

# — Node.js

*Ch 1: introduction and installation*

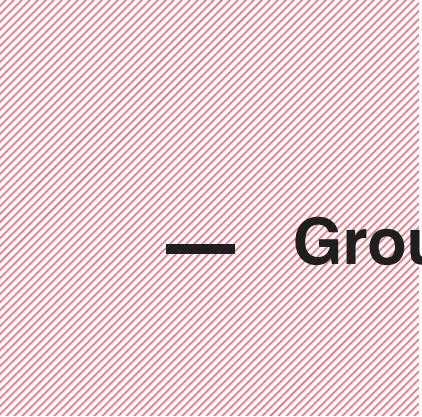
D. HOSTENS & M. DIMA



## — Lecturers

Dirk Hostens ([dirk.hostens@vives.be](mailto:dirk.hostens@vives.be))

Milan Dima ([milan.dima@vives.be](mailto:milan.dima@vives.be))



## — Ground Rules

- Grading: 100% Project + oral defense examen  
See Toledo for details (Practical organization)
- Questions:
  - (practical + course material questions) via Forum on Toledo!
  - Private matters => mail /Teams
- Best preparation => presence + exercises

## — Git (recap)

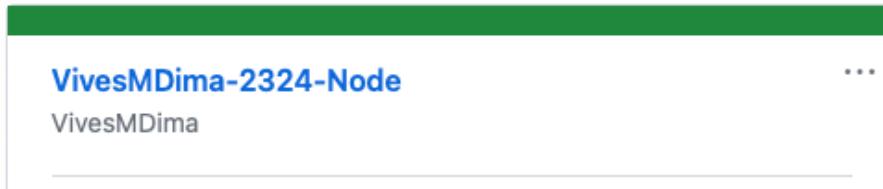
```
git init  
git remote add origin [URL]  
git add .  
git commit -m 'eerste commit'  
git push origin main (master?)
```



## Github classroom



Join the classroom:



node®

## Download Node.js®

Get Node.js® v22.12.0 (LTS) for macOS using fnm with npm

Info Installation methods that involve community software are supported by the teams maintaining that software.

```
1 # Download and install fnm:  
2 curl -o https://fnm.vercel.app/install | bash  
3  
4 # Download and install Node.js:  
5 fnm install 22  
6  
7 # Verify the Node.js version:  
8 node -v # Should print "v22.12.0".  
9  
10 # Verify npm version:  
11 npm -v # Should print "10.9.0".
```

Bash

Copy to clipboard

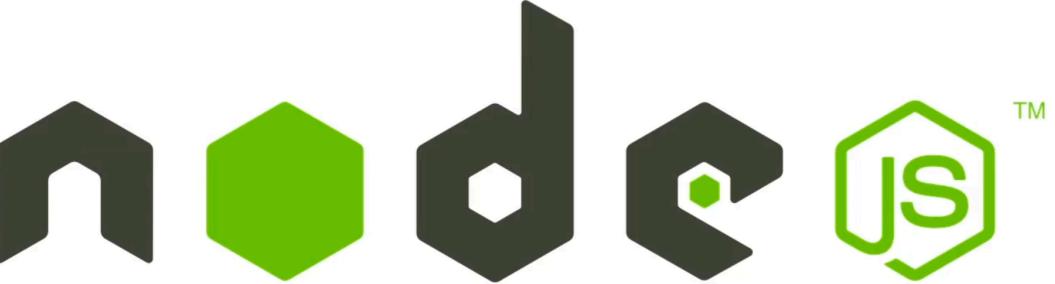
"fnm" is a cross-platform Node.js version manager. If you encounter any issues please visit [fnm's website](#)

Or get a prebuilt Node.js® for macOS running a x64 architecture.

[macOS Installer \(.pkg\)](#) [Standalone Binary \(.gz\)](#)

Read the [changelog](#) for this version.  
Read the [blog post](#) for this version.  
Learn how to [verify](#) signed SHASUMS.  
Learn how to [build Node.js](#) from source.  
Check out our [nightly](#) binaries or all [previous releases](#) or the [unofficial](#) binaries for other platforms.

