* Install NPM(Node package manager) which installs Node.js.
  + NPM provide package structure for node js files Ex: how in servlet web application, where to place static files, dynamic files, java files etc…
* Install node plugin(nodeclipse) in eclipse
* Node.js is a single-threaded application, but it can support concurrency via the concept of **event** and **callbacks**.
* As soon as Node starts its server, it simply initiates its variables, declares functions and then simply waits for the event to occur.
* Although events look quite similar to callbacks, the difference lies in the fact that callback functions are called when an asynchronous function returns its result, whereas event handling works on the observer pattern.

**How Node Applications Work?**

In Node Application, any async function accepts a callback as the last parameter and a callback function accepts an error as the first parameter.

var fs = require("fs");

/\*\*

 \* Here fs.readFile() is a async function whose purpose is to read a file. If an

 \* error occurs during the read operation, then the err object will contain the

 \* corresponding error, else data will contain the contents of the file.

 \* readFile passes err and data to the callback function after the read

 \* operation is complete, which finally prints the content.

 \*/

fs.readFile('views/input.txt', function(err, data) {

if (err)

return console.error(err);

console.log(data.toString());

});

console.log("Program Ended");

**How Node.js server and client works?**

1. We can create static web server from server node with below…
   1. http.createServer( function (request, response) { }
2. We can create http client with below. Ex: calling a rest service
   1. var req = http.request(options, callback);