

Visual Studio Code interface showing a terminal window and a Task Manager window.

Task Manager Window:

Name	PID	Status	User name	CPU	Memory (a...	UAC virtualizat...
lsass.exe	392	Running	SYSTEM	00	5,100 K	Not allowed
Microsoft.Photos.exe	11640	Suspended	Malinconia	00	0 K	Disabled
mongod.exe	4568	Running	NETWORK...	00	103,088 K	Not allowed
node.exe	1004	Running	Malinconia	00	7,304 K	Disabled
node.exe	2804	Running	Malinconia	00	7,256 K	Disabled
node.exe	11756	Running	Malinconia	00	7,296 K	Disabled
node.exe	3668	Running	Malinconia	00	7,284 K	Disabled
node.exe	8572	Running	Malinconia	00	7,308 K	Disabled
node.exe	12508	Running	Malinconia	00	7,312 K	Disabled
node.exe	13284	Running	Malinconia	00	7,300 K	Disabled
node.exe	14764	Running	Malinconia	00	7,380 K	Disabled
node.exe	14644	Running	Malinconia	00	7,336 K	Disabled
node.exe	2364	Running	Malinconia	00	28,608 K	Disabled
node.exe	13596	Running	Malinconia	00	28,400 K	Disabled
node.exe	14360	Running	Malinconia	00	28,236 K	Disabled
node.exe	9164	Running	Malinconia	00	28,644 K	Disabled
node.exe	10324	Running	Malinconia	00	28,296 K	Disabled
node.exe	9788	Running	Malinconia	00	28,484 K	Disabled
node.exe	13976	Running	Malinconia	00	28,792 K	Disabled
node.exe	10184	Running	Malinconia	00	28,440 K	Disabled
node.exe	6580	Running	Malinconia	00	15,904 K	Disabled
node.exe	5532	Running	Malinconia	00	28,800 K	Disabled
notepad.exe	9544	Running	Malinconia	00	6,716 K	Disabled
NVDisplay.Container...	1960	Running	SYSTEM	00	2,680 K	Not allowed
NVDisplay.Container...	2480	Running	SYSTEM	00	7,280 K	Not allowed

Terminal Window:

```
(node:2396) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency
Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
$ forever start server.js -m CLUSTER -p 8090
warn: --minUptime not set. Defaulting to: 1000ms
warn: --spinSleepTime not set. Your script will exit if it does not stay up for at least 1000ms
info: Forever processing file: server.js
(node:11496) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency
(Use 'node --trace-warnings ...' to show where the warning was created)
(node:11496) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency
Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
$
```

File Edit Selection View Go Run Terminal Help

server.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

▼ SERVER-CON-BALANCE-DE-CARGA

contenedores

DB

node_modules

routes

JS randomNumbers.js

JS routeRandom.js

src

.env

.gitignore

forever-fork-pue... U

forever-fork-pue... U

nginx.conf

nodemon-cluste... U

nodemon-cluste... U

nodemon-fork-p... U

nodemon-fork-p... U

package-lock.json

package.json

readme.md U

JS server.js

JS usuarios.js

JS server.js > ...

202 }

203

204

205

206

207

208

209

210

211

212

213

214

215

216

217

218

219

220

221

222

223

Task Manager

File Options View

Processes Performance App history Startup Users Details Services

Name	PID	Status	User name	CPU	Memory (a...	UAC virtualizat...
fontdrvhost.exe	1060	Running	UMFD-0	00	1,056 K	Disabled
GameBar.exe	10400	Running	Malinconia	00	7,048 K	Disabled
GameManagerServic...	4520	Running	SYSTEM	00	10,996 K	Not allowed
lsass.exe	392	Running	SYSTEM	00	5,048 K	Not allowed
Microsoft.Photos.exe	11640	Suspended	Malinconia	00	0 K	Disabled
mongod.exe	4568	Running	NETWORK...	00	103,048 K	Not allowed
node.exe	13612	Running	Malinconia	00	7,284 K	Disabled
node.exe	276	Running	Malinconia	00	7,248 K	Disabled
node.exe	7260	Running	Malinconia	00	7,268 K	Disabled
node.exe	3264	Running	Malinconia	00	7,296 K	Disabled
node.exe	12508	Running	Malinconia	00	7,320 K	Disabled
node.exe	14812	Running	Malinconia	00	7,280 K	Disabled
node.exe	14628	Running	Malinconia	00	7,284 K	Disabled
node.exe	13228	Running	Malinconia	00	7,308 K	Disabled
node.exe	9928	Running	Malinconia	00	15,824 K	Disabled
node.exe	13212	Running	Malinconia	00	28,404 K	Disabled
node.exe	2804	Running	Malinconia	00	7,248 K	Disabled
node.exe	3028	Running	Malinconia	00	28,324 K	Disabled
node.exe	15188	Running	Malinconia	00	28,792 K	Disabled
node.exe	960	Running	Malinconia	00	28,480 K	Disabled
node.exe	9952	Running	Malinconia	00	28,176 K	Disabled
node.exe	14200	Running	Malinconia	00	28,180 K	Disabled
node.exe	8860	Running	Malinconia	00	28,208 K	Disabled
node.exe	10060	Running	Malinconia	00	28,068 K	Disabled
node.exe	3032	Running	Malinconia	00	28,296 K	Disabled