## — What is Node?

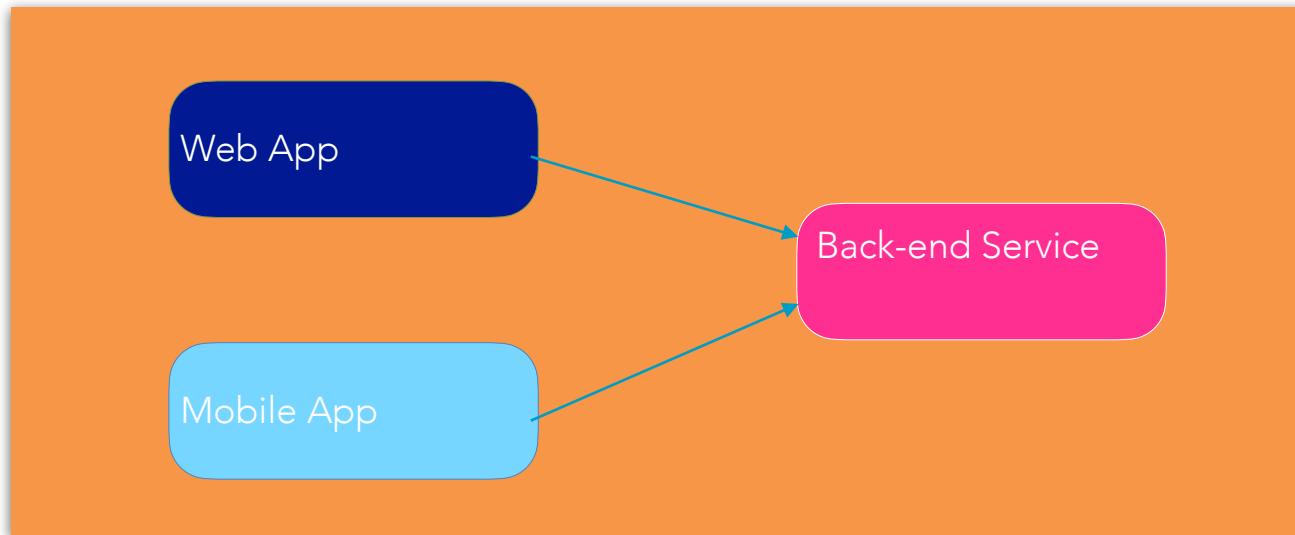


- ⬢ Open Source
- ⬢ Cross-platform
- ⬢ Runtime environment
- ⬢ To run javascript (or TS) outside the browser

A **runtime environment** for  
executing JavaScript code

## — What is it used for?

- ⌘ Usually for Back-end Services
- ⌘ Like API's (Application Programming Interface)
- ⌘ Services powering Client Applications



Great for prototyping and agile development

Superfast and highly scalable

- Used in Production by big companies like:
- Paypal, Netflix, Uber, LinkedIn, NASA, eBay, Groupon, Mozilla, Aliexpress...

Source: <https://trio.dev/blog/companies-use-node-js>

— **Ideal for:**

Highly-scalable, data-intensive  
and real-time apps

## — Used by:

# 15+ Popular Companies Using Node.js in 2024

Here are the companies using Node.js: Netflix, NASA, Trello, PayPal, LinkedIn, Walmart, Uber, Twitter, Yahoo, eBay, GoDaddy, and got much better results.



Tejas Kaneriya

March 3, 2021

⌚ 15 mins read

Last Updated February 02, 2024



Bron: <https://www.simform.com/blog/companies-using-nodejs/>

1. Netflix lowers startup time with Node.js.
2. NASA improved database access time with Node.js.
3. Trello achieved quick prototyping with Node.js.
4. PayPal decreased its loading time with Node.js.
5. LinkedIn improved its app performance with Node.js.
6. Walmart improved website experience with Node.js.
7. Uber eases ride processing with Node.js.
8. Twitter achieves 5 seconds loading speed with Node.js.
9. Yahoo simplified feature releases with Node.js.
10. eBay reduced resource utilization with Node.js.
11. GoDaddy showed 4x higher end-user performance with Node.js.
12. Groupon achieved scalability with Node.js.
13. Medium speeds up content processing with Node.js.
14. Yandex created numerous APIs with Node.js.
15. Citibank achieved 50% lower latency with Node.js.

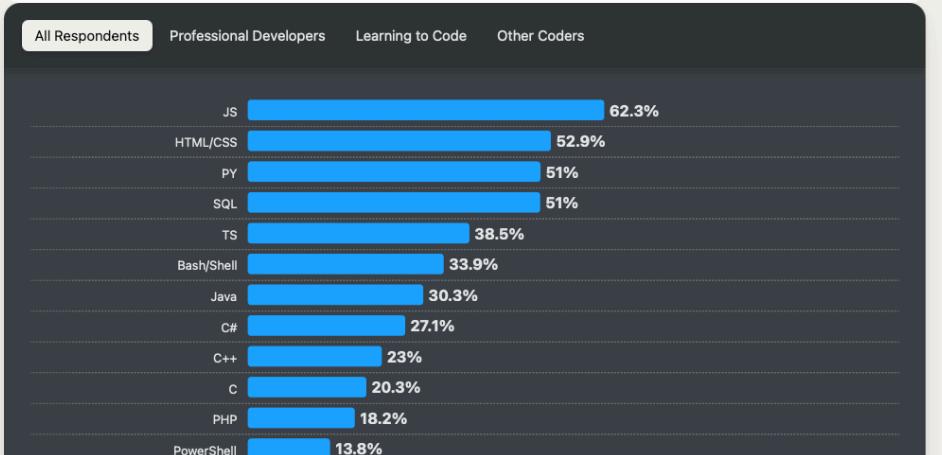
# — StackOverflow Developer Survey 2024

## Most popular technologies ..... 2.1

### Programming, scripting, and markup languages

JavaScript has been a mainstay in the developer survey and on Stack Overflow since our first survey. The most popular programming language has been JavaScript every year we have done the survey except for 2013 and 2014, when SQL was the most popular language.

💡 Which **programming, scripting, and markup languages** have you done extensive development work in over the past year, and which do you want to work in over the next year? (If you both worked with the language and want to continue to do so, please check both boxes in that row.)



### Web frameworks and technologies

Node.js and React.js are the two most common web technologies used by all respondents.

