res) {  
  res.write('Hello World!'); //write a response to the client  
  res.end(); //end the response  
}).listen(8080); //the server object listens on port 8080

The function passed into the http.createServer() method, will be executed when someone tries to access the computer on port 8080.

Save the code above in a file called "demo\_http.js", and initiate the file:

Initiate demo\_http.js:

Initiate demo\_http.js:

C:\Users\*Your Name*>node demo\_http.js

Result: displayed on [http://localhost:8080](http://localhost:8080/)

Conclusion: Node.js worked **as a web server** successfully.