

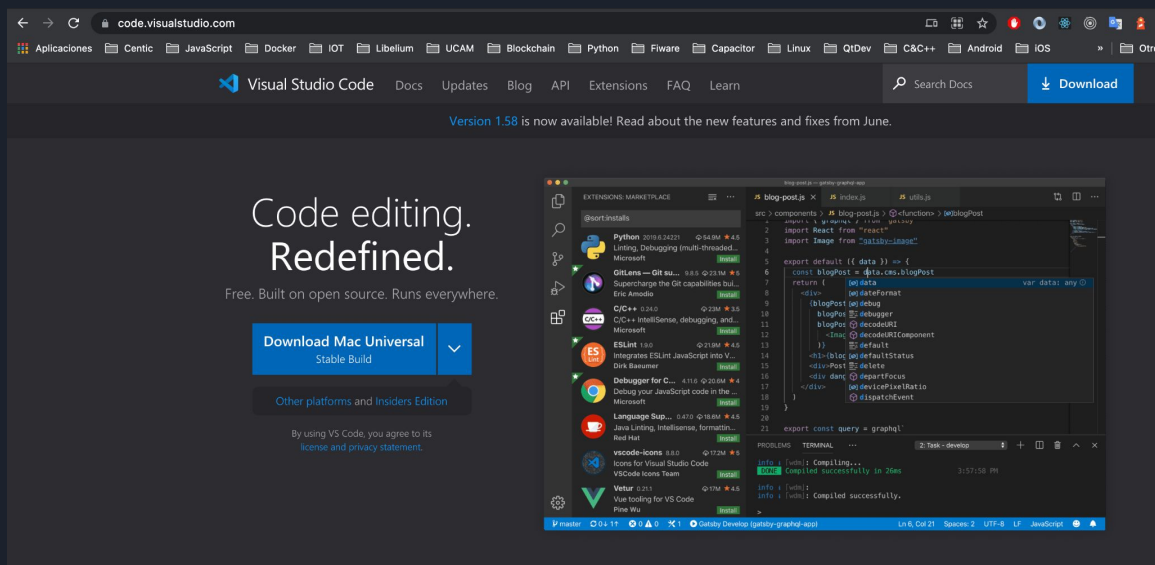


---

# Entorno de desarrollo

# Visual Studio Code

## Descargar Visual Studio Code



The image shows the Visual Studio Code website and a preview of the IDE interface. The website header includes the Visual Studio Code logo, navigation links (Docs, Updates, Blog, API, Extensions, FAQ, Learn), a search bar, and a 'Download' button. A banner below the header states 'Version 1.58 is now available! Read about the new features and fixes from June.' The main content area features the text 'Code editing. Redefined.' and 'Free. Built on open source. Runs everywhere.' Below this is a 'Download Mac Universal Stable Build' button and a link to 'Other platforms and Insiders Edition'. A note at the bottom of the main content area states 'By using VS Code, you agree to its license and privacy statement.' The right side of the image shows a preview of the Visual Studio Code IDE interface, displaying the 'EXTENSIONS MARKETPLACE' on the left, a code editor with a JavaScript file named 'blog-post.js', and a 'TERMINAL' window at the bottom showing compilation output.

code.visualstudio.com

Visual Studio Code Docs Updates Blog API Extensions FAQ Learn

Search Docs Download

Version 1.58 is now available! Read about the new features and fixes from June.

Code editing.  
Redefined.

Free. Built on open source. Runs everywhere.

Download Mac Universal  
Stable Build

Other platforms and Insiders Edition

By using VS Code, you agree to its  
license and privacy statement.

EXTENSIONS MARKETPLACE

Python 2018.8.24221 4.6.0 4.6.0  
Linting, Debugging (multi-threaded...  
Microsoft

GitLens — Git superpowers 1.8.0 4.6.0 4.6.0  
Supercharge the Git capabilities bu...  
Eric Amadio

C/C++ 0.24.0 4.6.0 4.6.0  
C/C++ IntelliSense, debugging, and...  
Microsoft

ESLint 1.8.0 4.6.0 4.6.0  
Integrates ESLint JavaScript into V...  
Orin Bassman

Debugger for C++ 4.11.6 4.6.0 4.6.0  
Debug your JavaScript code in the ...  
Microsoft

Language Supp... 0.20.0 4.6.0 4.6.0  
Java Linting, IntelliSense, formatin...  
Red Hat

vscode-icons 3.9.0 4.6.0 4.6.0  
Icons for Visual Studio Code  
vscode-icons Team

Vetur 0.20.1 4.6.0 4.6.0  
Vue tooling for VS Code  
Pine Hu

blog-post.js index.js utils.js

```
src > components > A blog-post.js > @-function > @blogPost
import React from "react"
import Image from "gatsby-image"

export default ({ data }) => {
  const blogPost = data.cms.blogPost
  return (
    <div>
      <h1>@dataformat
      <h2>@blogPost
      <h3>@blogPost
      <h4>@blogPost
      <h5>@blogPost
      <h6>@blogPost
      <h7>@blogPost
      <h8>@blogPost
      <h9>@blogPost
      <h10>@blogPost
      <h11>@blogPost
      <h12>@blogPost
      <h13>@blogPost
      <h14>@blogPost
      <h15>@blogPost
      <h16>@blogPost
      <h17>@blogPost
      <h18>@blogPost
      <h19>@blogPost
      <h20>@blogPost
      <h21>@blogPost
      <h22>@blogPost
      <h23>@blogPost
      <h24>@blogPost
      <h25>@blogPost
      <h26>@blogPost
      <h27>@blogPost
      <h28>@blogPost
      <h29>@blogPost
      <h30>@blogPost
      <h31>@blogPost
      <h32>@blogPost
      <h33>@blogPost
      <h34>@blogPost
      <h35>@blogPost
      <h36>@blogPost
      <h37>@blogPost
      <h38>@blogPost
      <h39>@blogPost
      <h40>@blogPost
      <h41>@blogPost
      <h42>@blogPost
      <h43>@blogPost
      <h44>@blogPost
      <h45>@blogPost
      <h46>@blogPost
      <h47>@blogPost
      <h48>@blogPost
      <h49>@blogPost
      <h50>@blogPost
      <h51>@blogPost
      <h52>@blogPost
      <h53>@blogPost
      <h54>@blogPost
      <h55>@blogPost
      <h56>@blogPost
      <h57>@blogPost
      <h58>@blogPost
      <h59>@blogPost
      <h60>@blogPost
      <h61>@blogPost
      <h62>@blogPost
      <h63>@blogPost
      <h64>@blogPost
      <h65>@blogPost
      <h66>@blogPost
      <h67>@blogPost
      <h68>@blogPost
      <h69>@blogPost
      <h70>@blogPost
      <h71>@blogPost
      <h72>@blogPost
      <h73>@blogPost
      <h74>@blogPost
      <h75>@blogPost
      <h76>@blogPost
      <h77>@blogPost
      <h78>@blogPost
      <h79>@blogPost
      <h80>@blogPost
      <h81>@blogPost
      <h82>@blogPost
      <h83>@blogPost
      <h84>@blogPost
      <h85>@blogPost
      <h86>@blogPost
      <h87>@blogPost
      <h88>@blogPost
      <h89>@blogPost
      <h90>@blogPost
      <h91>@blogPost
      <h92>@blogPost
      <h93>@blogPost
      <h94>@blogPost
      <h95>@blogPost
      <h96>@blogPost
      <h97>@blogPost
      <h98>@blogPost
      <h99>@blogPost
      <h100>@blogPost
    </div>
  )
}
```