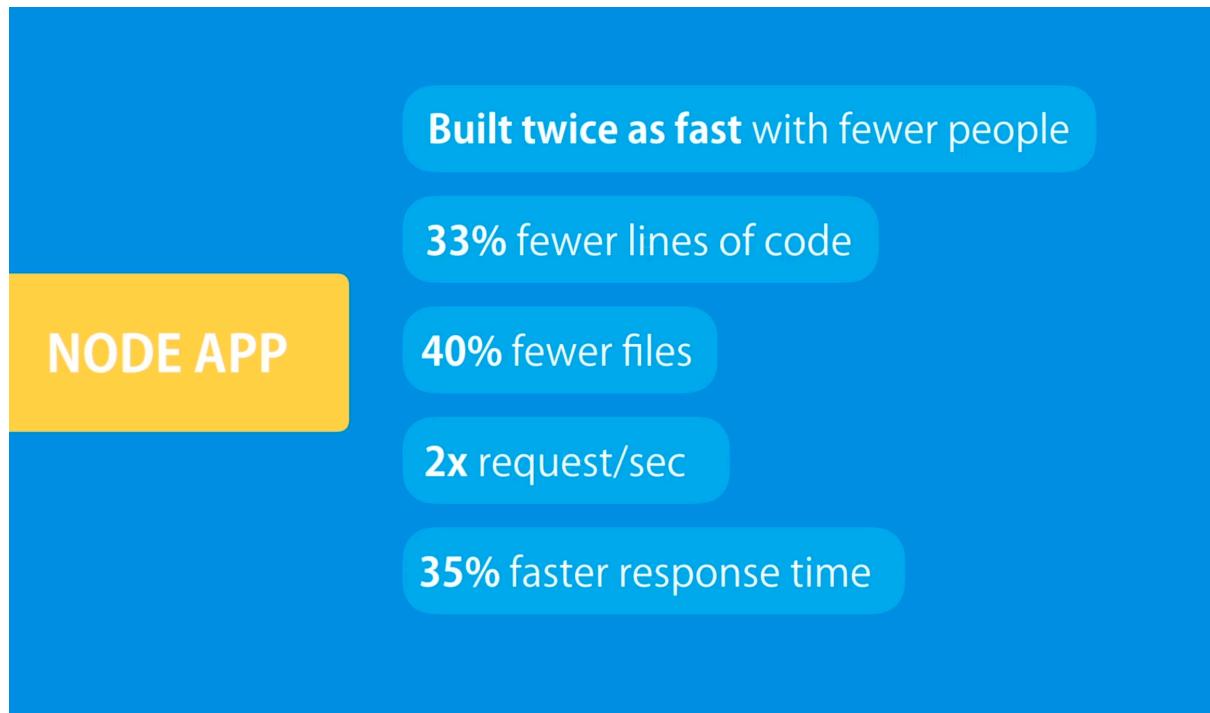
Professional Developers use both fairly equally and those learning to code use Node.js more than React (52% vs. 48%). jQuery and Express are the next two popular web technologies for all respondents, and jQuery is used more by Professional Developers than those learning to code (24% vs 18%), whereas Express is used more by those learning than professionals (25% vs. 20%).

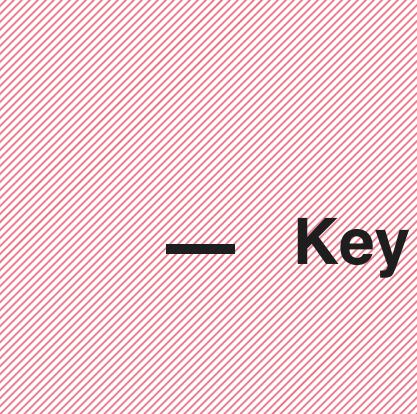
Next.js moved from 11th place in 2022 to 6th this year, likely driven by its popularity with those learning to code.



## — Case Paypal

Java Spring Application -> Node.js





## — Key points

Great for prototyping and agile development

Super fast and highly scalable

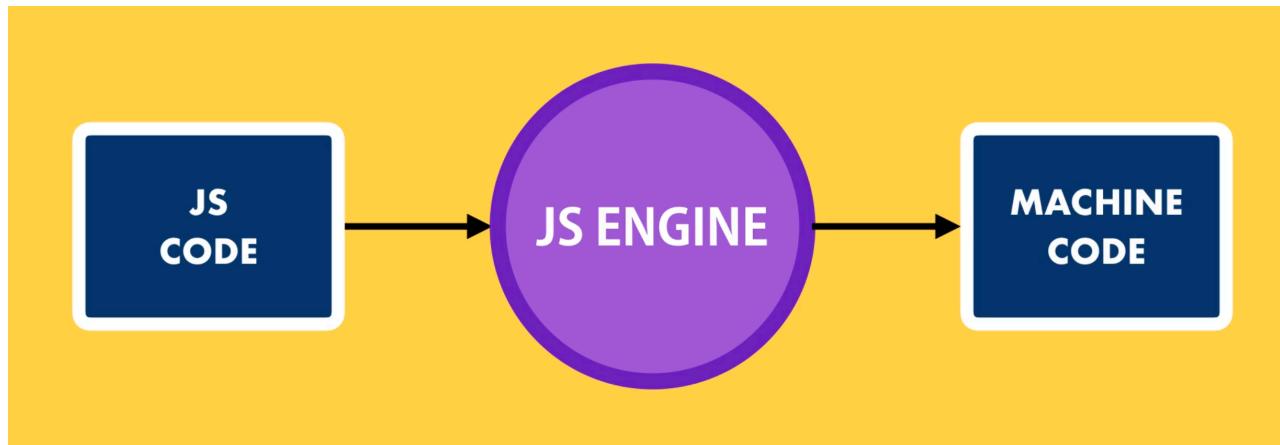
JavaScript everywhere

Cleaner and more consistent code

Large ecosystem of open-source libs

## — Before Node:

Javascript was meant for inside the browser



Different browsers = different engines!



