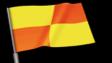
Nodes.js Basic

cuong@techmaster.vn



Slide tóm tắt bài lab thực hành tại Techmaster

Để nắm vững kỹ năng lập trình Node.js bạn cần thực hành tại phòng lab theo các ví dụ và dự án cụ thể





Nodes.js

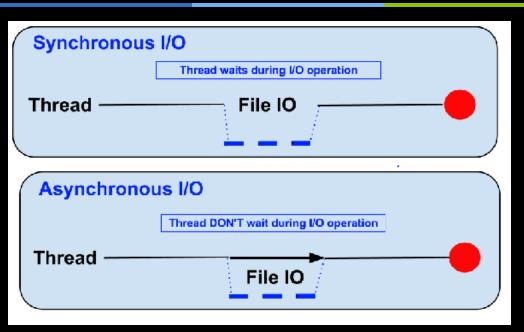
- Node.js chay trên Chrome's JavaScript run time
 V8 https://code.google.com/p/v8-wiki/
- Node.js sử dụng event driven, non-blocking I/O
- PHP: resquest response, blocking I/O

Tại sao xài Node.js?

- Phong trào? Không, để lập trình Node chỉ cần JavaScript cả server lẫn client
- Chạy nhanh hơn PHP, tốn ít RAM hơn Java trong hầu hết các ứng dụng
- JavaScript dễ học, phổ biến.
- Node module đa dạng

