- // Function statements //-

```
Function sayhi() {
  Console.log("HII");
}
sayhi()
```

---// Function Expression //--

```
var sayHi = function () {
  console.log("hi")
}
sayHi().
```

* var fs = require('fs')

var = fs.readFileSync('filename', 'utf-8').

eg:- fs.readFileSync('readme', 'utf-8').

- use to read file content Synchronsly

fs.writeFileSync('filename',);

eg:- fs.writeFileS

- use to write file.

Page No.
Date

client ⟶ Request + Req Header ⟶ Server
⟵ Response + Resp Header ⟵

* Protocol - Protocol is used to communicate between client & server, It is set of communication which both sides agree

* Socket - Socket is a channel it is used to transfer data between them.

* Response Header :-
"Context-Type" ⇒ type of content is given in this
status ⇒ It was a error or Succesful.

* Buffer is a temperory storage spot for chunk of data that is transfered from one place to another.

It transfers small chuncks of data at a time

Page No.
Date

* Stream :- Stream is a flow of data. In node.js data will flow like chunks & those chunks will store in buffer & finally that buffer will share the data.

Creating stream in node.js to transfer data will increase the performance.

There are 3 types :-

- writable stream :- allow to write a stream
- Readable stream :- allow node.js to read data from stream
- Duplex :- Can read & write to stream.

* Pipes :- The data is converted in small chunks

Page No.
Date

* pipe - is used to Connecte multiple stream together. We can read & write stream together for transfer of data from one to another file.

* Routing : It is way to handle clients request

* Express :- Express in node.js is a web application framework that provides broad features for building web & mobile application. Used for building Single page, multipage web application

Command - npm install express

* nodemon :- It is a tool that helps develop Node.js based application by automatically restarting the node application when we make changes