Chakra



SpiderMonkey



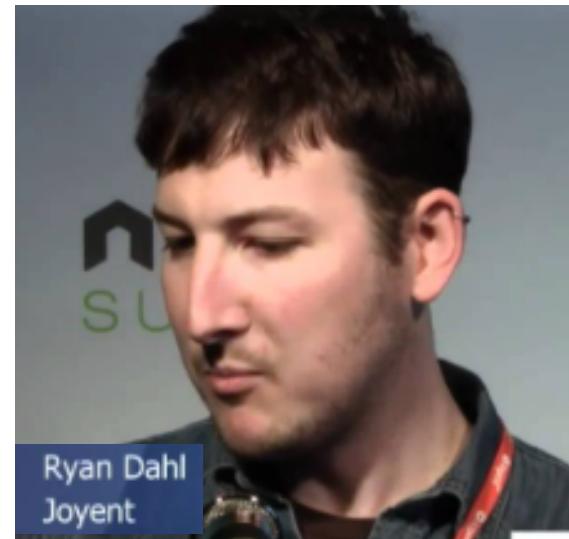
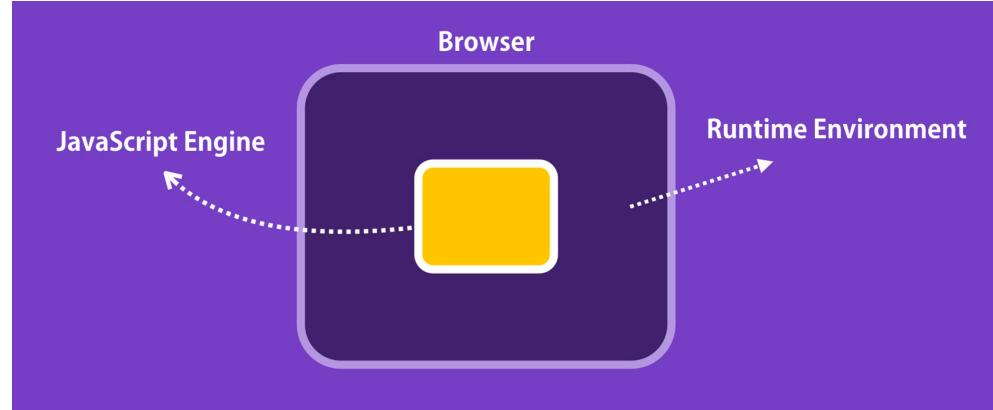
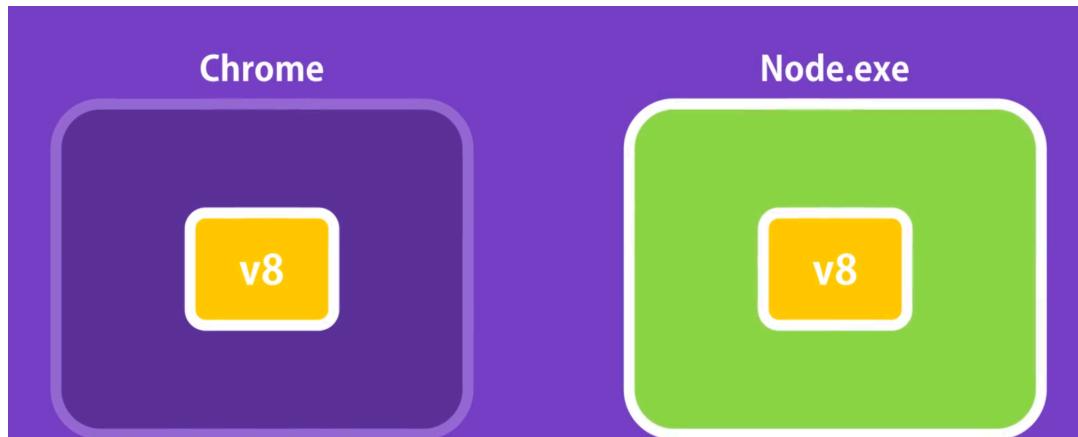
v8

## — Browser provides runtime environment

Elements like  
document & window

```
document.getElementById('');
```

In 2009 Ryan Dahl -> JS outside the browser!  
V8 embedded in C++ program = Node



## — Node

- Uses objects too
- Differs from browsers

```
document.getElementById('');  
fs.readFile()  
http.createServer()
```

**Recap:** Node is an application containing a V8 JS engine + additional modules for extra functionality. (File System, network ...)

Is not a FRAMEWORK! <> ASP.net, Rails, Django

Node is NOT a  
programming language!

Node is NOT a framework!

