Hi. I’m Tuba.

Front-end vs. Server-Side?

What is a web-server?

What is a web-server?

We usually think of a web server as a huge computer somewhere, sending pages back as we type an address on the browser. But it can also simply refer to the software serving the pages.

We can have more than one server running in the same machine (on different ports)

```
$ python -m SimpleHTTPServer 8000
$ python -m SimpleHTTPServer 3000
$ python -m SimpleHTTPServer 1234
```

Http Protocol?

Http Protocol?

History
Though implemented by Tim-Berners Lee at CERN, the term *hypertext* was coined by Ted Nelson in his Project Xanadu in 1960.

Specs
- The HTTP takes a REQUEST and sends back a RESPONSE
- The REQUEST usually includes information such as:
    - HTTP methods: GET, POST (+ PUT and DELETE)
    - url
    - User-agent

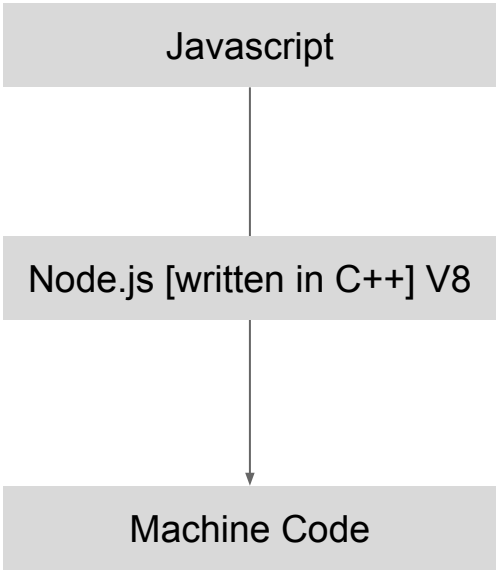
So what is node.js?

So what is node.js?

Basically, it's a server-side Javascript!

Runs on Google's super powerful V8 JS Engine that compiles the Javascript into low-level code/machine code.

(If you don't know what that means, just understand that it's **SUPER FAST**.)



What is procedural programming?

What about node.js?

It’s the complete opposite.

It’s ‘**asynchronous/non-blocking**’.

Instead of procedural, it’s event-driven.

Wait, it’s blocking/non-blocking
What Exactly?!

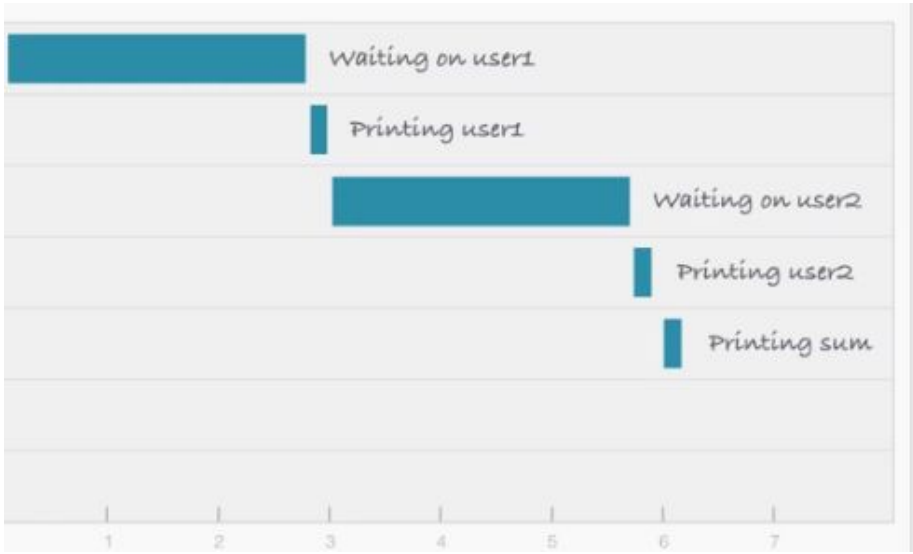
In node.js?