PROBLEM

Malinconia

\$ forever list

(node:8620) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency

(node:8620) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency

info: Forever processes running

data: uid command script

uptime

data: [0] v6ha "C:\Program Files\nodejs\node.exe" C:\Exequiel\curso-coderhouse-backend\practica\servidor-con-balance-de-carga\server.js -m CLUSTER 9928 13212 C:\Users\Ma

linconia\.forever\v6ha.log 0:0:0:7.49

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)

\$

bash

End task

ga (master)

OUTLINE

TIMELINE

Ln 223, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript Go Live

4:19 PM 3/21/2023

server.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

SERVIDOR-CON-BALANCE-DE-CARGA

contenedores

DB

node_modules

routes

randomNumbers.js

routeRandom.js

src

.env

.gitignore

forever-fork-pue... U

nginx.conf

nodemon-cluste... U

nodemon-cluste... U

nodemon-fork-p... U

nodemon-fork-p... U

package-lock.json

package.json

readme.md U

server.js

usuarios.js

server.js

server.js > ...

```
202 }
203
204 const PORT = options.server.PORT
205
206 if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {
207   for(let i = 0; i < numCores; i++) {
208     cluster.fork();
209   };
210   cluster.on("exit", (worker) => {
211     console.log(`Proceso ${worker.process.pid} dejó de funcionar`);
212     cluster.fork();
213   });
214 } else {
215   const server = app.listen(PORT, () => {
216     console.log(`Servidor escuchando en puerto ${JSON.stringify(PORT)} con
el proceso ${process.pid}`);
217   })
218   server.on('error', error => {
219     console.error(`Error en el servidor ${error}`);
220   });
221 }
222
223
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

bash

+

-

🗑

⋮

^

×

```
Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
$ forever list
(node:14980) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency
(Use `node --trace-warnings ...` to show where the warning was created)
(node:14980) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency
info:   Forever processes running
data:   uid      command      script
      uptime
data:   [0] 5RSs "C:\Program Files\nodejs\node.exe" C:\Exequiel\curso-coderhouse-backend\practica\servidor-con-balance-de-carga\server.js -p 8090 15300 8572 C:\Users\Malinc
onia\.forever\5RSs.log 0:0:0:7.206