PROBLEMS TERMINAL

Task: develop

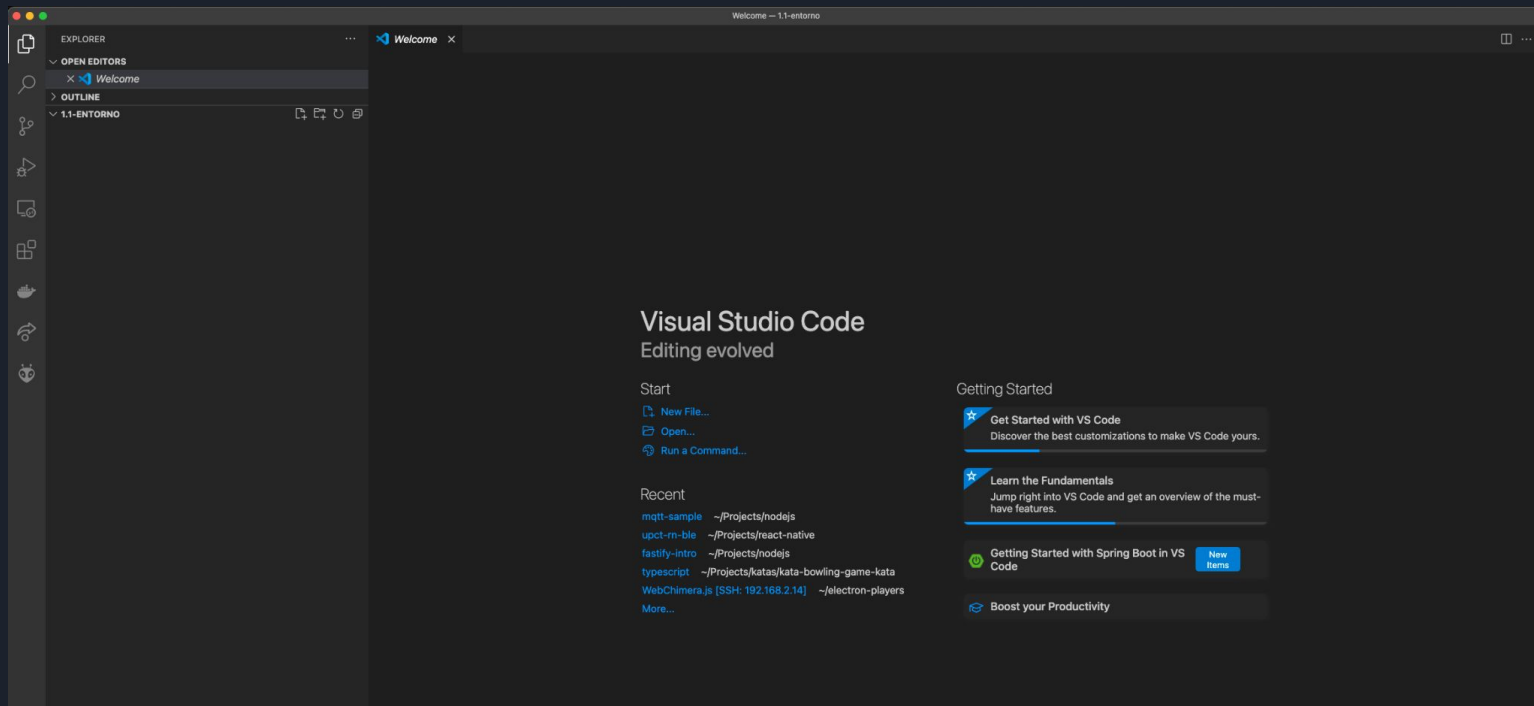
3:57:58 PM

Ln 8, Col 21 SyntaxError: UTS-8 LF JavaScript

<https://code.visualstudio.com/>

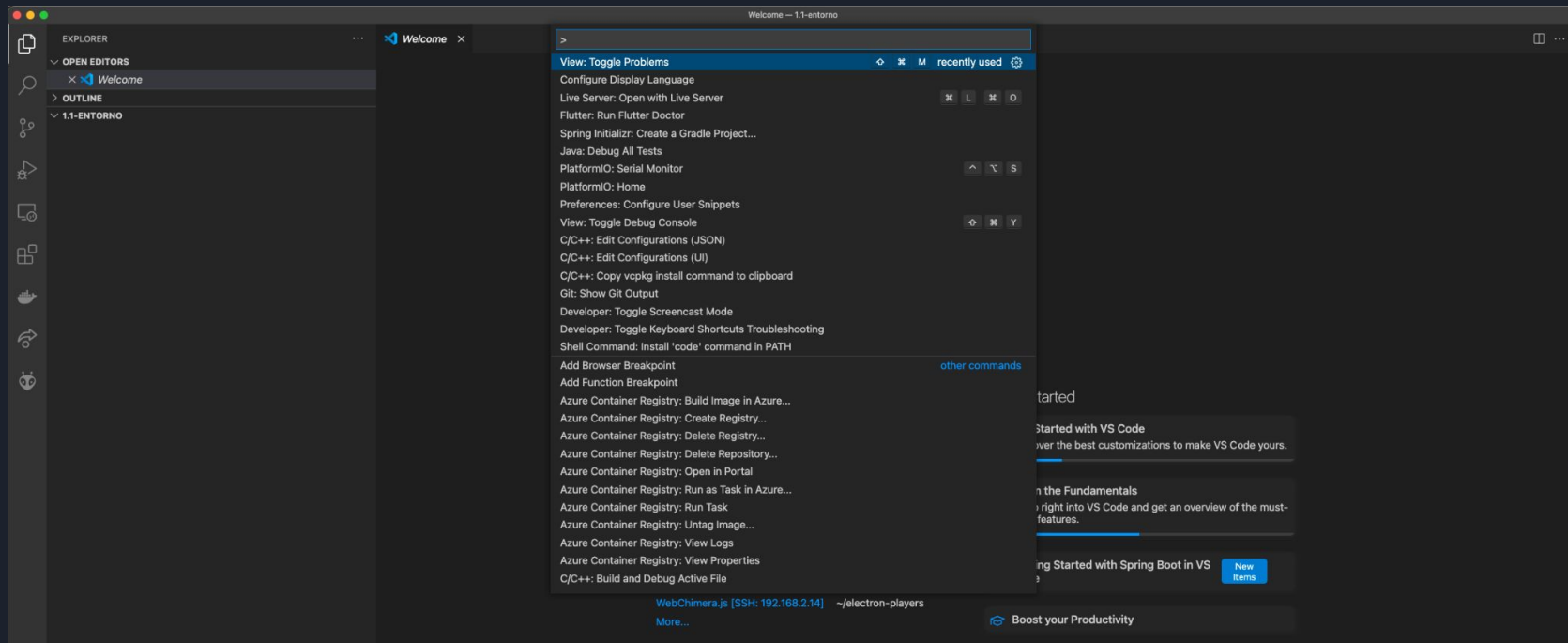
# Visual Studio Code

## Entorno de Visual Code



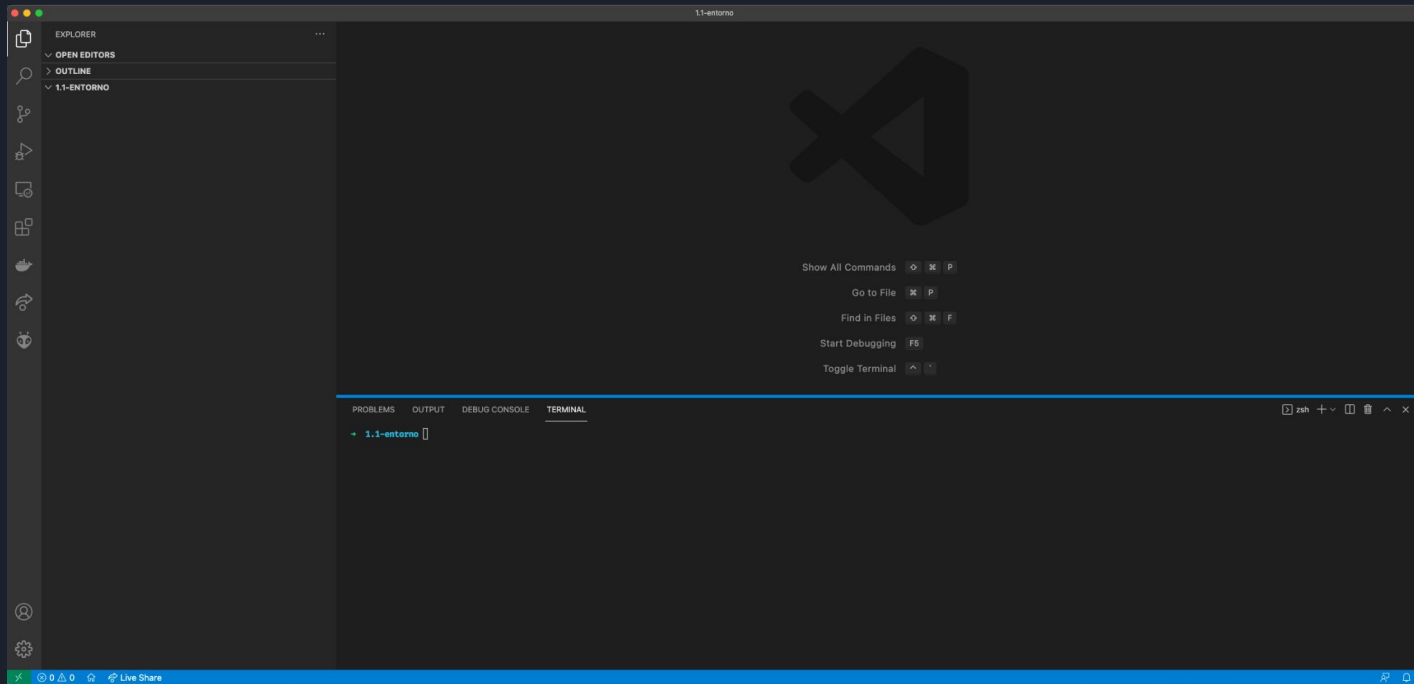
# Visual Studio Code

Ejecutar comandos. Ctrl/Cmd + Shift + P



# Visual Studio Code

## Terminal integrada



# Visual Studio Code

## Plugins

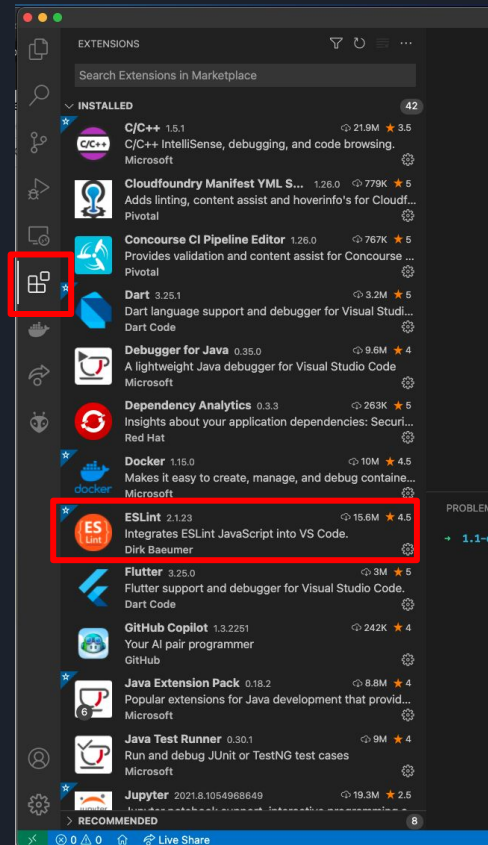
### ESLint

1. Analizador estático de código
2. Corrección de problemas de código
3. Personalización de reglas

Añadir reglas en “**eslintrc.json**”

```
.eslintrc — 1.1-entorno

{} .eslintrc x JS main.js
{} .eslintrc > ...
1 {
2   "rules": {
3     "nombre de la regla": ["error|warn|off", "parametro de la regla"],
4     "semi": ["error", "always"],
5     "quotes": ["error", "single"]
6   }
7 }
8
```



# Visual Studio Code

## Plugins

**Prettier:** Formateador de código

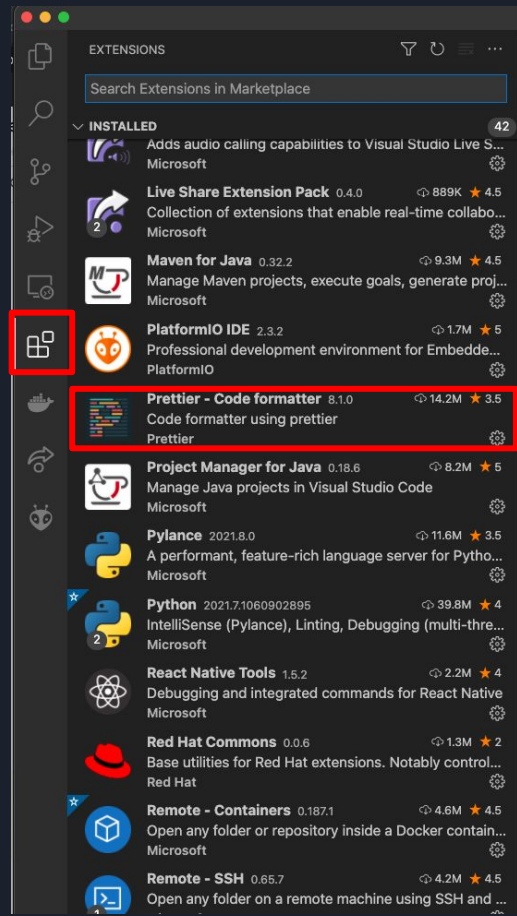
Añadir reglas en “**.prettierrc.json**”

Formatear al guardar

```
{ } .prettierrc ×  
{ } .prettierrc > ...  
1  
2 "arrowParens": "avoid",  
3 "bracketSpacing": true,  