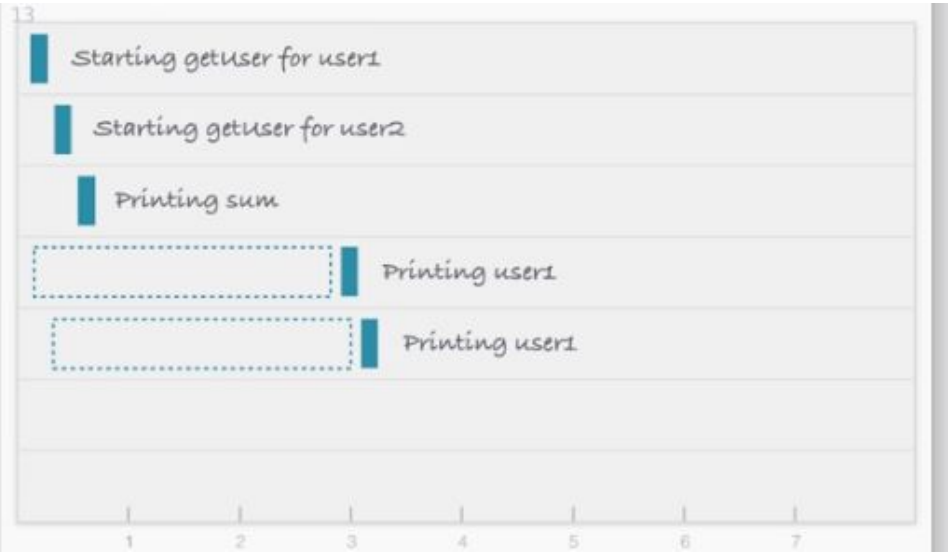
- Line 2 cannot depend on line 1, because line 1 might be taking a while.
- And all the program wants to do is get to the end of the file!

The program would be idle most of the time, waiting for things to happen, before
receiving an event and continuing what it needs to do.

Consider a scenario where we request a backend database for the details of user1 and user2 and then print them on the screen/console.

The response to this request takes time, but both of the user data requests can be carried out independently and at the same time.





In the blocking method, user2's data request is not initiated until user1's data is printed to the screen.

On the other hand, using a non-blocking request, you can initiate a data request for user2 without waiting for the response to the request for user1. You can initiate both requests in parallel.

Interactive vs. Dynamic Website?

What can we do with NodeJS

- Build simple to complex web servers and API servers

- Communicate with databases - CRUD files

- Communicate with device's low-level hardware - sound card, GPU, network configurations, etc
  . Node.js + PComp = IOT!

- Implement websocket protocols
  . Build an IRC chat bots
  . Make collaborative drawing tool

- Scrape websites (for data visualization)

What Node is NOT:

- A programming language. (We write the same exact JS!)
- A web framework.

How it works

Node.js requires modules, as Processing’s import or C++ (oF)’s include.

- Node.js

```
var http = require(‘http’);
```

- Processing

```
import http.requests.*;
```

- C++ (oF)

```
#include "ofxHttpUtils.h"
```

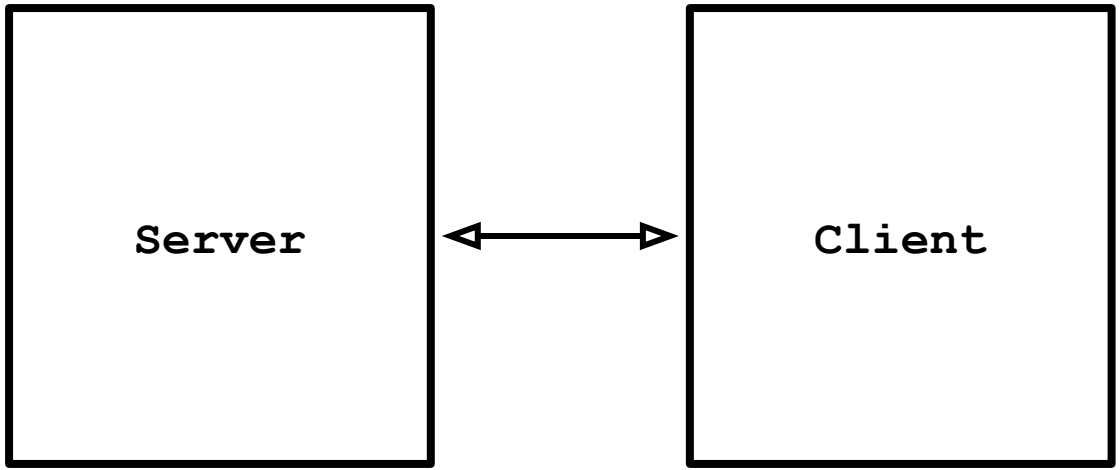
Useful info:

- **Npm (NODE PACKAGE MANAGER)** is the package manager for Javascript. It makes it easy for JavaScript developers to **share and reuse code**.

