



The Institution of Engineers, NITK Chapter

INTRO TO EXPRESS.JS & BUILD A BASIC CHAT APP

Mishal Shah

Convenor, IE NITK 2019-20

GSoC '19 Student, GSoC ' 20 Mentor





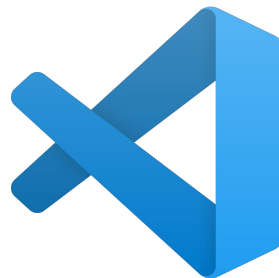
Agenda

- Understand about express.js
- Create simple REST APIs with express.js (Hands-on)
- About socket.io library
- Build a minimal chat app (Hands-on)



Tools

- Node
- NPM
- Visual Studio Code
- Postman



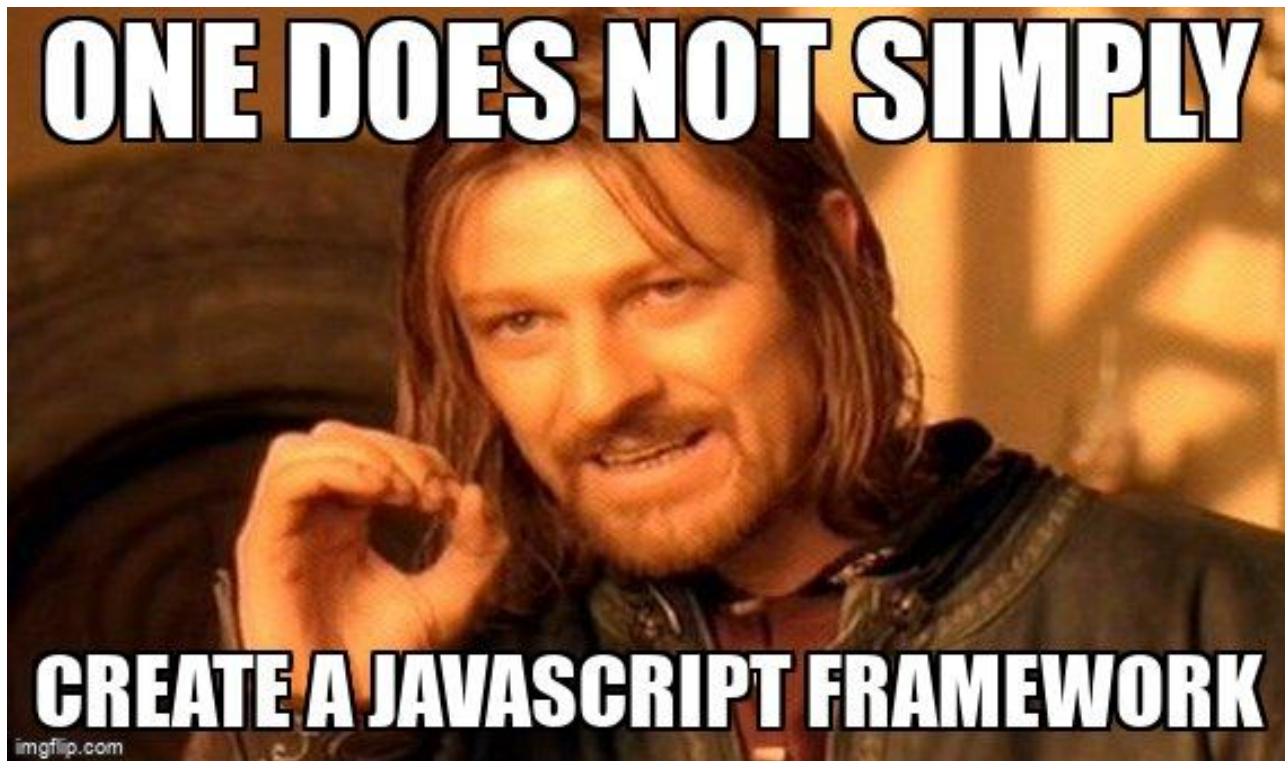
What is Express?

Fast, unopinionated, minimalist web framework for node.

Note: It is a backend or server-side framework!

DO I REALLY HAVE TO PICK

A JAVASCRIPT FRAMEWORK?



Why use Express?