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
$
```

master*

0 0 0

Ln 223, Col 1

Spaces: 4

UTF-8

CRLF

{}

JavaScript

Go Live

4:16 PM

3/21/2023

server.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

SERVERIDOR-CON-BALANCE-DE-CARGA

contenedores

DB

node_modules

routes

randomNumbers.js

routeRandom.js

src

env

.gitignore

nginx.conf

nodemon-cluste...

nodemon-fork-p...

nodemon-fork-p...

package-lock.json

package.json

readme.md

server.js

usuarios.js

server.js

server.js > ...

202 }

203

204 const PORT = options.server.PORT

205

206 if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {

207 for(let i = 0; i < numCores; i++) {

208 cluster.fork();

209 }

210 cluster.on("exit", (worker) => {

211 console.log(`Proceso \${worker.process.pid} dejó de funcionar`);

212 cluster.fork();

213 });

214 } else {

215 const server = app.listen(PORT, () => {

216 console.log(`Servidor escuchando en puerto \${JSON.stringify(PORT)} con

217 el proceso \${process.pid}`);

218 });

219 server.on('error', error => {

220 console.error(`Error en el servidor \${error}`);

221 });

222 }

223

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

bash

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)

\$ forever list

(node:14452) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency

(Use `node --trace-warnings ...` to show where the warning was created)

(node:14452) Warning: Accessing non-existent property 'padLevels' of module exports inside circular dependency

info: Forever processes running

data: uid command script forever pid id logfile

uptime

data: [0] wBcC "C:\Program Files\nodejs\node.exe" C:\Exequiel\curso-coderhouse-backend\practica\servidor-con-balance-de-carga\server.js 3084 14304 C:\Users\Malinconia\.f

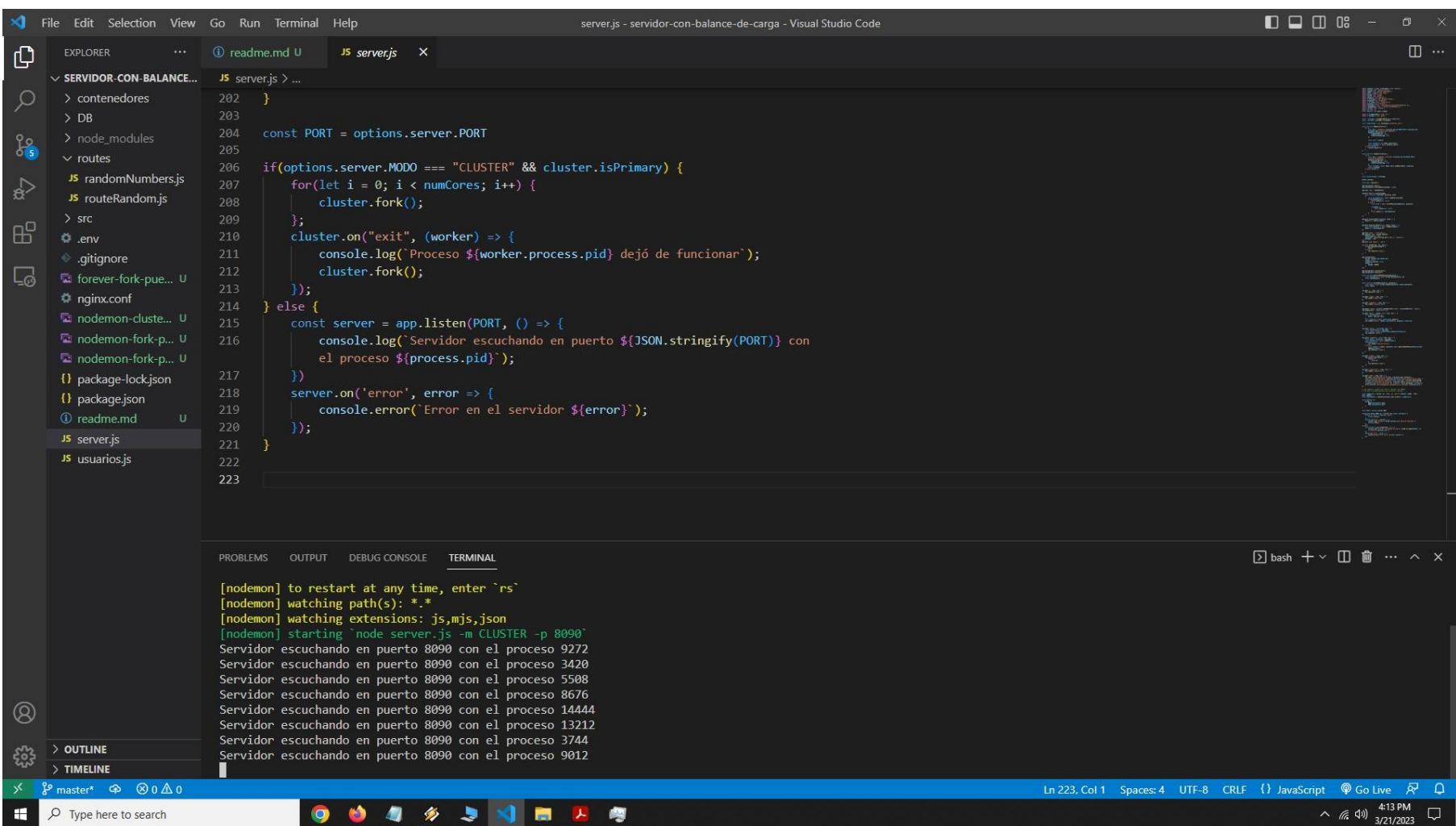
orever\wBcC.log 0:0:0:13.233

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)

\$

Ln 223, Col 1 Spaces: 4 UTF-8 CRLF {} JavaScript Go Live

4:11 PM 3/21/2023



FileEditSelectionViewGoRunTerminalHelpserver.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

SERVERIDOR-CON-BALANCE-...> contenedores> DB> node_modules> routesJS randomNumbers.jsJS routeRandom.js> src> .env> .gitignore> nginx.confnodemon-fork-p... Unodemon-fork-p... U{} package-lock.json{} package.json① readme.md UJS server.jsJS usuarios.js

202 }203204 const PORT = options.server.PORT205206 if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {207 for(let i = 0; i < numCores; i++) {208 cluster.fork();209 };210 cluster.on("exit", (worker) => {211 console.log(`Proceso \${worker.process.pid} dejó de funcionar`);212 cluster.fork();213 });214 } else {215 const server = app.listen(PORT, () => {216 console.log(`Servidor escuchando en puerto \${JSON.stringify(PORT)} con el proceso \${process.pid}`);217 });218 server.on('error', error => {219 console.error(`Error en el servidor \${error}`);220 });221 }222223

PROBLEMSOUTPUTDEBUG CONSOLETERMINAL

bash

[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node server.js -m CLUSTER`
Servidor escuchando en puerto 8080 con el proceso 3660
Servidor escuchando en puerto 8080 con el proceso 1452
Servidor escuchando en puerto 8080 con el proceso 13444
Servidor escuchando en puerto 8080 con el proceso 1384
Servidor escuchando en puerto 8080 con el proceso 224
Servidor escuchando en puerto 8080 con el proceso 12096
Servidor escuchando en puerto 8080 con el proceso 3396
Servidor escuchando en puerto 8080 con el proceso 13304

Ln 223, Col 1Spaces: 4UTF-8CRLF{} JavaScriptGo Live4:09 PM3/21/2023

server.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

SERVERIDOR-CON-BALANCE-DE-CARGA

contenedores

DB

node_modules

routes

randomNumbers.js

routeRandom.js

src

env

.gitignore

nginx.conf

nodemon-fork-p...

package-lock.json

package.json

readme.md

server.js

usuarios.js

server.js

202 }
203
204 const PORT = options.server.PORT
205
206 if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {
207 for(let i = 0; i < numCores; i++) {
208 cluster.fork();
209 };
210 cluster.on("exit", (worker) => {
211 console.log(`Proceso \${worker.process.pid} dejó de funcionar`);
212 cluster.fork();
213 });
214 } else {
215 const server = app.listen(PORT, () => {
216 console.log(`Servidor escuchando en puerto \${JSON.stringify(PORT)} con
el proceso \${process.pid}`);
217 });
218 server.on('error', error => {
219 console.error(`Error en el servidor \${error}`);
220 });
221 }
222
223

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

bash

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
\$ nodemon server.js -p 8090