I prefer Node to  
ASP.NET



## — Hoe does Node work?

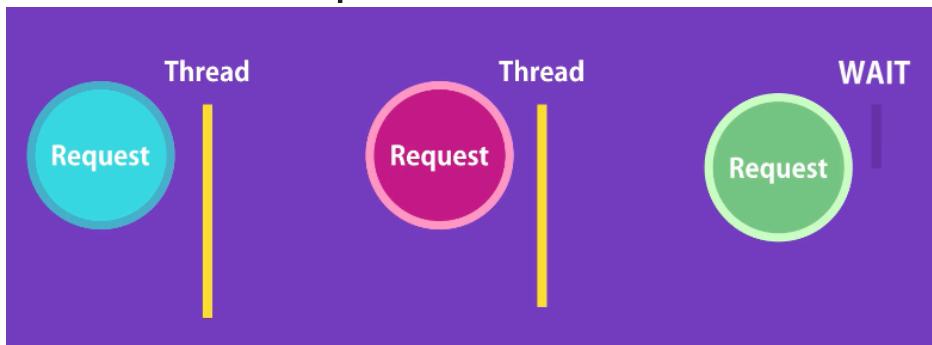
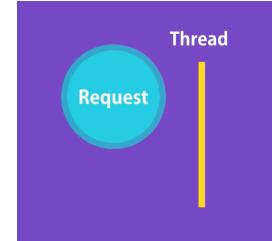
Node applications are

- Very scalable
- Real-time
- Non-blocking -> Asynchronous
- Javascript

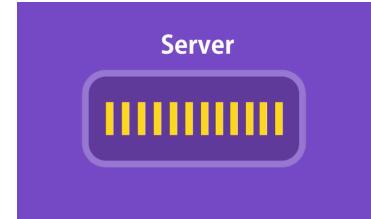


## — Synchronous (usually not in Node)

- Request comes in (e.g. DB call)
- Thread is allocated and waits for end
- New request = new thread

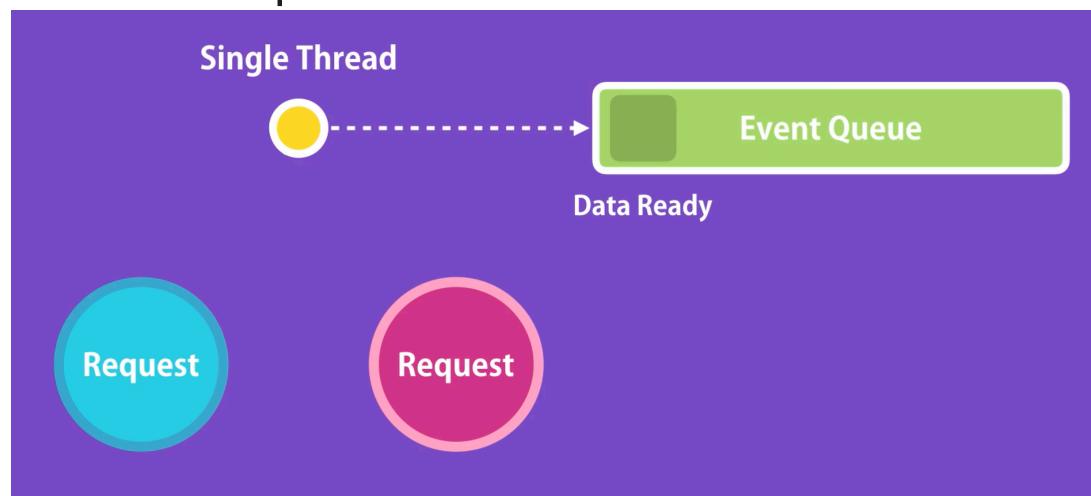


- With many clients: threads get exhausted!
- -> clients have to wait or deploy more hardware
- E.g. default ASP.net



## — With Node (asynchronous)

- js Only 1 thread. Handles all requests
- js Event Queue contains responses (is constantly monitored)
- js E.g. read from DB -> While data is downloaded
- js Thread handles next request



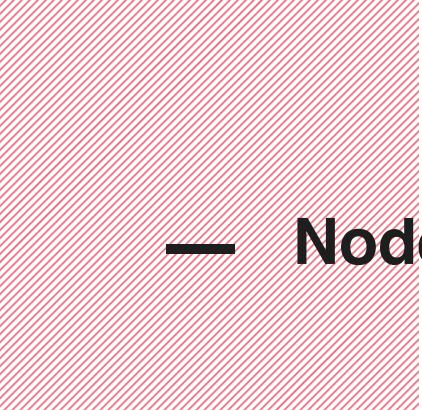
## — What is asynchronous?

