

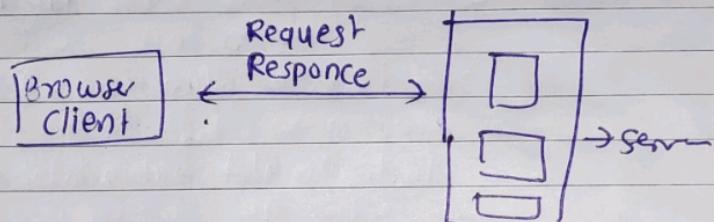
#Express JS

node \Rightarrow node is great source to do anything which is related to hardware with javascript only

but express is built for specifically for web app development

\Rightarrow But for backend express is very specific for the express js

\Rightarrow we can able to create a server directly with express js



Browser & server talk in language of response request

so with listen
see code

$\$ \ const \ Server = require('express');$

$Server = Server.listen() \rightarrow$ this will start
server

$\$ \ Server(115) \text{ in (Port 192)}$



So upto it we create a socket at port 3000

```
const one = require('express')
one app server One. express() create
exp() a ser
port = 3000;
```

server.listen(port)

So If you run this snippet only you will get
only the server listening but not get any
response for that we have to use

```
serve.get("location", function to be called)
          ↑
          after get request
the get request is request which
serve receive & it give response
```

localhost:3000 → local
It is not here

) function(request, response) {

console.log(request); -

// It will print request you get from browser

& response.send("msg you want to send");

}

and that's it

⇒ You can able to give multiple response
to multiple location

get & listen are request which help
to do conversation b/w host & server

⇒ You can send wrapped html under get & send
as our respon

app.send(" <h1> Hello world </h1> ")
response Inside double quotes

⇒ use nodeman so that you don't have close &
restart server again as it take lot of time
to run Close server.js file

app.send(" / ")

route different routes different values
for get

⇒ Response with & html file

to send entire file of html

response.sendFile("location of file")

-- dirname in send file will give location of
(current file folder)

-- dirname it is like pwd in terminal

so we can map our file location
properly

--dirname is a Javascript keyword for
pwd

so send like

response.sendFile(`--dirname + '/index.html')`

① Cannot post index.html

action tab in a form will send received data to that specific location through data object & variable name will be name you specified in input

⇒ so study how to deal with post requests

```
app.post('/1', function(req, res){})
```

res.send('thank you')

status code 200 = OK

so we have to deal with outputs

so we will use body parse npm package

& now we have add parser packet in js file:

as

```
const f = require("body-parser")
```

then we can parse request & get info out of it

-app.use('body')

Code

```
const bp = require('body-parser')
app.use(bp.urlencoded({ extended : true }))

app.use(bp)
```

& now we can use body parser

& now this will activate your route &

So if we use

req.body you will get a JS object
having value written from the html & posted
to you via post

& After that we are able to parse 'get access'
to data

so we can tap value from it via .dot
operator

req.body
is parsed data part