- **Express** is a framework for building web applications on top of Node.js. It **simplifies the server creation process** that is already available in Node.

Download Node

https://nodejs.org/en/



app.js

index.html
style.css
main.js

Node.js 101 --- Starting a New Project

- Create a new folder for your Node project
- Open terminal and:
  - **cd link/to/your/project/folder**
  - **npm init**
  // Initializes a project, creating a package.json file that will store our project information and list its dependencies
  - **npm install some-node-module another-module --save**
  // You might need "sudo" before those npm, depending on your permissions
  // --save is specifying that the modules should be registered as dependencies in you
  // After this command, you should be able to see a node_modules folder
  Example: npm install express --save

Demo 1.1

Express / App

Express is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

It facilitates the rapid development of Node based Web applications. Following are some of the core features of Express framework -

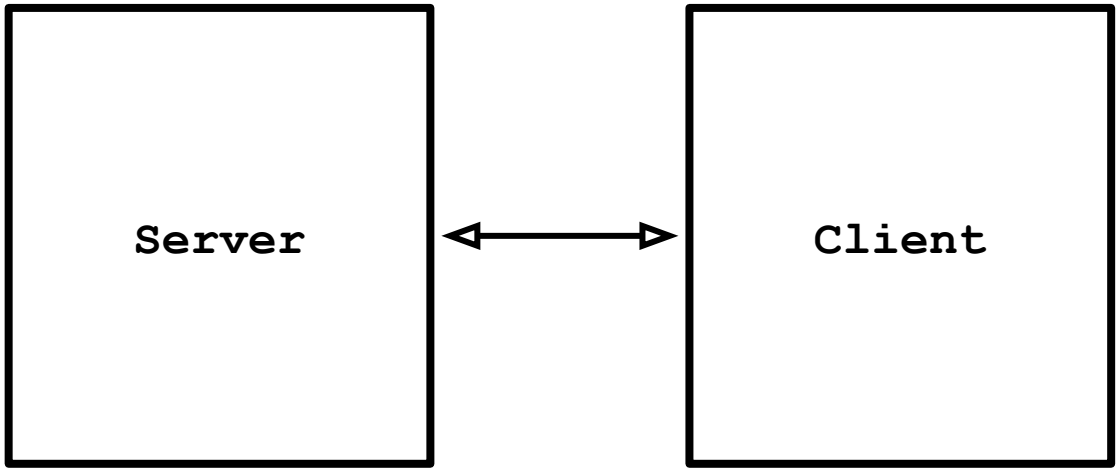
- Allows to set up middlewares to respond to HTTP Requests.