Restaurant analogy:

- Waiter writes order by order (Asynchronous)
- Waiter writes down one order and waits until it's ready to deliver it and write the next order down. (Synchronous)

Several waiters = threads



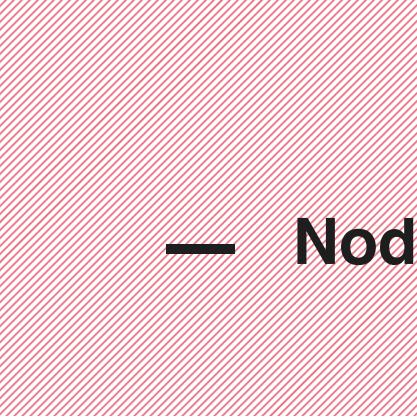


## — Node is good for:

Intensive I/O Applications (++ disk or network access)

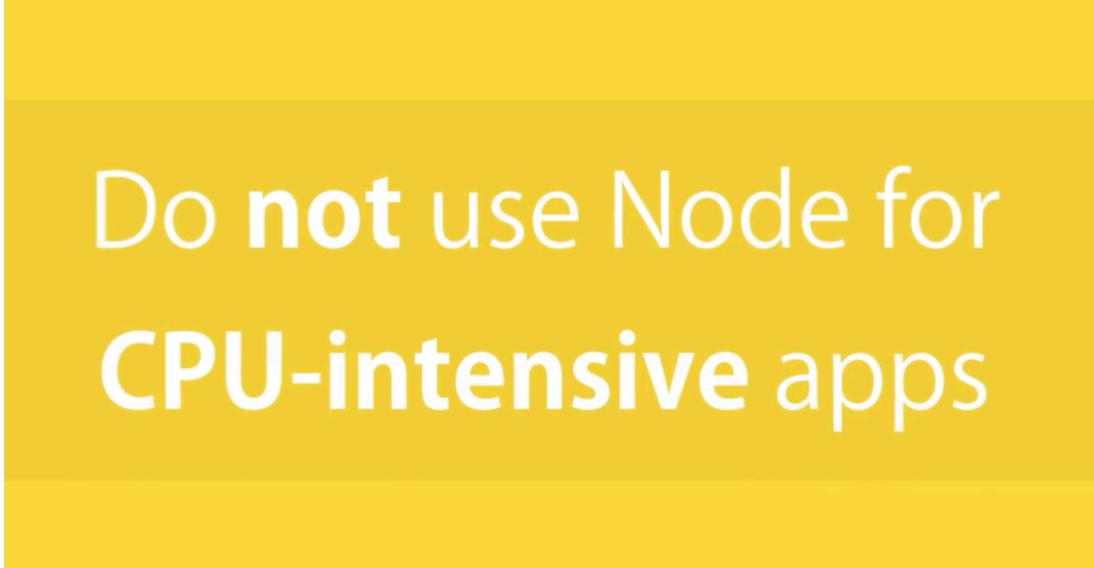


Node is ideal for  
**I/O-intensive** apps



## — Node is not good for:

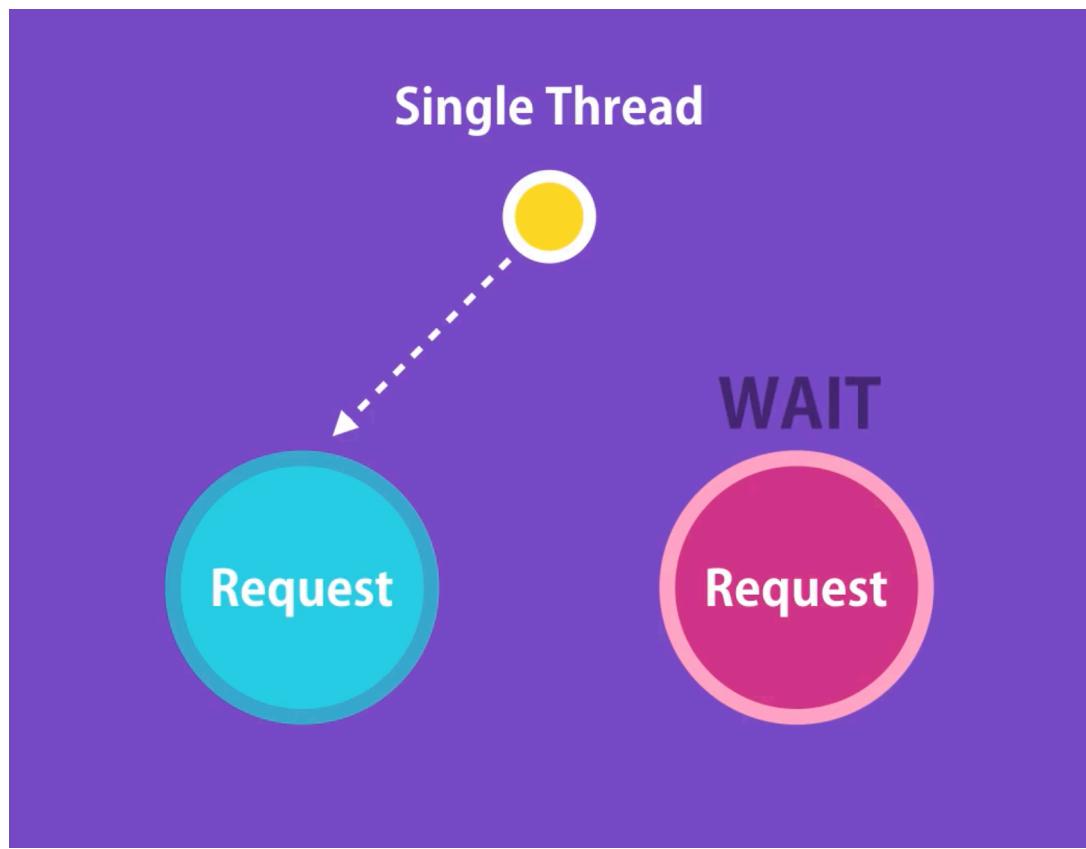
CPU Intensive Applications (Video-encoding, Photo ...)