[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node server.js -p 8090`
Servidor escuchando en puerto 8090 con el proceso 14368
█

Ln 223, Col 1

Spaces: 4

UTF-8

CRLF

{ } JavaScript

Go Live

4:07 PM
3/21/2023

FileEditSelectionViewGoRunTerminalHelpserver.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

> SERVID...> contenedores> DB> node_modules> routesJS randomNumbers.jsJS routeRandom.js> src> .env> .gitignore> nginx.conf() package-lock.json() package.json()@ readme.mdUJS server.jsJS usuarios.js

server.js

202 }203204 const PORT = options.server.PORT205206 if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {207 for(let i = 0; i < numCores; i++) {208 cluster.fork();209 };210 cluster.on("exit", (worker) => {211 console.log(`Proceso \${worker.process.pid} dejó de funcionar`);212 cluster.fork();213 });214 } else {215 const server = app.listen(PORT, () => {216 console.log(`Servidor escuchando en puerto \${JSON.stringify(PORT)} con el proceso \${process.pid}`);217 });218 server.on('error', error => {219 console.error(`Error en el servidor \${error}`);220 });221 }222223

PROBLEMSOUTPUTDEBUG CONSOLETERMINAL

bash

\$ nodemon server.js[nodemon] 2.0.20[nodemon] to restart at any time, enter `rs`[nodemon] watching path(s): *.*[nodemon] watching extensions: js,mjs,json[nodemon] starting `node server.js`

New version of nodemon available!
Current Version: 2.0.20
Latest Version: 2.0.21

Servidor escuchando en puerto 8080 con el proceso 9196

Ln 214, Col 9Spaces: 4UTF-8CRLF{} JavaScriptGo Live4:05 PM3/21/2023

The image shows a Visual Studio Code editor with a project named 'servidor-con-balance-de-carga'. The Explorer sidebar on the left displays the project structure, including files like 'server.js', 'routes', 'package.json', and 'package-lock.json'. The main editor window shows the 'server.js' file, which is a Node.js script using the 'cluster' module to create multiple worker processes for load balancing. The code defines a 'PORT' constant and a 'server' instance, then forks multiple workers based on the number of CPU cores.

The Output window at the bottom shows the command 'pm2 start server.js' and its output, indicating that the application is running in cluster mode with 8 instances. The output also shows the PM2 daemon successfully spawning the application.

The bottom status bar shows the file path 'Ln 223, Col 1' and the encoding 'UTF-8'.

server.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

▼ SERVER-CON-BALANCE-DE-CARGA

> contenedores

> DB

> node_modules

▼ routes

JS randomNumbers.js

JS routeRandom.js

> src

.env

.gitignore

forever-cluster-p... U

forever-cluster-p... U

forever-fork-pue... U

forever-fork-pue... U

nginx.conf

nodemon-cluste... U

nodemon-cluste... U

nodemon-fork-p... U

nodemon-fork-p... U

{ } package-lock.json

{ } package.json

pm2-fork-puerto... U

① readme.md U

JS server.js

JS usuarios.js

① readme.md U

JS server.js

202 }
203 }
204 const PORT = options.server.PORT
205
206 if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {
207 for(let i = 0; i < numCores; i++) {
208 cluster.fork();
209 };
210 cluster.on("exit", (worker) => {
211 console.log(`Proceso \${worker.process.pid} dejó de funcionar`);
212 cluster.fork();
213 });
214 } else {
215 const server = app.listen(PORT, () => {
216 console.log(`Servidor escuchando en puerto \${JSON.stringify(PORT)} con
el proceso \${process.pid}`);
217 });
218 server.on('error', error => {
219 console.error(`Error en el servidor \${error}`);
220 });
221 }
222
223

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

bash

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
\$ pm2 start server.js --watch -- 8090
[PM2] Starting C:\Exequiel\curso-coderhouse-backend\practica\servidor-con-balance-de-carga\server.js in fork_mode (1 instance)
[PM2] Done.

id	name	namespace	version	mode	pid	uptime	U	status	cpu	mem	user	watching
0	server	default	1.0.0	fork	15960	0s	0	online	0%	29.1mb	Mal...	enabled

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)
\$

Ln 223, Col 1

Spaces: 4

UTF-8

CRLF

{ } JavaScript

Go Live

5:53 PM
3/21/2023

server.js - servidor-con-balance-de-carga - Visual Studio Code

EXPLORER

SERVIDOR-CON-BALANCE-DE-CARGA

contenedores

DB

node_modules

routes

randomNumbers.js

routeRandom.js

src

env

.gitignore

forever-cluster-p...

forever-cluster-p...

forever-fork-pue...

forever-fork-pue...

nginx.conf

nodemon-cluste...

nodemon-cluste...

nodemon-fork-p...

nodemon-fork-p...

package-lock.json

package.json

readme.md

server.js

usuarios.js

readme.md

server.js