4 "embeddedLanguageFormatting": "auto",  
5 "htmlWhitespaceSensitivity": "css",  
6 "insertPragma": false,  
7 "jsxBracketSameLine": false,  
8 "jsxSingleQuote": true,  
9 "printWidth": 80,  
10 "proseWrap": "preserve",  
11 "quoteProps": "as-needed",  
12 "requirePragma": false,  
13 "semi": true,  
14 "singleQuote": true,  
15 "tabWidth": 4,  
16 "trailingComma": "none",  
17 "useTabs": false,  
18 "vueIndentScriptAndStyle": false  
19
```

```
{ } settings.json ×  
Users > juanjo > Library > Application Support > Code >  
54 "dart.lineLength": 160,  
55 "redhat.telemetry.enabled": true,  
56 "window.zoomLevel": 1,  
57 "editor.formatOnSave": true  
58
```

```
≡ Settings { } settings.json ×  
Users > juanjo > Library > Application Support > Code > User > { } settings.json > ...  
5 "editor.defaultFormatter": "esbenp.prettier-vscode",  
6 "[javascript]": {  
7 "editor.defaultFormatter": "esbenp.prettier-vscode"  
8 },  
9 "[typescript]": { "editor.defaultFormatter": "esbenp.prettier-vscode" },  
10 "[typescriptreact]": {  
11 "editor.defaultFormatter": "esbenp.prettier-vscode"  
12 },
```

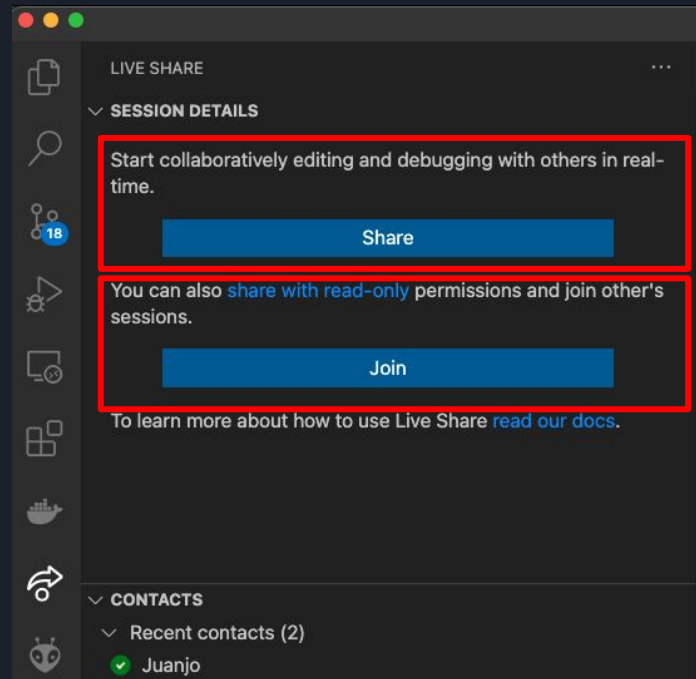
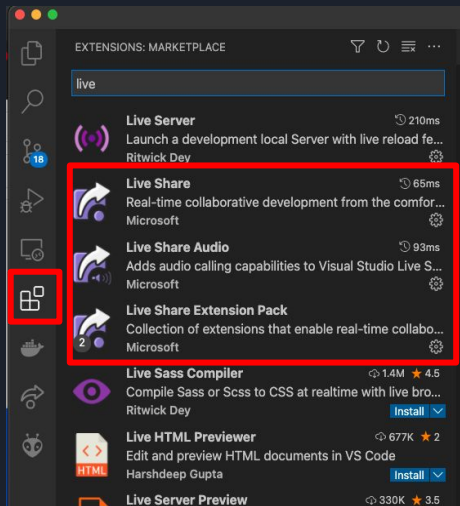