Sava Script V8 3.40 3.41 491,468 373 1% 1% 1% 100% ava 20.94 20.95 567,276 1377 2% 9% 1% 100% ava 20.94 20.95 567,276 1377 2% 9% 1% 100% ava 20.94 20.95 567,276 1377 2% 9% 1% 100% ava 20.95 20	Program Source Code C	PU secs E	Elapsed secs	Memory KB C	Code B	≈ CPU Load	Program Source Co	ode CPU secs E	lapsed secs	Memory KB C	ode B	≈ CPU Load
## Phi	regex-dna						fannkuch-redux					
Spectral-norm Spectral-nor	JavaScript V8	3.40	3.41	491,468	373	1% 1% 1% 100%	JavaScript V8	75.22	75.24	8,364	539	0% 0% 1% 100%
Second Series 15.71 15.71 16,472 328 0% 0% 1% 100% 100% 17.10 25,552 514 1% 0% 0% 100% 100% 17.10 25,552 514 1% 0% 0% 100% 100% 17.10 17	Java	20.94	20.95	567,276	1377	2% 9% 1% 100%	PHP	2,958.79	2,959.76	3,264	441	0% 0% 0% 100%
Php 431.80 431.94 7,240 397 0% 0% 0% 100% 10	spectral-norm						spectral-norm					
Paralle Para	JavaScript V8	15.71	15.71	16,472	328	0% 0% 1% 100%	JavaScript V8	15.71	15.71	16,472	328	0% 0% 1% 100%
Sava Script V8	Java	17.09	17.10	25,552	514	1% 0% 0% 100%	PHP	431.80	431.94	7,240	397	0% 0% 0% 100%
PHP 703.90 704.12 3,340 1082 0% 0% 0% 100% binary-trees JavaScript V8 35.90 35.91 10,492 1287 1% 0% 1% 100% ava 22.67 23,812 1424 1% 0% 1% 100% binary-trees JavaScript V8 95.27 95.37 71,280 1249 1% 0% 0% 100% binary-trees JavaScript V8 95.27 95.37 71,280 1249 1% 0% 0% 100% binary-trees JavaScript V8 95.27 95.37 71,280 1249 1% 0% 0% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 00% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 00% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 00% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 00% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 00% 100% binary-trees JavaScript V8 49.09 49.17 925,688 467 1% 1% 00% 100% binary-trees JavaScript V8 16.84 16.87 515,100 584 0% 1% 1% 100% 100% binary-trees JavaScript V8 21.88 21.91 54,016 923 1% 1% 00% 100% 100% 100% 100% 100% 100	fannkuch-redux						n-body					
Descript V8 35.90 35.91 10,492 1287 1% 0% 1% 100% 1	JavaScript V8	75.22	75.24	8,364	539	0% 0% 1% 100%	JavaScript V8	35.90	35.91	10,492	1287	1% 0% 1% 100%
JavaScript V8	Java	69.87	69.88	41,336	1282	0% 0% 0% 100%	PHP	703.90	704.12	3,340	1082	0% 0% 0% 100%
PHP 640.28 641.07 1,025,292 472 0% 0% 0% 100% regex-dna SavaScript V8 95.27 95.37 71,280 1249 1% 0% 0% 1010% lava 34.78 34.84 1,119,084 1844 0% 1% 1% 100% loom lava 34.78 34.84 1,119,084 1844 0% 1% 1% 100% loom lava 34.78 49.09 49.17 925,688 467 1% 1% 0% 100% lava 16.84 16.87 515,100 584 0% 1% 1% 100% lava 5.35 5.36 32,916 2457 0% 1% 1% 100% lava 5.35 5.36 32,916 2457 0% 1% 1% 100% lava 5.35 5.36 32,916 2457 0% 1% 1% 100% lava 1.63 1.63 524,612 745 0% 2% 1% 99% lava 1.63 1.63 524	n-body						binary-trees					
Renucleotide ReavaScript V8 95.27 95.37 71,280 1249 1% 0% 0% 100% Reava 34.78 34.84 1,119,084 1844 0% 1% 1% 100% Replace of the state of the stat	JavaScript V8	35.90	35.91	10,492	1287	1% 0% 1% 100%	JavaScript V8	49.09	49.17	925,688	467	1% 1% 0% 100%
Sava Script V8	Java	22.67	22.67	23,812	1424	1% 0% 1% 100%	PHP	640.28	641.07	1,025,292	472	0% 0% 0% 100%