Do **not** use Node for  
**CPU-intensive** apps

## — Single thread with CPU intensive use

With CPU intensive operations, the other clients need to wait until the calculations are finished (CPU is free)



## — Install Node

<https://nodejs.org/en/>

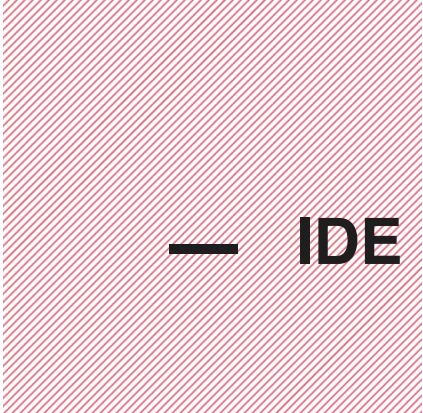
Windows: <https://nodejs.org/dist/v22.12.0/node-v22.12.0-x64.msi>

MacOS 64-bit & ARM: <https://nodejs.org/dist/v22.12.0/node-v22.12.0.pkg> (**or** brew install node)

Linux: <https://nodejs.org/dist/v22.12.0/node-v22.12.0-linux-x64.tar.xz> (**or** sudo apt install nodejs)

Docker: [https://hub.docker.com/\\_/node/](https://hub.docker.com/_/node/)

Others: <https://nodejs.org/en/download/>



## — IDE

### Free Choice of IDE

Visual Studio Code, WebStorm, Atom, Sublime, IntelliJ Idea, Komodo, Webmatrix, Koding, Brackets ...

Slides: VS Code <https://code.visualstudio.com/docs/nodejs/nodejs-tutorial>

Plugins: npm intellisense, ES Lint, Prettier, REST Client (Document This)

Also see: <https://docs.microsoft.com/en-us/shows/beginners-series-to-nodejs/how-to-setup-vs-code-for-nodejs-development-5-of-26>

# — JavaScript vs TypeScript vs JSDoc

Javascript: known from previous course web development

TypeScript: Developed by MS, **Static Typing**, ECMAScript compatibility, own compiler. Used in Angular?

JSDoc: Markup language and documentation tool. Type checking with Annotations!

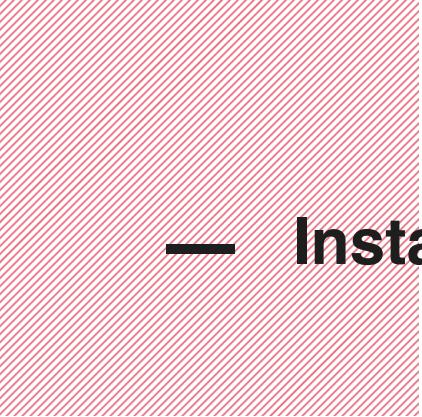
<https://jsdoc.app/>

The screenshot shows a code editor with the following code:

```
1 // @ts-check
2
3 /**
4 * Say something to the console
5 * @param {string} message - message
6 * @return {void}
7 */
8 function saySomething(message) {
9     console.log(message);
10
11     Type 'string' is not assignable to type 'void'. ts(2322)
12     View Problem Quick Fix... (⌘.)
13     return 'foo';
14 }
15
16     Argument of type '{}' is not assignable to parameter of type
17     'string'. ts(2345)
18     View Problem Quick Fix... (⌘.)
19
20     saySomething({});
```

Annotations and TypeScript errors are highlighted with red arrows and callouts:

- An annotation `@ts-check` is annotated with "Enable TS type checking".
- A JSDoc block with parameters and return types is annotated with "Define types in JSDoc".
- A TypeScript error message "Type 'string' is not assignable to type 'void'. ts(2322)" is annotated with "Receive type validations and warnings!".
- A second TypeScript error message "Argument of type '{}' is not assignable to parameter of type 'string'. ts(2345)" is also annotated with "Receive type validations and warnings!".



