

NODE.JS - WEB MODULE

http://www.tutorialspoint.com/nodejs/nodejs_web_module.htm

Copyright © tutorialspoint.com

What is Web Server?

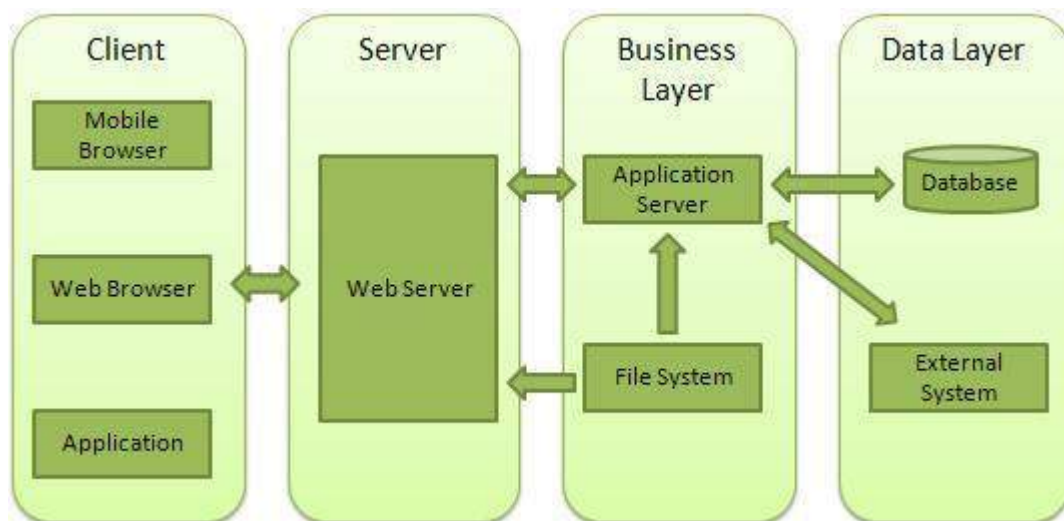
Web Server is a software application which handles HTTP requests sent by the HTTP client, like web browsers, and returns web pages in response to the clients. Web servers usually delivers html documents along with images, style sheets and scripts.

Most of the web server support server side scripts using scripting language or redirect to application server which perform the specific task of getting data from database, perform complex logic etc. and then sends a result to the HTTP client through the Web server.

Apache web server is one of the most commonly used web server. It is an open source project.

Web Application Architecture

A Web application is usually divided into four layers:



- **Client** - This layer consists of web browsers, mobile browsers or applications which can make HTTP request to the web server.
- **Server** - This layer consists of Web server which can intercepts the request made by clients and pass them the response.
- **Business** - This layer consists of application server which is utilized by web server to do required processing. This layer interacts with data layer via data base or some external programs.
- **Data** - This layer consists of databases or any source of data.

Creating Web Server using Node

Node.js provides **http** module which can be used to create either HTTP client of server. Following is a bare minimum structure of HTTP server which listens at 8081 port.

Create a js file named server.js:

File: server.js

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

// Create a server
http.createServer( function (request, response) {
```

```
// Parse the request containing file name
var pathname = url.parse(request.url).pathname;

// Print the name of the file for which request is made.
console.log("Request for " + pathname + " received.");

// Read the requested file content from file system
fs.readFile(pathname.substr(1), function (err, data) {
  if (err) {
    console.log(err);
    // HTTP Status: 404 : NOT FOUND
    // Content Type: text/plain
    response.writeHead(404, {'Content-Type': 'text/html'});
  } else {
    // Page found
    // HTTP Status: 200 : OK
    // Content Type: text/plain
    response.writeHead(200, {'Content-Type': 'text/html'});

    // Write the content of the file to response body
    response.write(data.toString());
  }
  // Send the response body
  response.end();
});
}).listen(8081);

// Console will print the message
console.log('Server running at http://127.0.0.1:8081/');
```

Next let's create following html file named index.htm in the same directory where you created server.js

File: index.htm

```
<html>
<head>
<title>Sample Page</title>
</head>
<body>
Hello World!
</body>
</html>
```

Now let us run the server.js to see the result:

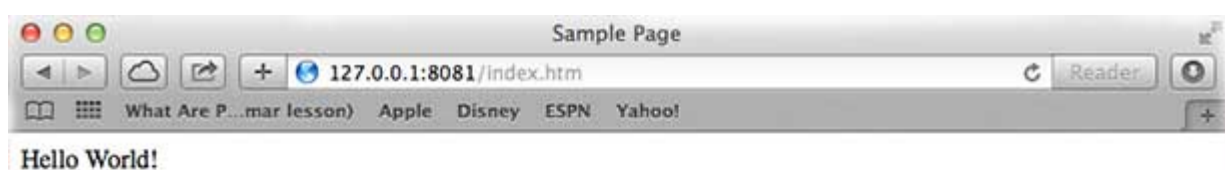
```
$ node server.js
```

Verify the Output

```
Server running at http://127.0.0.1:8081/
```

Make a request to Node.js server

Open <http://127.0.0.1:8081/index.htm> in any browser and see the below result.



Verify the Output at server end.

```
Server running at http://127.0.0.1:8081/  
Request for /index.htm received.
```

Creating Web client using Node

A web client can be created using **http** module. Let's check the following example.

Create a js file named client.js:

File: client.js

```
var http = require('http');  
  
// Options to be used by request  
var options = {  
  host: 'localhost',  
  port: '8081',  
  path: '/index.htm'  
};  
  
// Callback function is used to deal with response  
var callback = function(response){  
  // Continuously update stream with data  
  var body = '';  
  response.on('data', function(data) {  
    body += data;  
  });  
  
  response.on('end', function() {  
    // Data received completely.  
    console.log(body);  
  });  
}  
// Make a request to the server  
var req = http.request(options, callback);  
req.end();
```

Now run the client.js from a different command terminal other than server.js to see the result:

```
$ node client.js
```

Verify the Output.

```
<html>  
<head>  
<title>Sample Page</title>  
</head>  
<body>  
Hello World!  
</body>  
</html>
```

Verify the Output at server end.

```
Server running at http://127.0.0.1:8081/  
Request for /index.htm received.
```