PHP 26.34 26.35 219,356 459 0% 1% 0% 100% fasta BavaScript V8 49.09 49.17 925,688 467 1% 1% 0% 100% fasta BavaScript V8 21.88 21.91 54,016 923 1% 1% 0% 100% fasta BavaScript V8 21.88 21.91 54,016 923 1% 1% 0% 100% fasta BavaScript V8 21.88 21.91 54,016 923 1% 1% 0% 100% fasta BavaScript V8 21.88 21.91 54,016 923 1% 1% 0% 100% fasta BavaScript V8 7.39 7.40 397,064 787 0% 1% 1% 100% fasta BavaScript V8 7.39 7.40 397,064 787 0% 1% 0% 100% fasta BavaScript V8 7.39 7.40 397,064 787 0% 1% 0% 100% fasta Bava 0.28 0.30 424 938 0% 3% 0% 100% fasta-redux No program Bava 0.18 0.18 ? 1443 0% 0% 0% 100% fasta-redux BavaScript V8 Bad Output PHP 26.34 26.35 219,356 459 0% 1% 0% 100% fasta BavaScript V8 21.88 21.91 54,016 923 1% 1% 0% 100% fasta-redux No program Bava 0.18 0.18 ? 1443 0% 0% 0% 100% fasta-redux BavaScript V8 Bad Output PHP 0.68 0.68 3,588 1110 3% 0% 1% 100% fasta-redux BavaScript V8 Bad Output BavaScript V8 Date	k-nucleotide						regex-dna					
fasta	JavaScript V8	95.27	95.37	71,280	1249	1% 0% 0% 100%	JavaScript V8	3.40	3.41	491,468	373	1% 1% 1% 100%
Sava	Java	34.78	34.84	1,119,084	1844	0% 1% 1% 100%	PHP	26.34	26.35	219,356	459	0% 1% 0% 100%
PHP 148.29 151.55 3,332 1030 0% 0% 0% 100% fasta	binary-trees						fasta					
Fasta PavaScript V8	JavaScript V8	49.09	49.17	925,688	467	1% 1% 0% 100%	JavaScript V8	21.88	21.91	54,016	923	1% 1% 0% 100%
Sava	Java	16.84	16.87	515,100	584	0% 1% 1% 100%	PHP	148.29	151.55	3,332	1030	0% 0% 0% 100%
PHP 6.41 6.42 444,380 262 0% 0% 0% 100% or everse-complement Sava	fasta						reverse-compleme	ent				
Reverse-complement Reverse	JavaScript V8	21.88	21.91	54,016	923	1% 1% 0% 100%	JavaScript V8	7.39	7.40	397,064	787	0% 1% 0% 100%
Sava 7.39 7.40 397,064 787 0% 1% 0% 100% 100	Java	5.35	5.36	32,916	2457	0% 1% 1% 100%	PHP	6.41	6.42	444,380	262	0% 0% 0% 100%
PHP 46.69 46.71 247,820 1036 1% 0% 0% 100% pidigits No program N	reverse-complement						k-nucleotide					
pidigits No program No program No program No program No program PHP O.11 O.12 O.28 O.30 No program No	JavaScript V8	7.39	7.40	397,064	787	0% 1% 0% 100%	JavaScript V8	95.27	95.37	71,280	1249	1% 0% 0% 100%
No program No program No program No program No program PHP O.11 O.12 O.28 O.30 A24 O.30 A24 O.30 O.3	Java	1.63	1.63	524,612	745	0% 2% 1% 99%	PHP	46.69	46.71	247,820	1036	1% 0% 0% 100%
PHP 0.11 0.12 ? 394 0% 0% 0% 100% fasta-redux No program No program O.18 0.18 0.18 ? 1443 0% 0% 0% 100% PHP 0.68 0.68 3,588 1110 3% 0% 1% 100% mandelbrot JavaScript V8 Bad Output Bay O.28 0.30 424 938 0% 3% 0% 100% 100% PHP 0.68 0.68 3,588 1110 3% 0% 1% 100% PHP 0.68 0.68 3,588 1110 3% 0% 1% 100% PHP 0.68 0.68 0.68 3,588 1110 3% 0% 1% 100% PHP 0.68 0.68 0.68 0.68 0.68 0.68 0.68 0.68	pidigits						pidigits					
fasta-redux No program No pr			No program						No program			
No program No program No program No program No program PHP 0.68 0.68 3,588 1110 3% 0% 1% 100% mandelbrot JavaScript V8 Bad Output Bad Output Bad Output	Java	0.28	0.30	424	938	0% 3% 0% 100%	PHP	0.11	0.12	?	394	0% 0% 0% 100%
PHP 0.68 0.68 3,588 1110 3% 0% 1% 100% mandelbrot JavaScript V8	fasta-redux						fasta-redux					
mandelbrot JavaScript V8 Bad Output Bad Output Bad Output			No program									
JavaScript V8 Bad Output Bad Output	Java	0.18	0.18	?	1443	0% 0% 0% 100%		0.68	0.68	3,588	1110	3% 0% 1% 100%
add output	mandelbrot											
ava 0.26 0.26 680 796 0% 0% 8% 100% PHP 4.71 4.71 3,272 443 0% 1% 1% 100%	JavaScript V8		Bad Output				•					
	Java	0.26	0.26	680	796	0% 0% 8% 100%	PHP	4.71	4.71	3,272	443	0% 1% 1% 100%