```
JS server.js > ...
202   }
203
204   const PORT = options.server.PORT
205
206   if(options.server.MODO === "CLUSTER" && cluster.isPrimary) {
207     for(let i = 0; i < numCores; i++) {
208       cluster.fork();
209     };
210     cluster.on("exit", (worker) => {
211       console.log(`Proceso ${worker.process.pid} dejó de funcionar`);
212       cluster.fork();
213     });
214   } else {
215     const server = app.listen(PORT, () => {
216       console.log(`Servidor escuchando en puerto ${JSON.stringify(PORT)} con
el proceso ${process.pid}`);
217     });
218     server.on('error', error => {
219       console.error(`Error en el servidor ${error}`);
220     });
221   }
222
223
```

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

bash

+

-

🗑

⋮

^

×

0	server	default	1.0.0	fork	5664	0s	0	online	0%	32.4mb	Mal...	disabled
---	--------	---------	-------	------	------	----	---	--------	----	--------	--------	----------

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)

\$ pm2 list

id	name	namespace	version	mode	pid	uptime	U	status	cpu	mem	user	watching
0	server	default	1.0.0	fork	5664	9s	0	online	0%	50.5mb	Mal...	disabled

Malinconia@DESKTOP-CNAGB20 MINGW64 /c/Exequiel/curso-coderhouse-backend/practica/servidor-con-balance-de-carga (master)

\$

master*

0 0 0

Ln 223, Col 1

Spaces: 4

UTF-8

CRLF

{}

JavaScript

Go Live

🔔

5:49 PM

3/21/2023