- Modules can be single file or collection of multiple files.

- Types:

(I) core modules - Node.js has many built in modules.

- These modules can be loaded into the program by using require function.

- Some of core modules are HTTP, assert, fs, path, process, url.

(II) local modules :-

- It created locally in Node.js appli

(III) Third party modules :-

- It are available online using the package node package manager.
- These modules installed in the project folder or globally.

- Some of popular third-party modules are mongoose, express, angular and react.

• exports keyword make properties & method available outside the module file

DATE / /
PAGE

require :

• require() is a built-in function to include external modules that exist in separate files.

• require() basically reads a javascript file, executes it and then proceeds to return the export object.

Module patterns :

• It is design pattern which is used to group a set of variables and functions together in a single scope.

The Node Event Emitter :

• Every action ~~on~~ a computer is a event.

•

• Node-JS has built-in module, called "Events", where we create, -fire and listen for your own events.

• all event properties and method are an instance of an EventEmitter object.

• you can assign event handlers to your own events with the Event-Emitter object.

• To fire an event, use emit() metho

Reading & writing files (FS):

- when require module typically sets var name to equal to module name.

synchronous version.

- fs.readFileSync() this function reads the read file.

- fs.writeFileSync() writes.

- easiest way to write to files in Node.js is to use the fs.writeFile.

- The fs module gives us both asynchronous and synchronous options to handle files:-

- Synchronous option blocks the execution of the code until the file operation is completed

- Asynchronous option does not block the execution of code.

- asynchronous version fs.readFile() It takes two arg filename, a callback function.

- fs.writeFile() asynchronous version. It takes three arg - the file name, the data to write, and a callback function.

Creating & Removing Directories :-

- To delete file from directory use fs.unlink('writeme.txt');

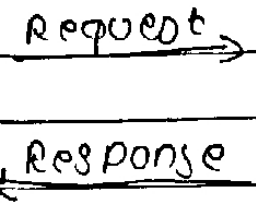
- There are two way to create directories i.e synchronous, asynchronous

- fs.mkdirsync("name"); to create. // synchronous version
- fs.rmdirsync('name'); to remove

- For asynchronous we use async.
- fs.mkdir("name", function(){});

when we used asynchronous version pass callback function.

=> clients & servers:



- Browser send an request to server and server handle the request and send the response to client. (each web)
- Protocols: A set of communication rules, that two side agree to use when communicating.
- Each computer of server identified by its own unique IP address
- If we want to connect client to server first we need to connect to IP address- then it open a socket b/w two computer which is channel that which information can be sent.
- We need different protocol to send