Block vs Non-Blocking I/O



The cost of I/O

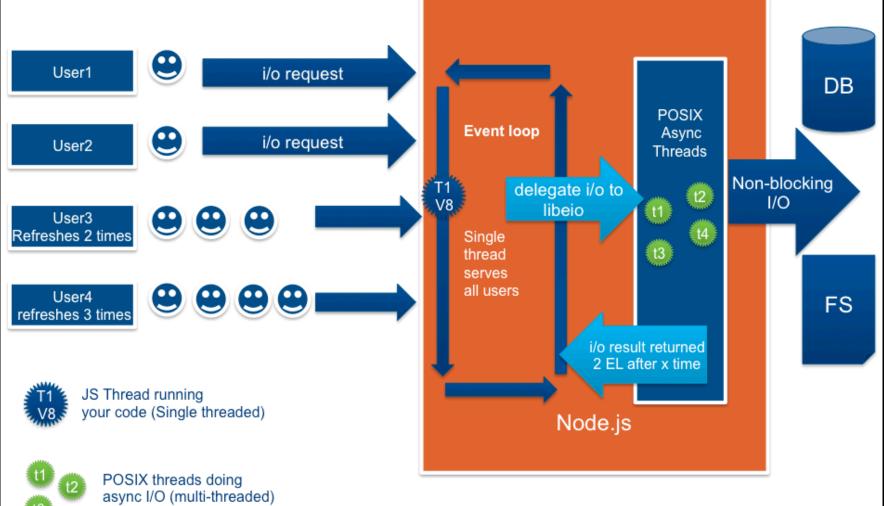
```
      L1-cache
      3 cycles

      L2-cache
      14 cycles

      RAM
      250 cycles

      Disk
      41 000 000 cycles

      Network
      240 000 000 cycles
```



Nodes vs PHP http://webapplog.com/php-vs-node-js/

Cài Node.js trên Mac

Cách 1: Sử dụng HomeBrew

http://thechangelog.com/install-node-js-with-homebrew-on-os-x/

Cách 2: Tải bộ cài rồi cài đặt

http://coolestguidesontheplanet.com/installing-node-js-on-osx-10-10-yosemite/

Kiểm tra sau khi cài đặt

```
$ which node
/usr/local/bin/node
$ node −v
v0.10.33
$ npm -v
```

2.1.8

http://techmaster.vn



Sử dụng node để chạy JavaScript

nano helloNode.js

```
var name = "Node.js";
console.log("Hello " + name);
```

Soạn thảo helloNode.js bằng lệnh nano hoặc code editor như SublimeText

node helloNode.js



JS Tutorial

1S HOME

JS Introduction JS Where To

JS Output

JS Syntax JS Statements

JS Comments JS Variables

JS Strings JS String Methods

JS Numbers JS Number Methods

JS Operators JS Math

JS Dates JS Date Methods

JS Arrays

1S Booleans JS Comparisons

JS Conditions

JS Switch

JS Loop For

JS RegExp

JS Type Of

1S Errors JS Debugging

JS Hoisting JS Strict Mode

JS Style Guide

JS Mistakes

JS Best Practices

JS Conversion

JS Loop While 1S Breaks

« Previous

JS Data Types You will soon JS Functions

JS Objects JS Scope

JavaScri JS Events

It is difficult Your code m

inva

JavaSd

Often, when indications v

JS Array Methods JavaScri

Searching fo

Debugging is

Built-in debu With a debug variables wh

Normally, ot F12 key, and

The cons

If your brows window:

Example

JS Performance JS Reserved Words <!DOCTYPE JS JSON <html>

Học JavaScript trong 21 giờ

A Smarter Way to Learn **JavaScript**

The new tech-assisted approach that requires half the effort



Read a 10-minute chapter of this book to get each concept.



Make the knowledge stick. Do the chapter's free interactive exercises at ASmarterWayToLearn.com.

Mark Myers



JON DUCKETT

Công cụ, IDE lập trình Node

- nano, vim
- Sublime Text, Text Wrangler, TextMate, NotePad++
- WebStorm, 99\$ đáng đồng tiền bát gạo. Plugins: Node.js, Spy-js, Stylus