Express / App

https://www.tutorialspoint.com/nodejs/nodejs_express_framework.htm

Demo 1.2

Serving Static Files



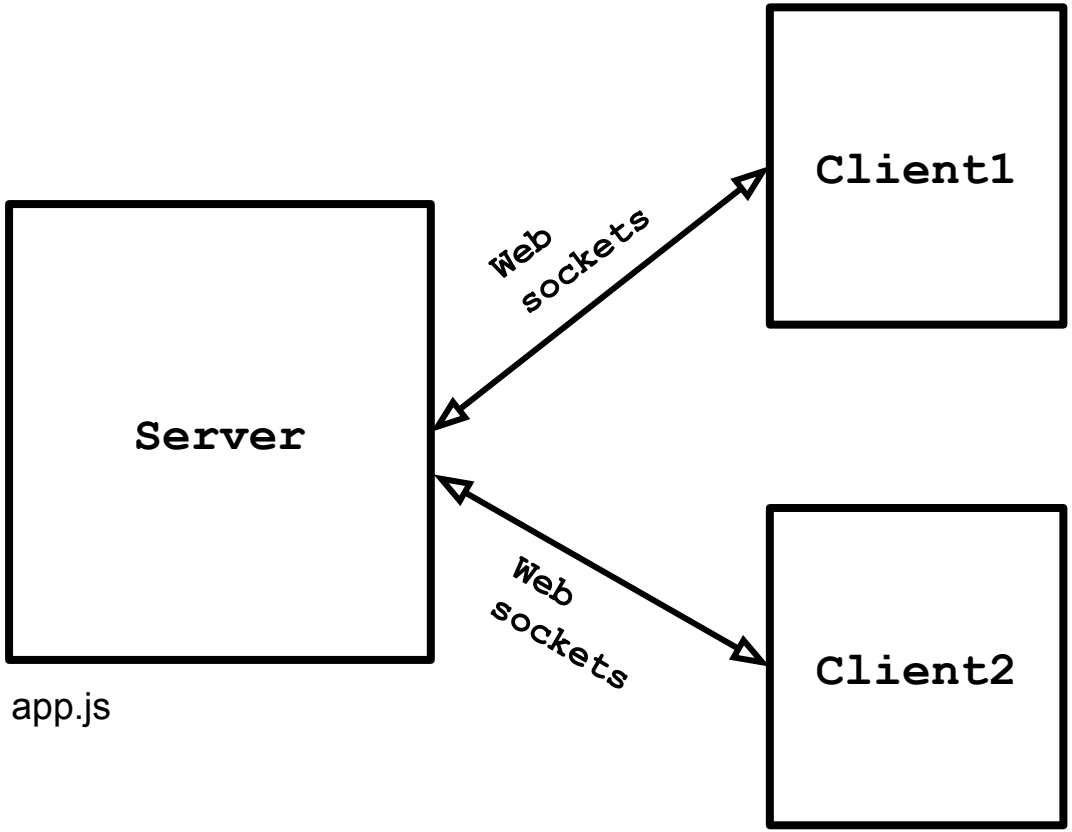
app.js

Index.html
Style.css
Main.js

Demo 1.3

Socket.io

Socket.io is one of the most prized libraries by those who develop with Node.js. Why? Because it allows synchronized communication to take place simply within your app, which means real-time communication!


Client1
Web sockets
Server
Web sockets
Client2
app.js

So what can you do with Sockets?

Let me say it another way: socket.io would allow you to set up a chat service on your website.

Or

To create a multiplayer web-game!

So what can you do with Sockets?

Twitter Bot using News Headlines from API -
https://twitter.com/TwoHeadlines

Socket.io Collaborative Drawing Tool -
http://socketio-intro.herokuapp.com/whiteboard/

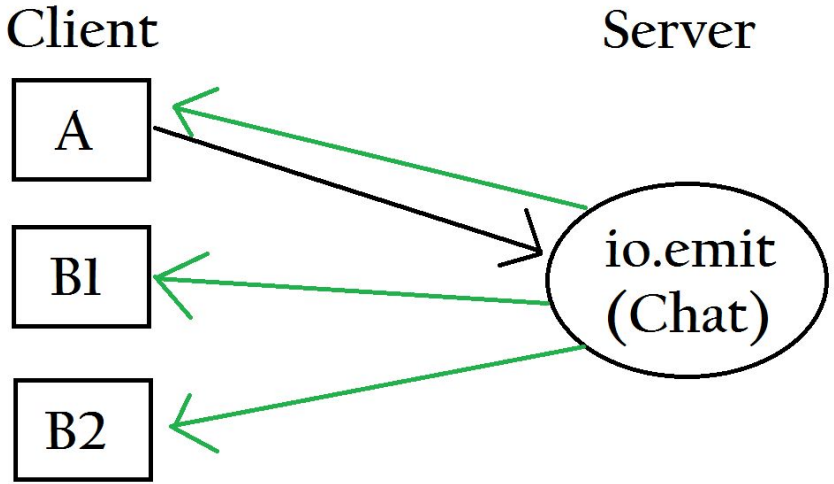
Browser-based multiplayer game -
http://agar.io

Data-Visualization with data collected from APIs -
https://pudding.cool/2017/02/vocabulary/

Data Visualization with real-time data -
https://earth.nullschool.net/

Download Socket package

https://socket.io/


Client
Server
A
B1
B2
io.emit
(Chat)

Demo 2.1

Demo 2.2

How to publish dynamic websites?

Glitch, Heroku ...

Questions?

https://github.com/ozkantuba/NodeClass_Fall2019

Html ?

Http ?

Libraries ?

P5.js ?

Node js ?

Node modules ?

Socket io / WebSockets ?