# Visual Studio Code

## Plugins

**Live Share:** Compartir entorno de trabajo

Podemos compartir o unirnos a una sesión (Share / Join)

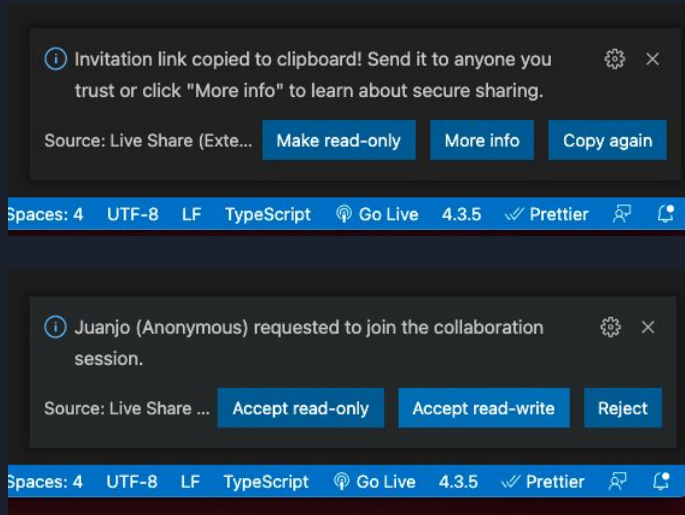




# Visual Studio Code

## Plugins

**Live Share:** Compartir entorno de trabajo



```
function createWindow() {  
  mainWindow = new BrowserWindow({  
    width: 800,  
    height: 600,  
    webPreferences: { Juanjo (Read-write)  
      preload: path.join(__dirname, './nodejs/index.js'  
    }  
  });  
  mainWindow.loadURL(path.join('file://', __dirname, 'inde
```

- Instalar Visual Code
- Instalar y configurar plugin ESLINT
- Instalar y configurar plugin PRETTIER
- Instalar plugin Live Share
- Compartir una sesión con Live Share
- Unirse a una sesión de Live Share

# NodeJS

Descargar e instalar Node.js

## Linux y Mac: NVM

<https://github.com/nvm-sh/nvm>

```
→ 1.1-entorno nvm list
    v14.17.2
→    v16.2.0
    v16.3.0
default → v16.2.0
iojs → N/A (default)
unstable → N/A (default)
node → stable (→ v16.3.0) (default)
stable → 16.3 (→ v16.3.0) (default)
lts/* → lts/fermium (→ v14.17.2)
lts/argon → v4.9.1 (→ N/A)
lts/boron → v6.17.1 (→ N/A)
lts/carbon → v8.17.0 (→ N/A)
lts/dubnium → v10.24.1 (→ N/A)
lts/erbium → v12.22.2 (→ N/A)
lts/fermium → v14.17.2
→ 1.1-entorno
```

## Windows: NVM-WINDOWS

<https://github.com/nvm-sh/nvm>

```
Windows PowerShell
PS C:\Users\HP> nvm list

    16.6.1
    14.17.2
    12.22.2

PS C:\Users\HP> nvm use 14.17.2
Now using node v14.17.2 (64-bit)
PS C:\Users\HP> node --version
v14.17.2
PS C:\Users\HP>
```

# NodeJS

## ## Comandos

Listar todas las versiones de disponibles para descargar e instalar:

```
```sh
nvm ls-remote
```
```

Instalar la última versión de Node.js:

```
```sh
nvm install node # "node" es un alias a la última versión
```
```

Instalar una versión específica de Node.js:

```
```sh
nvm install 6.14.4 # o 10.10.0, 8.9.1, etc
```
```

# NVM

La primera versión instalada se convierte en la versión por defecto (`nvm alias default`).

Cambiar la versión por defecto:

```
```sh
nvm alias default 16.0.2
```
```

Listar las versiones de Node.js instaladas:

```
```sh
nvm list
```
```

Cambiar una versión de Node.js concreta:

```
```sh
nvm use 16.2.0 # node para cambiar a la última versión
node --version
```
```



# NodeJS

## NVM-WINDOWS

### Comandos

Comprobar la arquitectura del equipo, 32 o 64:

```
nvm arch
```

Listar todas las versiones de disponibles para descargar e instalar:

```
nvm list available [32|64|all]
```

Instalar la última versión de Node.js:

```
nvm install latest [arch] # "latest" es un alias a la última versión
```

Instalar una versión específica de Node.js:

```
nvm install 6.14.4 [arch] # o 10.10.0, 8.9.1, etc
```

Listar las versiones de Node.js instaladas:

```
nvm list
```

Cambiar una versión de Node.js concreta:

```
nvm use 16.2.0 [] # node para cambiar a la última versión  
node --version
```

Desinstalar una versión en concreto:

```
nvm uninstall 16.0.2
```

- Instalar NVM / NVM-WINDOWS
- Instalar las dos últimas versiones LTS
- Instalar la última versión
- Mostrar la última versión
- Desinstalar la versión LTS más antigua

## Conceptos básicos

- Es un entorno de ejecución de JavaScript cross-platform y open-source
- Usa el motor de JS de Google (V8)
- Se ejecuta en un sólo proceso
- Su API está diseñada para usarse de forma asíncrona
- Implementa el estándar ECMAScript

JS main.js X

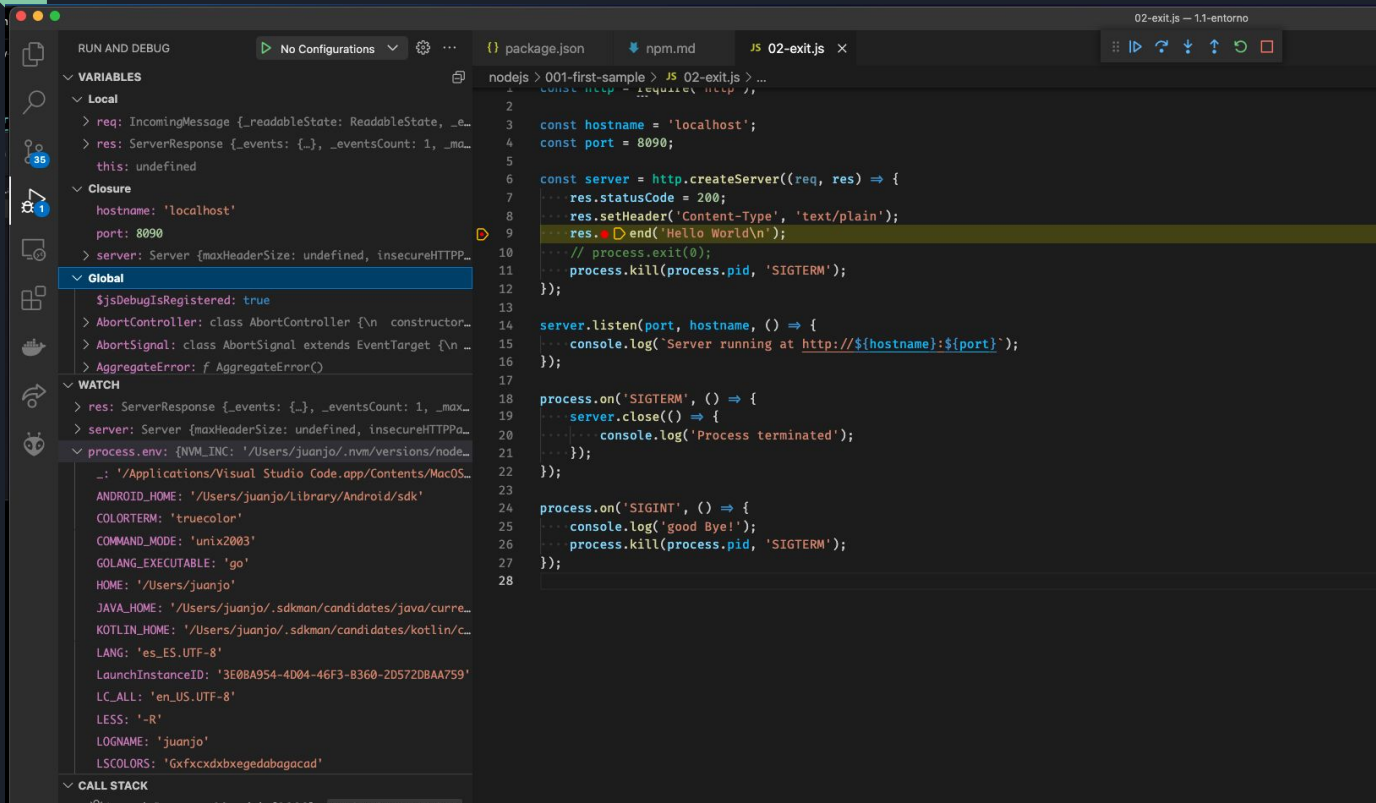
JS main.js > [?] server > http.createServer() callback

```
1  const http = require('http');
2
3  const hostname = 'localhost';
4  const port = 9090;
5
6  const server = http.createServer((req, res) => {
7    res.statusCode = 200;
8    res.setHeader('Content-Type', 'text/plain');
9    res.end('Hello World\n');
10 });
11
12 server.listen(port, hostname, () => {
13   console.log(`Server running at http://${hostname}:${port}`);
14 });
15
```