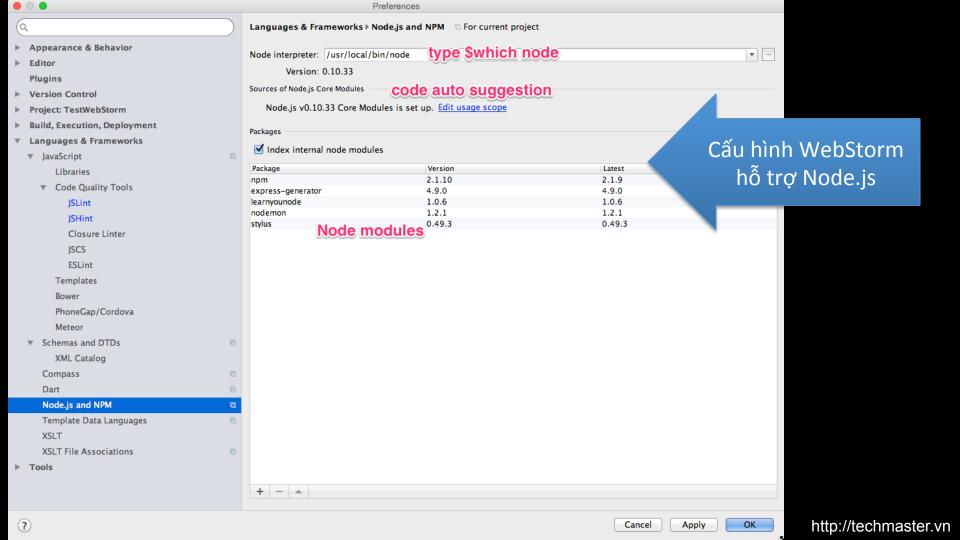
```
watcher-spawn.is - TestWebStorm - [/Volumes/DATA/notesis/Tes
TestWebStorm > node > is watcher-spawn.js

    watcher-argv.js 
    ×

                                                   is watcher-spawn.js ×
TestWebStorm (/Volumes/DATA/notesjs/TestWebStd
                                         <u>-</u>/*...*/
                                          "use strict":
▼ □ node
     playboy.jpg
                                               fs = require('fs'),
     servefile.is
     simplehttp.js
                                               spawn = require('child_process').spawn,
     target.txt
                                               filename = process.arqv[2];
     watcher-argv.js
     watcher-spawn.is
                                        hif (!filename) {
  node modules
                                               throw Error("A file to watch must be specified!");
   🖥 alert.html
   console.html
                                         docwrite.html
   docwrite2.html
                                               let ls = spawn('ls', ['-lh', filename]);
   helloDollv.html
                                               ls.stdout.pipe(process.stdout);
                                    18

■ test.is

                                    19
                                         å});
 External Libraries
                                          console.log("Now watching " + filename + " for changes...");
```



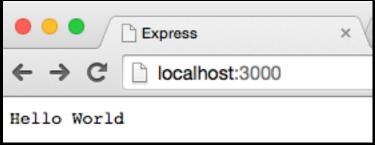


Vào đây xem chi tiết ý nghĩa của package.json http://browsenpm.org/package.json



Tạo web server đơn giản

```
var http = require('http');
http.createServer(function(reg, res) {
    res.writeHead(200, {'Content-Type': 'text/plain'});
    res.end('Hello World\n');
}).listen(3000);
console.log('Server is running at http://localhost:3000');
cuongs-Mac-Pro:basic techmaster$ node simplehttp.js
Server is running at http://localhost:3000
```



Viết tách ra, thay var bằng const

```
const http = require('http');
const server = http.createServer();
server.on('request', function(req, res) {
    res.writeHead(200, {'Content-Type': 'text/plain'});
    res.end('Hello World\n');
});
server_listen(3000);
console.log('Server is running at http://localhost:3000');
```

const http = require('http');

Module đóng gói các hàm JavaScript hữu ích được export ra để tái sử dụng lại

require('module_name') trả về object chứa các hàm

Return JSON

```
var http = require('http');
var server = http.createServer();
server.on('request', function(req, res) {
    res.setHeader('Content-Type', 'application/json');
    res.writeHead(200, 'json content');
    res.write('{"spy_agent": "0X-13"}');
    res.end();
});
server.listen(3000);
console.log('JSON server is running at http://localhost:3000');
```

Serve file

```
const http = require('http');
const fs = require('fs');
const server = http.createServer();
server.on('request', function(req, res){
    res.writeHead(200, {'Content-Type': 'image/jpg'});
    fs.createReadStream('./playboy.jpg').pipe(res);
});
server.listen(3000);
console.log('Serve play boy at http://localhost:3000');
```