- Makes building web applications with Node.js easier & faster
- Very light & extremely fast
- Provides robust routing
- Dynamically render HTML pages (templating engines -- 14+)

Basic Syntax

```
1  var express = require('express');
2  const app = express();
3  const PORT = 3000;
4
5  app.get('/', function(req, res){
6    |   res.send('Hello World');
7  });
8
9  app.listen(PORT, function(req, res) {
10 |   console.log(`Server listening at port ${PORT}`);
11 | });|
```

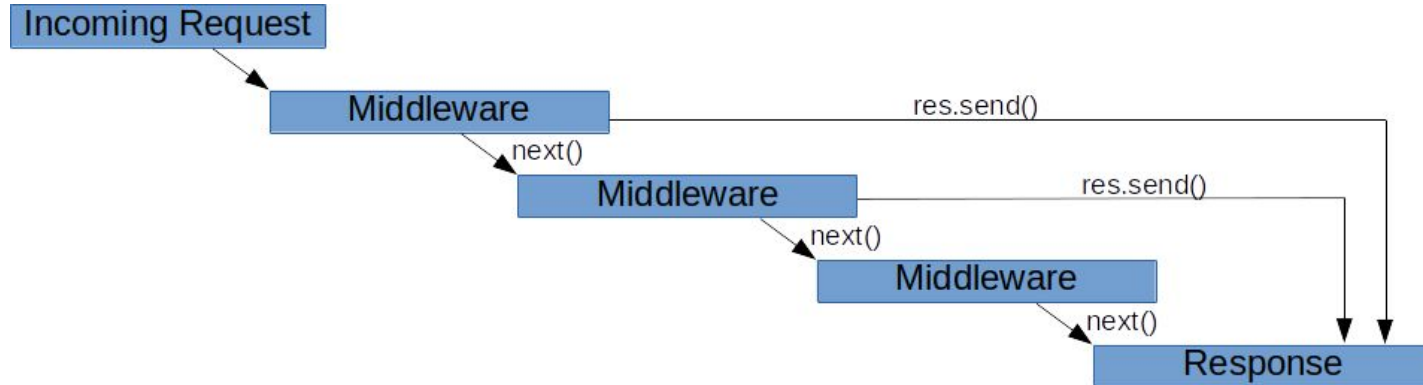
```
1  const http = require('http');
2  const url = require('url');
3  const PORT = 3000;
4
5  const server = http.createServer(function (req, res) {
6    |   const endpoint = url.parse(req.url).pathname;
7
8    |   if(req.method == 'GET') {
9    |     |   if(endpoint == '/') {
10    |     |     |   res.write('Hello World');
11    |     |     |   res.end();
12    |     |     }
13    |     }
14  });
15
16  server.listen(PORT, function(req, res){
17    |   console.log(`Server listening at port ${PORT}`);
18  });|
```

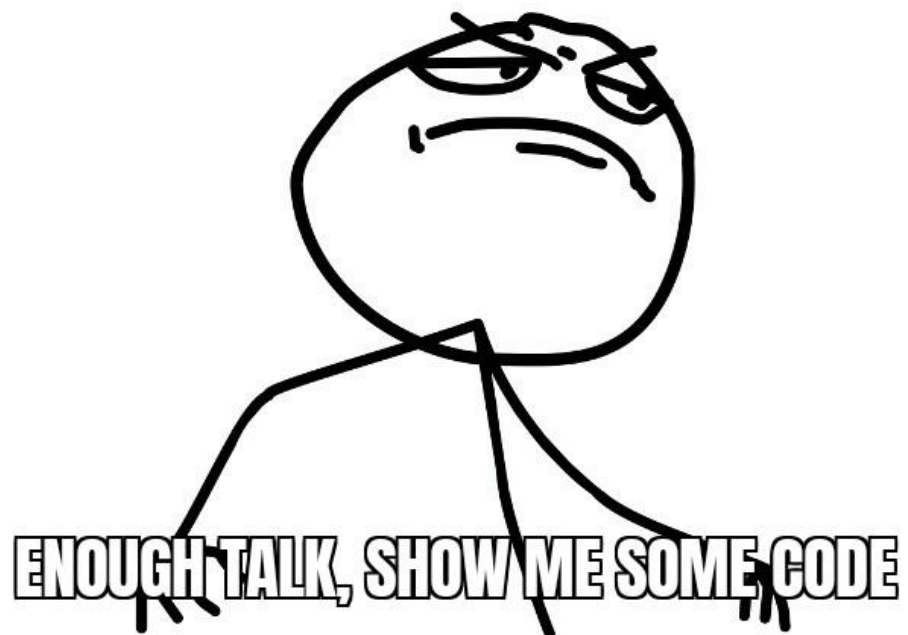
Express vs Node simple code!