# NodeJS

## Conceptos básicos



The image shows a screenshot of the Visual Studio Code editor with a Node.js application running in the background. The editor is displaying a file named `02-exit.js` with the following code:

```
1 const http = require('http');
2
3 const hostname = 'localhost';
4 const port = 8090;
5
6 const server = http.createServer((req, res) => {
7   res.statusCode = 200;
8   res.setHeader('Content-Type', 'text/plain');
9   res.end('Hello World\n');
10  // process.exit(0);
11  process.kill(process.pid, 'SIGTERM');
12 });
13
14 server.listen(port, hostname, () => {
15   console.log(`Server running at http://${hostname}:${port}`);
16 });
17
18 process.on('SIGTERM', () => {
19   server.close() => {
20     console.log('Process terminated');
21   });
22 });
23
24 process.on('SIGINT', () => {
25   console.log('good Bye!');
26   process.kill(process.pid, 'SIGTERM');
27 });
28
```

The left sidebar shows the **VARIABLES** pane, which is currently displaying the **Global** scope. The variables listed are:

- `$jsDebugIsRegistered`: true
- `AbortController`: class AbortController { \n constructor...
- `AbortSignal`: class AbortSignal extends EventTarget { \n ...
- `AggregateError`: f AggregateError()

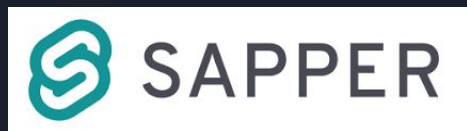
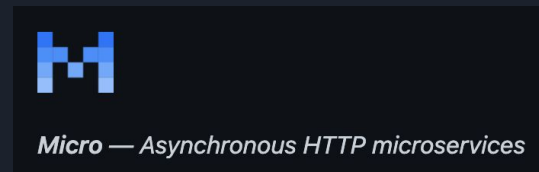
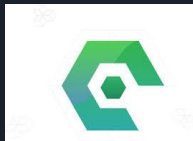
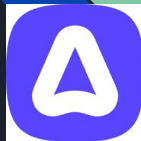
The **WATCH** pane is also visible, showing the following variables:

- `res`: ServerResponse { \_events: {..}, \_eventsCount: 1, \_max...
- `server`: Server { maxHeaderSize: undefined, insecureHTTP...
- `process.env`: { NVM\_INC: '/Users/juanjo/.nvm/versions/node...

The **CALL STACK** pane is currently empty.

# NodeJS

## Conceptos básicos



## Ejecutar aplicaciones

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

```
→ first-sample node main.js  
Server running at http://localhost:9090  
█
```

PROBLEMS    OUTPUT    DEBUG CONSOLE    TERMINAL

```
→ first-sample curl localhost:9090  
Hello Worldn  
→ first-sample █
```

## Cerrar aplicaciones

- Ctrl + c : Fuerza bruta
- `process.exit(status)` : Cierre con información
- `process.exitCode = status` : Información de cierre
- Usar Señales POSIX. SIGKILL & SIGTERM

```
process.kill(process.pid, 'SIGTERM');
process.on('SIGTERM', () => {
  ... server.close(() => {
  ... | ... console.log('Process terminated');
  ... | ... });
});
```

# NodeJS

## Variables de Entorno

- Se almacenan en process.env
- Podemos exportar variables
- Podemos usar herramientas como **dotenv**

```
require('dotenv').config();

console.log(
  ... 'Use info:',
  ... process.env.USER_ID,
  ... process.env.USER_KEY
);
```

```
→ 002-env-variables node dot-env.js
Use info: 239482 foobar
```

```
1 # .env file
2 USER_ID="239482"
3 USER_KEY="foobar"
```

```
→ 002-env-variables node -r dotenv/config main.js
Use info: 239482 foobar
```

```
TERM: 'xterm-256color',
SHELL: '/bin/zsh',
TMPDIR: '/var/folders/6t/58yglkcs47g2zp_hy_dw6l580000gn/T/',
TERM_PROGRAM_VERSION: '1.59.1',
ORIGINAL_XDG_CURRENT_DESKTOP: 'undefined',
SDKMAN_PLATFORM: 'darwinx64',
ZSH: '/Users/juanjo/.oh-my-zsh',
NVM_DIR: '/Users/juanjo/.nvm',
USER: 'juanjo',
COMMAND_MODE: 'unix2003',
```

```
console.log('Env:', process.env);
console.log(
  ... 'Use info:',
  ... process.env.USER_ID,
  ... process.env.USER_KEY);
```

```
002-env-variables > load-app.sh
1 USER_ID=123 USER_KEY=SuperSecretKey node main.js
```

```
→ 002-env-variables ./load-app.sh
Use info: 123 SuperSecretKey
```

# NodeJS

## REPL

Read Evaluate Print Loop

- Línea de comandos de Node.js

```
→ nodejs node
Welcome to Node.js v16.6.1.
Type ".help" for more information.
> .help
.break      Sometimes you get stuck, this gets you out
.clear      Alias for .break
.editor      Enter editor mode
.exit        Exit the REPL
.help        Print this help message
.load        Load JS from a file into the REPL session
.save        Save all evaluated commands in this REPL session to a file

Press Ctrl+C to abort current expression, Ctrl+D to exit the REPL
> █
```

```
> Array.from({length: 10}, (value, index) => index + 1)
[
  1, 2, 3, 4, 5,
  6, 7, 8, 9, 10
]
> _.map(i => i * i)
[
  1, 4, 9, 16, 25,
  36, 49, 64, 81, 100
]
> _.reverse()
[
  100, 81, 64, 49, 36,
  25, 16, 9, 4, 1
]
█
```