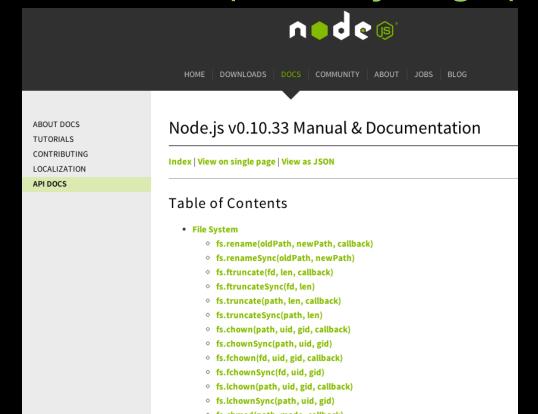
route căn bản phần luồn request

```
var http = require('http');
var url = require('url');
var server = http.createServer();
server.on('request', function(req, res) {
    var url_parsed = url.parse(reg.url, true);
    if (req.method === 'GET') {
        handleGETRequest(res, url_parsed);
});
server listen (3000);
```

```
handleGETRequest = function(res, url parsed) {
       console.log('path: ' + url parsed.pathname);
       switch (url parsed.pathname) {
           case "/":
               res.writeHead(200, {'Content-Type': 'text/plain'});
               res.end('Front page is here\n');
               break:
           case "/hi":
               res.writeHead(200, {'Content-Type': 'text/plain'});
Route
               res.end('Hello World\n');
requests
               break:
           case "/api":
               res.setHeader('Content-Type', 'application/json');
               res.writeHead(200, 'json content');
               res.write('{"spy agent": "0X-13"}');
               res_end();
               break:
```

Tìm tài liệu các Node.js module

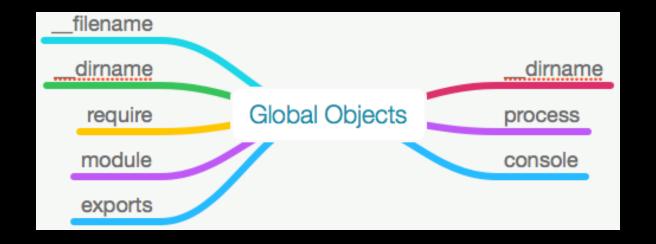
file system (fs) -> http://nodejs.org/api/fs.html





Global Object

http://nodejs.org/api/globals.html





Lập trình web app bằng framework mới nhanh được! Express!!!

http://expressjs.com/

Cài đặt express

- \$ mkdir myapp & cd myapp
- \$ npm init//Tao ra package.json
- \$ npm install express --save

npm: Notes Package Management

- Cài đặt quản lý phát hành các package (gói thư viện tái sử dụng)
- https://www.npmjs.org/



npm: Notes Package Management

- \$ npm ls //liệt kê các package
- \$ 1s ~/.npm //thư mục chứa Nodes package ở / Users/YourAccount/.npm
- \$ npm update //Cập nhật các package mới

npm init

```
name: (nodeproject) NodeProject
version: (1.0.0) 2.0.1
description: Cuong shows how to use npm init to create package.json
entry point: (index.js) boot.js
test command:
git repository: https://github.com/TechMaster
keywords: Techmaster.vn npm init
author: Trinh Minh Cuong
license: (ISC) MIT
```

package.json là kết quả của npm init

```
"name": "NodeProject",
"version": "2.0.1",
"description": "Cuong shows how to use npm init to create package.json",
"main": "boot.js",
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1"
"repository": {
 "type": "git",
 "url": "https://github.com/TechMaster"
"keywords": [
 "Techmaster.vn",
 "npm",
  "init"
"author": "Trinh Minh Cuong",
"license": "MIT"
```



Các buổi học tiếp theo chúng ta sẽ học lập trình web site dùng Express:

- 1- Tao API Rest
- 2- CRUD dữ liệu vào CSDL MongoDB hoặc Postgresql
- 3- Socket.IO
- 4- Native mobile app kết nối vào Socket.IO