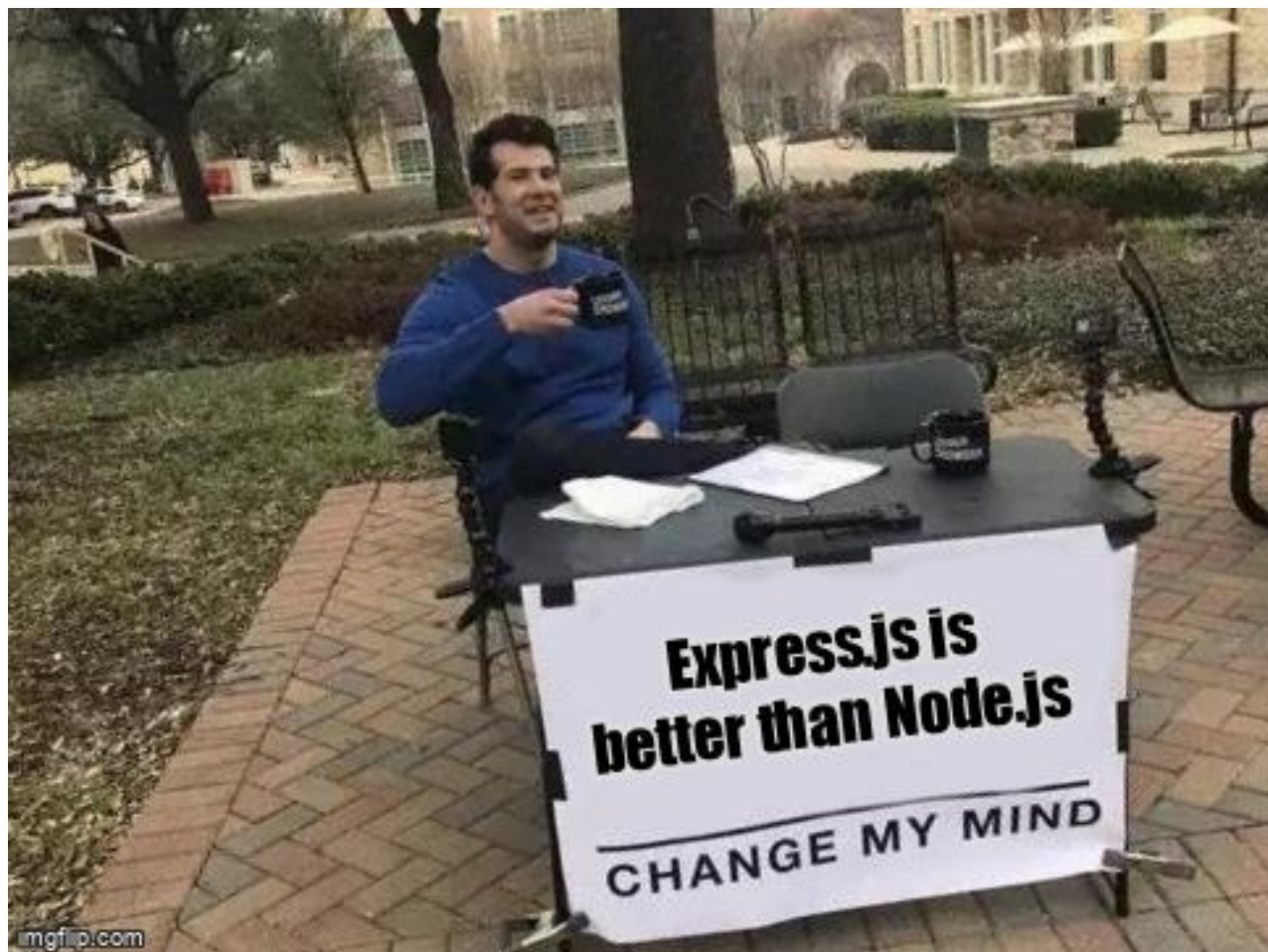
Middleware

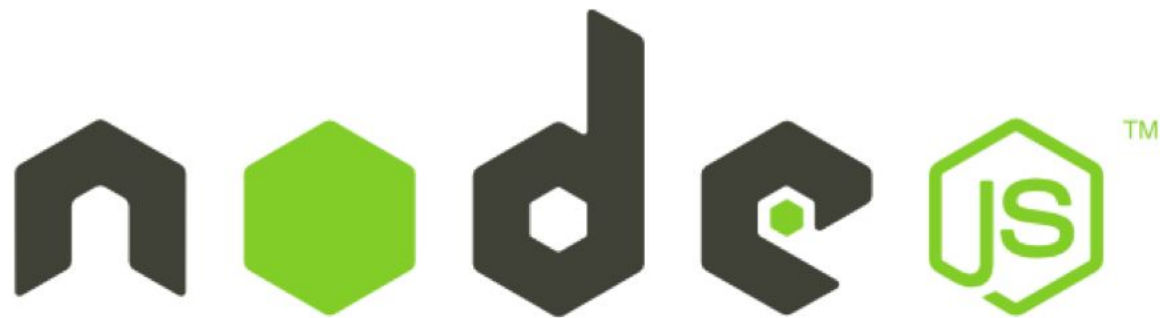
Middleware functions are those pieces of code that have access to the user's request, application response and next middleware to be used.

- Gives developers access to add, remove/modify various features in the application









 **socket.io**

Real-time Communication

- Polling
- Web Sockets

Polling

- Consumes server threads & bandwidth 😞




WebSockets

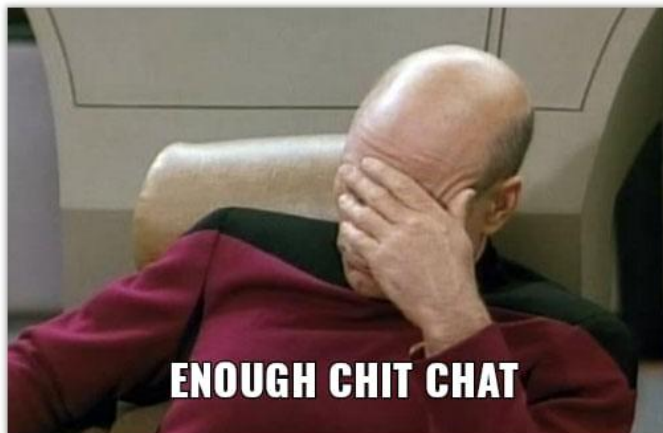
- Provides bi-directional, full-duplex communications over one single TCP connection
- Communications done over TCP port 80 or 443

Socket.IO