# NodeJS

## REPL

Read Evaluate Print Loop

- Explorar objetos

```
> global.  
global.__proto__      global.hasOwnProperty  global.isPrototypeOf  global.propertyIsEnumerable  global.toLocaleString  global.toString      global.valueOf  
  
global.constructor  
  
global.AbortController  global.AbortSignal      global.AggregateError  global.Array               global.ArrayBuffer      global.Atomics        global.BigInt  
global.BigInt64Array    global.BigUint64Array   global.Boolean         global.Buffer             global.DataView         global.Date           global.Error  
global.EvalError        global.Event            global.EventTarget     global.EventTarget        global.Float32Array     global.Float64Array   global.Function  
global.Infinity         global.Int16Array       global.Int32Array      global.Int8Array          global.Intl            global.JSON           global.Map  
global.Math             global.MessageChannel   global.MessageEvent    global.MessagePort        global.NaN              global.Number         global.Object  
global.Promise          global.Proxy            global.RangeError      global.Reflect            global.RegExp           global.Set            global.Object  
global.ShareArrayBuffer global.String            global.Symbol           global.ReferenceError     global.TextDecoder      global.SetError       global.TypeError  
global.URLError         global.URL              global.Symbol           global.SyntaxError        global.TextEncoder      global.Uint8Array     global.Uint8ClampedArray  
global.WeakMap          global.WeakRef          global.Symbol           global.Uint16Array        global.Uint32Array      global.TypeError     global.assert  
global.async_hooks      global.atob             global.Symbol           global.WebAssembly        global._error           global.clearImmediate global.clearInterval  
global.clearTimeout     global.atob             global.btoa             global.buffer              global.decodeURI        global.decodeURIComponent  
global.dgram            global.atob             global.console          global.constants          global.encodeURI        global.encodeURIComponent  
global.eval             global.atob             global.dns              global.domain             global.global           global.escape  
global.https            global.atob             global.fs               global.global             global.globalThis       global.http           global.http2  
global.parseFloat       global.atob             global.isFinite         global.isNaN              global.module           global.os              global.os  
global.parseInt         global.atob             global.path             global.perf_hooks         global.performance      global.punycode       global.setInterval  
global.querystring      global.atob             global.readline         global.repl               global.require           global.setInterval    global.setInterval  
global.setTimeout       global.atob             global.string_decoder   global.sys                global.timers            global.trace_events   global.trace_events  
global.tty              global.atob             global.unescape         global.url                global.util              global.v8              global.v8  
global.wasi             global.atob             global.zlib
```

# NodeJS

## Argumentos

- Entrada de valores desde la línea de comandos.  
**process.argv**

```
003-argumentos > JS main.js
```

```
1   console.log('Args:', process.argv);  
2
```

```
→ 003-argumentos node main.js userId=123
```

```
Args: [
```

```
  '/Users/juanjo/.nvm/versions/node/v16.6.1/bin/node',
```

```
  '/Users/juanjo/Projects/teamcamp/1.1-entorno/nodejs/003-argumentos/main.js',
```

```
  'userId=123'
```

```
]
```

```
→ 003-argumentos █
```

- Mejor usar herramientas cómo **minimist**

```
const args = require('minimist')(process.argv.slice(2))  
args['name'] //joe
```

```
// node main.js --name=joe
```

```
$ node example/parse.js -x 3 -y 4 -n5 -abc --beep=boop foo bar baz  
{ _: [ 'foo', 'bar', 'baz' ],  
  x: 3,  
  y: 4,  
  n: 5,  
  a: true,  
  b: true,  
  c: true,  
  beep: 'boop' }
```



```
-> Path to find files: ./statics
```

```
-----  
[ X ] fingers.json: ..... 0  
[ X ] ibuttons.json: ..... 0  
[ X ] templates.md: ..... 0  
-----
```

```
Files: 3          Parsed: 0          Failed: 3  
Fingers: 0       Uniques: 0         Overwrites: 0  
-----
```

```
Output:  
-----
```

## Salida por consola

- Mostrar texto por consola (process.stdout): **console.log()**
- Múltiples variables: **console.log('hello', x, y, z)**
- Texto formateado: **console.log('My %s has %d years', 'cat', 2)**
  - %s** === string
  - %d** === number
  - %i** === integer
  - %o** === object
- Limpiar la consola: **console.clear()**
- Mostrar la traza de la llamada. **console.trace()**
- Calcular incrementos de tiempo. **console.time()** / **console.timeEnd()**
- Módulo “**chalk**”. Formatear la salida por consola
- Salida de errores (process.stderr): **console.error()**



# NodeJS

## Entrada por consola

- módulo readline (`process.stdin`)
- módulo “**`inquirer`**”

## Módulos (CommonJS)

- Importar módulos: “**require**”
  1. Rutas relativas/absolutas para ficheros, nombres de módulos para módulos contenidos dentro de `node_modules`.
  2. Elementos no exportados quedan privados.
  3. El primer `require` añade a una caché el contenido del módulo usando su ruta como clave
- Exportar módulos: “**module.exports**”

Hay dos formas de exportar código:

  1. Exportar un módulo completo: `module.exports = {}`
  2. Exportar contenido con nombre: `exports.name = {}`
- Podemos obtener la ruta del módulo con `__dirname`
- **Tenemos** que exportar el código de forma **asíncrona**.

- Crear la siguiente estructura de proyecto
  1. módulo lib1.js exporta la función showMyPath que imprime su ruta.
  2. módulo lib2.js exporta el objeto User, que tiene nombre y edad.
  3. módulo principal (main.js) importa ambos módulos y ejecuta showMyPath e imprime el objeto Users



# NodeJS

## Node Package Manager

- NPM es el gestor de paquetes estándar de Node.js
- Yarn y PNPM son alternativas a NPM
- Crear un proyecto: **npm init**
- Instalar las dependencias del proyecto: **npm install**
- Instalar una dependencia: **npm install package-name**
  - save-dev**
  - no-save**
- Actualizar dependencias: **npm update**
- Actualizar una dependencia: **npm update package-name**
- Restricciones de versionado: **SemVer**
- Ejecutar tareas: **npm run task**



# NodeJS

## Node Package Manager

### **Instalación:**

Local: `npm install package`. Instala la dependencia en el directorio `node_modules` del proyecto

Global: `npm install package -g`. Instala la dependencia en el directorio `node_modules` de la instalación de Node.js

### **Rutas de `node_modules`:**

`npm root`

`npm root -g`

### **Ejecutar paquetes de `node_modules`:**

`./node_modules/.bin/package`

`npx package`



# NodeJS

## Node Package Manager

**package.json:** Es el manifest de nuestra aplicación.

**name:** Nombre de la aplicación / paquete

**version:** Indica la versión actual

**description:** Breve descripción de la aplicación / paquete

**main:** Punto de entrada para la aplicación

**scripts:** Conjunto de scripts que node.js que puede ejecutar

**dependencies:** Lista de paquetes npm instalados como dependencias

**devDependencies:** establece una lista de paquetes npm instalados como dependencias de desarrollo

**engines:** Versiones de Node.js donde funciona este paquete / aplicación

**browsersList:** Indica qué navegadores (y sus versiones) admite.

```
{
  "name": "projectname",
  "version": "0.0.1",
  "description": "Descripción del proyecto",
  "main": "build/main.js",
  "scripts": {
    "build": "tsc -p tsconfig-build.json",
    "start": "node build/main.js",
    "test": "jest --watchAll"
  },
  "author": "juanjojp",
  "repository": "",
  "license": "MIT",
  "devDependencies": {
    "codecov": "^3.8.3",
    "eslint": "^7.31.0",
    "eslint-config-prettier": "^8.3.0",
    "eslint-plugin-filenames": "^1.3.2",
    "husky": "^7.0.1",
    "jest": "^27.0.6",
    "jest-watch-typeahead": "^0.6.4"
  },
  "dependencies": {
    "convict": "^6.2.0",
    "fastify": "^3.19.1",
    "make-promises-safe": "^5.1.0",
    "mongodb": "^4.0.1",
    "pino": "^6.12.0"
  },
  "engines": {
    "node": "≥ 6.0.0",
    "npm": "≥ 3.0.0"
  },
  "browserslist": ["> 1%", "last 2 versions", "not ie ≤ 8"]
}
```





# NodeJS

## Node Package Manager

### package-lock.json:

Se introduce a partir de la versión 5 de npm.