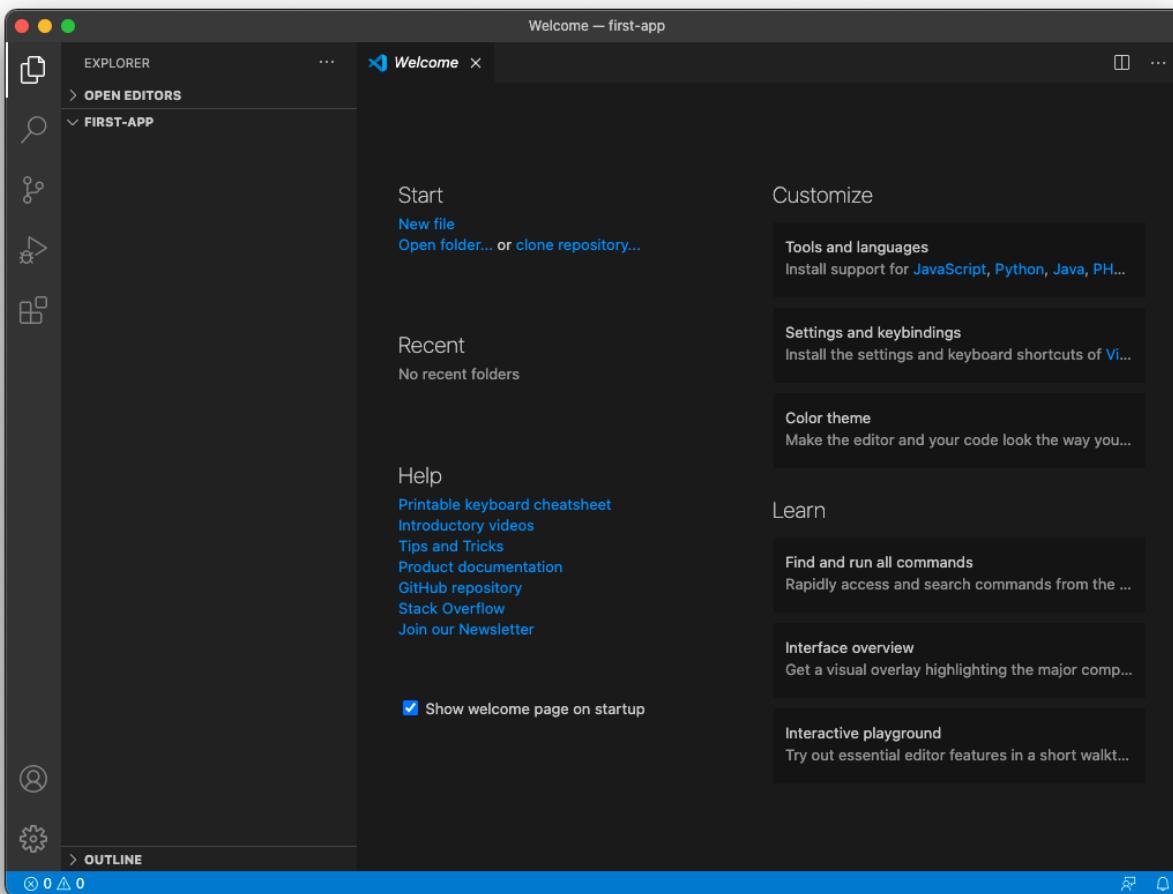
## — Install

```
~ ➤ brew install node
Running `brew update --preinstall`...
...
...
~ ➤ node --version
v23.5.0
~ ➤
```

# Install Latest Stable Version Command line or installer

## — First application

```
milan@Dev ~ mkdir first-app  
milan@Dev ~ cd first-app  
milan@first-app ~ code .
```



## — First application (cont.)

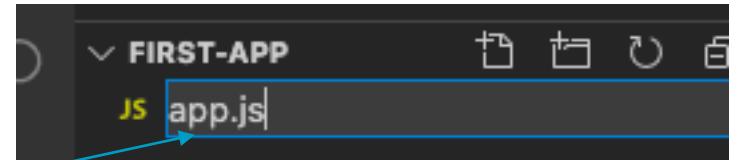
Create a new file app.js

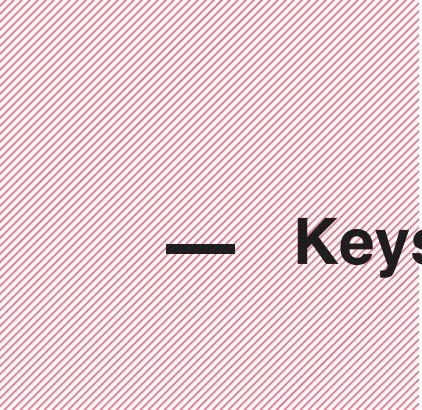
```
function sayHello(name){  
    console.log('Hello ' + name );  
}  
sayHello('Vives');
```

Run Code:

```
milan@first-app ~ node app.js  
Hello Vives  
milan@first-app ~
```

Try: `console.log(window);`





## — Keys VS Code

Ctrl + ` = open integrated terminal

Ctrl + Shift + ` = create new terminal

Ctrl + Shift + P = show Command Palette

Shift+Alt+↓ / ↑ = copy current line lijn to above/under

Ctrl (Cmd mac) + Shift + L = select all instances of current selection

Alt + Shift + A: comment/uncomment block

Ctrl (Cmd mac) + Shift + : = comment/uncomment line

<https://code.visualstudio.com/shortcuts/keyboard-shortcuts-windows.pdf>

<https://code.visualstudio.com/shortcuts/keyboard-shortcuts-macos.pdf>

## — Lab:

Download the starter from Github Classroom (see Toledo for link). Write a program that divides the students in 7 different groups based on their date of birth (YYYYMMDD hardcoded) (Tip modulus-division)

Extra 1: User is asked to input his date of birth in terminal

```
milan@first-app ~ node groepen.js
Insert date of birth (YYYYMMDD): 20201010
You are assigned to group 5
milan@first-app ~
```

Extra 2: see next slide

## — Lab cont.

Extra 2: Write a switch statement where depending on picked group another Lorem Ipsum sentence is shown.

```
milan@first-app ~ node groepen-switch.js
Insert date of birth (YYYYMMDD): 20201010
You are assigned to group 5
Your quote of the day is:
Sed ut perspiciatis unde omnis iste natus error sit
voluptatem accusantium doloremque laudantium, totam rem
aperiam, eaque ipsa quae ab illo inventore veritatis et
quasi architecto beatae vitae dicta sunt explicabo.
milan@first-app ~
```