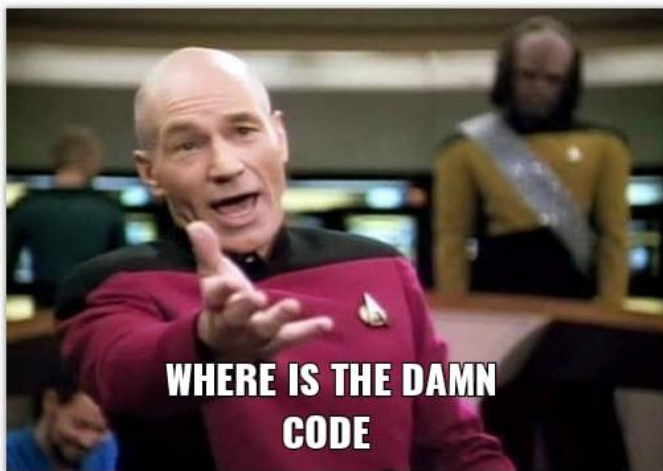
- Enables real-time bi-directional event-based communication
- Works on every platform, browser and device focusing equally on reliability and speed.

Used for:

- Real-time analytics 
- Streaming 
- Instant messaging & chat 
- Documents Collaboration



ENOUGH CHIT CHAT



**WHERE IS THE DAMN
CODE**



Assignment

- User typing functionality
- Number of users in the room functionality
- User left functionality

Many more...

Source code

- All source code will be made available on IE-NITK GitHub Page

<https://github.com/IE-NITK/>

- Will be updated in the video's description

Thank You!