Se encarga de mantener controladas las versiones instaladas

Garantiza que nuevas instalaciones usen las mismas versiones de dependencias

Se actualiza con npm update

Debemos añadirlo al gestor de código

```
{
  "name": "nodejs",
  "lockfileVersion": 2,
  "requires": true,
  "packages": {
    "": {
      "dependencies": {
        "cowsay": "^1.5.0"
      },
      "devDependencies": {
        "chalk": "^4.1.2",
        "inquirer": "^8.1.2"
      }
    },
    "node_modules/ansi-escapes": {
      "version": "4.3.2",
      "resolved": "https://registry.npmjs.org/ansi-escapes/-/ansi-escapes-4.3.2.tgz",
      "integrity": "sha512-gKXj5ALrKWQlsYG9jlTRmR/xKLuxHV+Z9QEwNIgCfM1/uwPMCuzVVnh5mw",
      "dev": true,
      "dependencies": {
        "type-fest": "^0.21.3"
      },
      "engines": {
        "node": "≥ 8"
      },
      "funding": {
        "url": "https://github.com/sponsors/sindresorhus"
      }
    },
    "node_modules/ansi-regex": {
      "version": "5.0.0",
      "resolved": "https://registry.npmjs.org/ansi-regex/-/ansi-regex-5.0.0.tgz",
      "integrity": "sha512-bY6jB6887830616p0H4GvYIv3111y+vZC+7C5hDzU43YYWIIWg",
      "engines": {
        "node": "≥ 8"
      }
    },
    "node_modules/ansi-styles": {
      "version": "4.3.0",
      "resolved": "https://registry.npmjs.org/ansi-styles/-/ansi-styles-4.3.0.tgz",
      "integrity": "sha512-BR38U1Wp1dIN3EKyHdU0H13CSj5Q2yFwN51Y1Xz4G3VZLHhK",
      "dependencies": {
        "color-convert": "^2.0.1"
      },
      "engines": {
        "node": "≥ 8"
      },
      "funding": {
        "url": "https://github.com/chalk/ansi-styles?sponsor=1"
      }
    },
  }
}
```

# NodeJS

## Node Package Manager

**npm list:** muestra el listado de dependencias instaladas y su versión

-g: dependencias globales

--depth=X: Dependencias hasta el nivel X

```
→ cep git:(master) * npm list -g
/Users/juanjo/.nvm/versions/node/v16.6.1/lib
├─ npm-check-updates@11.8.3
└─ npm@7.20.3
```

```
→ cep git:(master) * npm list --depth=2
tribeca-cep@0.0.1 /Users/juanjo/Projects/3bk/cep
├─ @types/agenda@2.0.9
├─ @types/mongodb@3.5.27 deduped
├─ @types/node@14.11.2 deduped
├─ @types/convict@5.2.1
├─ @types/node@14.11.2 deduped
├─ @types/jest@26.0.14
├─ jest-diff@25.5.0
├─ chalk@3.0.0
├─ diff-sequences@25.2.6
├─ jest-get-type@25.2.6
├─ pretty-format@25.5.0 deduped
├─ pretty-format@25.5.0
├─ @jest/types@25.5.0
├─ ansi-regex@5.0.0
├─ ansi-styles@4.2.1
├─ react-is@16.12.0
├─ @types/mongodb@3.5.27
├─ @types/bson@4.0.2
├─ @types/node@14.11.2 deduped
├─ @types/node@14.11.2 deduped
├─ @types/node-fetch@2.5.7
├─ @types/node@14.11.2 deduped
├─ form-data@3.0.0
├─ async-kit@0.4.0
├─ combined-stream@1.0.8
└─ mime-types@2.1.26
```

```
→ cep git:(master) * npm list
tribeca-cep@0.0.1 /Users/juanjo/Projects/3bk/cep
├─ @types/agenda@2.0.9
├─ @types/convict@5.2.1
├─ @types/jest@26.0.14
├─ @types/mongodb@3.5.27
├─ @types/node-fetch@2.5.7
├─ @types/node@14.11.2
├─ @types/pino@6.3.1
├─ @typescript-eslint/eslint-plugin@2.34.0
├─ @typescript-eslint/parser@2.34.0
├─ agenda@3.1.0
├─ ajv-errors@1.0.1
├─ ajv@6.12.5
├─ codecov@3.7.2
├─ convict@6.0.0
├─ escape-string-regexp@4.0.0
├─ eslint-plugin-filenames@1.3.2
├─ eslint@6.8.0
├─ fastify-bearer-auth@5.0.2
├─ fastify-cors@4.1.0
├─ fastify-metrics@6.0.2
├─ fastify-swagger@3.3.0
├─ fastify@3.3.0
├─ geolib@3.3.1
├─ jest@25.5.4
├─ liquidjs@9.15.0
├─ make-promises-safe@5.1.0
├─ mongodb@3.6.2
├─ nock@13.0.4
├─ node-fetch@2.6.1
├─ nodemon@2.0.4
├─ prom-client@12.0.0
├─ ts-jest@25.5.1
├─ ts-node@9.0.0
└─ typescript@3.9.7
```

# NodeJS

## Node Package Manager

**npm list package-name**

Versión de una dependencia instalada

```
→ cep-ui git:(master) x npm list react
cep-ui@0.1.0 /Users/juanjo/Projects/3bk/cep-ui
├── @material-ui/core@4.11.4
│   ├── @material-ui/styles@4.11.4
│   │   └── react@17.0.2 deduped
│   ├── @material-ui/system@4.11.3
│   │   └── react@17.0.2 deduped
│   ├── @material-ui/utils@4.11.2
│   │   └── react@17.0.2 deduped
│   ├── react-transition-group@4.4.1
│   │   └── react@17.0.2 deduped
│   └── react@17.0.2 deduped
├── @material-ui/icons@4.11.2
│   └── react@17.0.2 deduped
├── @material-ui/lab@4.0.0-alpha.58
│   └── react@17.0.2 deduped
├── @testing-library/react-hooks@7.0.0
│   ├── react-error-boundary@3.1.3
│   │   └── react@17.0.2 deduped
│   └── react@17.0.2 deduped
└── @testing-library/react@11.2.7
    └── react@17.0.2 deduped
```

# NodeJS

## Node Package Manager

### npm view package-name

Muestra información sobre la dependencia

```
+ cep-ui git:(master) x npm view react
```

```
react@17.0.2 | MIT | deps: 2 | versions: 577
```

```
React is a JavaScript library for building user interfaces.
```

```
https://reactjs.org/
```

```
keywords: react
```

```
dist
```

```
.tarball: https://registry.npmjs.org/react/-/react-17.0.2.tgz
```

```
.shasum: d0b5cc516d29eb3eee383f75b62864cfb6800037
```

```
.integrity: sha512-gnhPt75i/dq/z3/6q/0asP78D0u592D5L1pd7M8P+dck6Fu/jJeL6iVVK23fptSUZj8Vjf++7wXASUNclGQcbA==
```

```
.unpackedSize: 291.1 kB
```

```
dependencies:
```

```
loose-envify: ^1.1.0 object-assign: ^4.1.1
```

```
maintainers:
```

```
- sehmarkbage <sebastian@calyptus.eu>
```

```
- gaearon <dan.abramov@gmail.com>
```

```
- acdlite <npm@andrewclark.io>
```

```
- brianvaughn <briandavidvaughn@gmail.com>
```

```
- fb <opensource+npm@fb.com>
```

```
- trueadm <dg@domgan.com>
```

```
- sophiebits <npm@sophiebits.com>
```

```
- lunaruan <lunaris.ruana@gmail.com>
```

```
dist-tags:
```

```
alpha: 18.0.0-alpha-46a0f050a-20210828
```

```
experimental: 0.0.0-experimental-46a0f050a-20210828
```

```
latest: 17.0.2
```

```
next: 18.0.0-alpha-46a0f050a-20210828
```

```
published 5 months ago by gaearon <dan.abramov@gmail.com>
```

# NodeJS

## Node Package Manager

### **npm view package-name version**

Muestra la última versión de la dependencia

### **npm view package-name versions**

Muestra las versiones de la dependencia

```
→ cep-ui git:(master) ✕ npm view react version
17.0.2
```

```
→ cep-ui git:(master) ✕ npm view react versions
[
  '0.0.0-375616788',
  '0.0.0-00d4f95c2',
  '0.0.0-0203b6567',
  '0.0.0-0935a1db3',
  '0.0.0-0c756fb-697f004',
  '0.0.0-0c756fb-f7f79fd',
  '0.0.0-0cf9fc10b',
  '0.0.0-0e4135e8c',
  '0.0.0-0eea57724',
  '0.0.0-1022ee0ec',
  '0.0.0-10cc40018',
  '0.0.0-11565a207',
  '0.0.0-11b07597e',
  '0.0.0-1214b302e',
  '0.0.0-12adaffef',
  '0.0.0-132b72d7b',
  '0.0.0-172e89b4b',
  '0.0.0-1a2d79250',
  '0.0.0-157472624'
]
```

# NodeJS

## Node Package Manager

### npm outdated

Muestra las versiones de las dependencias que están desactualizadas

```
→ cep-ui git:(master) x npm outdated
```

| Package                      | Current        | Wanted         | Latest         | Location                                  | Depended by |
|------------------------------|----------------|----------------|----------------|---|-------------|
| @material-ui/core            | 4.11.4         | 4.12.3         | 4.12.3         | node_modules/@material-ui/core            | cep-ui      |
| @material-ui/lab             | 4.0.0-alpha.58 | 4.0.0-alpha.60 | 4.0.0-alpha.60 | node_modules/@material-ui/lab             | cep-ui      |
| @testing-library/react       | 11.2.7         | 11.2.7         | 12.0.0         | node_modules/@testing-library/react       | cep-ui      |
| @testing-library/react-hooks | 7.0.0          | 7.0.2          | 7.0.2          | node_modules/@testing-library/react-hooks | cep-ui      |
| @testing-library/user-event  | 13.0.16        | 13.2.1         | 13.2.1         | node_modules/@testing-library/user-event  | cep-ui      |
| @types/history               | 4.7.8          | 4.7.9          | 4.7.9          | node_modules/@types/history               | cep-ui      |
| @types/jest                  | 26.0.23        | 26.0.24        | 27.0.1         | node_modules/@types/jest                  | cep-ui      |
| @types/node                  | 15.12.4        | 15.14.9        | 16.7.8         | node_modules/@types/node                  | cep-ui      |
| @types/react                 | 17.0.11        | 17.0.19        | 17.0.19        | node_modules/@types/react                 | cep-ui      |
| @types/react-dom             | 17.0.8         | 17.0.9         | 17.0.9         | node_modules/@types/react-dom             | cep-ui      |
| @types/react-router-dom      | 5.1.7          | 5.1.8          | 5.1.8          | node_modules/@types/react-router-dom      | cep-ui      |
| @types/uuid                  | 8.3.0          | 8.3.1          | 8.3.1          | node_modules/@types/uuid                  | cep-ui      |
| nock                         | 13.1.0         | 13.1.3         | 13.1.3         | node_modules/nock                         | cep-ui      |
| prettier                     | 2.3.1          | 2.3.2          | 2.3.2          | node_modules/prettier                     | cep-ui      |
| query-string                 | 7.0.0          | 7.0.1          | 7.0.1          | node_modules/query-string                 | cep-ui      |
| react-router-dom             | 5.2.0          | 5.2.1          | 5.2.1          | node_modules/react-router-dom             | cep-ui      |
| typescript                   | 4.3.4          | 4.4.2          | 4.4.2          | node_modules/typescript                   | cep-ui      |

```
→ cep-ui git:(master) x
```



# NodeJS

## Node Package Manager

**npm install [-g] <package-name>@<version>**

Instalar una versión concreta de la dependencia

**npm uninstall [-g] [--save] <package-name>**

Eliminar una dependencia

**npm-check-updates**

Facilita la gestión de actualización de dependencias

```
→ cep-ui git:(master) ✕ ncu
Checking /Users/juanjo/Projects/3bk/cep-ui/package.json
[=====] 25/25 100%

@testing-library/react-hooks      ^7.0.0 → ^7.0.2
@testing-library/user-event      ^13.0.16 → ^13.2.1
@types/react-router-dom          ^5.1.7 → ^5.1.8
nock                             ^13.1.0 → ^13.1.3
prettier                         ^2.3.1 → ^2.3.2
@material-ui/core                 ^4.11.4 → ^4.12.3
@material-ui/lab                  ^4.0.0-alpha.58 → ^4.0.0-alpha.60
@testing-library/react            ^11.2.7 → ^12.0.0
@types/history                     ^4.7.8 → ^4.7.9
@types/jest                       ^26.0.23 → ^27.0.1
@types/node                       ^15.12.4 → ^16.7.8
@types/react                      ^17.0.11 → ^17.0.19
@types/react-dom                  ^17.0.8 → ^17.0.9
@types/uuid                       ^8.3.0 → ^8.3.1
query-string                      ^7.0.0 → ^7.0.1
react-router-dom                  ^5.2.0 → ^5.2.1
typescript                        ^4.3.4 → ^4.4.2

Run ncu -u to upgrade package.json
→ cep-ui git:(master) ✕
```

### Comandos básicos de NPM

#### Instalar dependencias

```
npm install [-g] [--save-dev] [--no-save] <package-name>@<version>
```

#### Actualizar dependencias según sus reglas de semver

```
npm update [--no-save]
```

```
npm update package-name
```

#### Ejecutar scripts

```
npm run script-name
```

#### Ruta a node\_modules

```
npm root  
npm root -g
```

### Listado de dependencias

```
npm list [-g] [--depth=N]
```

```
npm list <package-name>
```

### Información sobre dependencias

```
npm view package-name [version]
```

### Dependencias desactualizadas

```
npm outdated
```

### Eliminar una dependencia

```
npm uninstall [-g] [--save] <package-name>
```

### npm-check-updates

```
npm i npm-check-updates  
ncu  
ncu -u  
npm i
```





# NodeJS

## Node Package Manager

### Semantic Versioning (SemVer)

- Se usa en todos los paquetes de npm
- Se compone de 3 dígitos: X.Y.Z
  - X: Versión principal
  - Y: Versión menor
  - Z: Versión del parche
- Reglas de npm update
  - Sin símbolo: Versión exacta
  - Con símbolo: ^, ~, >, >=, <, <=, =, -, ||
  - latest: La última versión disponible

# NodeJS

# Node Package Manager

# npX

Fué lanzado con la versión 5.2

## Permite ejecutar código publicado en NPM

## No es necesario tener el paquete instalado

## Permite ejecutar en Github gist

```
→ npx npx create-react-app my-app
```

Creating a new React app in `/Users/juanjo/Projects/nodejs/npx/my-app`.

Installing packages. This might take a couple of minutes.

Installing `react`, `react-dom`, and `react-scripts` with `cra-template`...

```
(( )) :: idealTree:webpack: sill fetch manifest schema-utils@^2.7.0
```

```
→ npx npx https://gist.github.com/zkat/4bc19503fe9e9309e2bfaa2c58074d32
```

yay gist

→ **npx**

```
→ npx cowsay "Hello Teamcamp"
```

< Hello Teamcamp >

$$\begin{array}{c} \wedge \\ \text{(oo)} \backslash \\ \text{( ) } \backslash \end{array}$$

→ npx

1. Crear un proyecto con npm
2. Añadir el módulo principal con un mensaje de bienvenida
3. Añadir configuración para prettier y eslint
4. Ejecutar prettier
5. Ejecutar eslint
6. Instalar prettier y eslint
7. Instalar nodemon
8. Crear scripts para ejecutar prettier y eslint
9. Crear script para ejecutar el proyecto
10. Crear script para ejecutar el proyecto cada vez que guardamos un cambio
11. Buscar la última versión de React
12. Instalar la versión 16.0.0 de React
13. Fijar las actualizaciones de React por debajo de la 17
14. Actualizar React a la última versión de la release 16

**Añadir al módulo principal las siguientes funciones:**

- 1. Iniciar con un parámetro que sea saludo o despedida**
- 2. Tiene que pedir el nombre de una persona por consola**
- 3. Si ha sido iniciado con saludo, al recibir el nombre de la persona muestra un saludo personalizado en color verde por consola**
- 4. Si ha sido configurado con salida muestra una despedida personalizada en color rojo**
- 5. Muestra un mensaje de despedida al cerrar la aplicación (Ctrl + C)**
- 6. Modificamos el módulo para que si no recibe el parámetro use el preestablecido en el fichero .env**
- 7. Guarda todos los nombres recibidos en un fichero**
- 8. Depurar el módulo principal para ver el contenido del nombre recibido por consola**



# NodeJS

## Ejercicios

### Investigar:

1. Diferencias entre NPM y YARN
2. Alternativas a node-check-update
3. Plugins de Visual code para Javascript
4. Recomendaciones de reglas de ESLint (Por ejemplo AirBnB)