

- Curso Introducción a JavaScript
- Profesora: Sachetti Sofia

Instalación de prompt

Antes de comenzar

Ahora que ya contamos con las herramientas básicas para la carrera, como son **VSCode**, vamos a preparar una última herramienta que nos será de mucha utilidad.

Tené en cuenta que no siempre vamos a entender todo a la primera, en este caso veremos a fondo esta herramienta más adelante, pero es necesario prepararla cuanto antes para poder utilizarla de la mejor manera, ya que nos proveerá de algunas particularidades que nos servirán para aprender mejor.

Nos estamos refiriendo a la herramienta llamada **Prompt Sync**, que cuenta con la característica de permitirnos introducir valores por la consola o terminal, algo que por defecto no es posible en Node (sí en la terminal de un navegador como Google Chrome, por ejemplo).

De esta manera, nuestros programas, aplicaciones y código en general que realicemos en Node a lo largo de la carrera tendrán mayor relevancia y un mejor entendimiento del uso que se le puede dar.

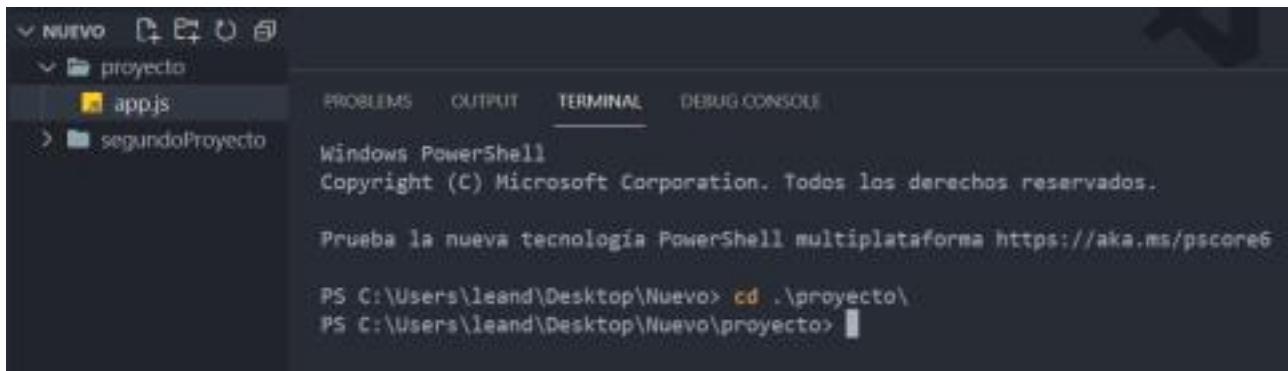
En el siguiente documento encontrarás una guía de cómo poder instalarlo y usarlo correctamente, así como algunos problemas que pueden surgir.

Instalación Prompt Sync

Vamos a realizar una serie de pasos y pruebas para verificar que todo funcione correctamente, o solucionarlo en caso de que no lo haga. No te preocupes si hay cosas que aún no terminas de entender, todo esto será visto a lo largo de esta semana y las que siguen.

1. En primer lugar, debemos instalar el módulo **Prompt Sync**, para eso:
 - a. Abrimos una nueva terminal de **VSCode**.

b. Debemos estar “**parados**” en la carpeta del proyecto en la que trabajaremos (moviéndonos con el comando **cd** seguido del nombre exacto de la carpeta a la que queremos acceder y carpetas previas a ella, en caso de que sea necesario)



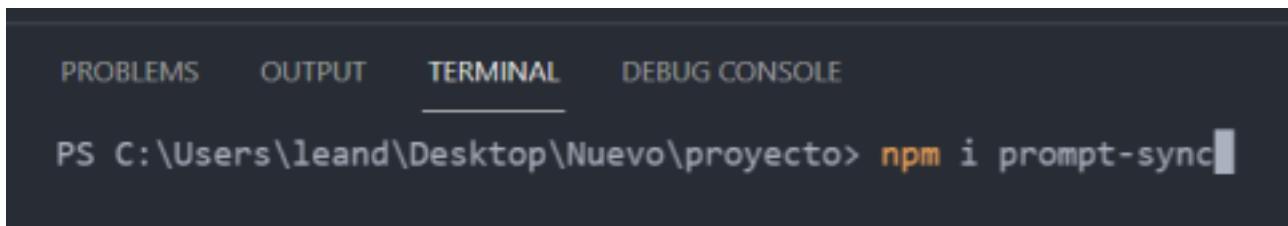
```
Windows PowerShell
Copyright (C) Microsoft Corporation. Todos los derechos reservados.

Prueba la nueva tecnologia PowerShell multiplataforma https://aka.ms/pscore6

PS C:\Users\leand\Desktop\Nuevo> cd ..\proyecto\

PS C:\Users\leand\Desktop\Nuevo\proyecto>
```

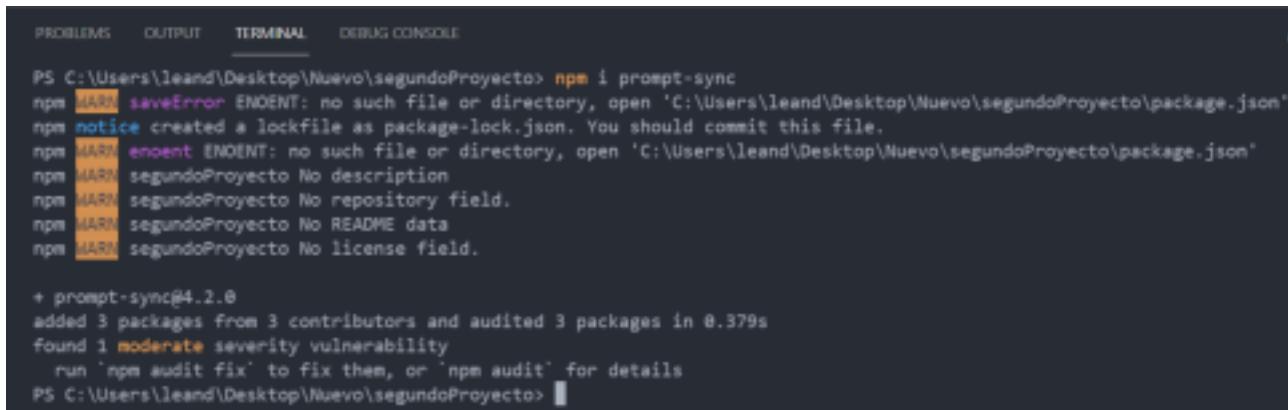
c. En la Consola o Terminal que se abrió escribir el siguiente comando:
npm i prompt-sync



```
PROBLEMS    OUTPUT    TERMINAL    DEBUG CONSOLE

PS C:\Users\leand\Desktop\Nuevo\proyecto> npm i prompt-sync
```

d. Una vez ingresado el comando apretamos Enter y deberá comenzar la instalación, algo así debería verse (o similar)

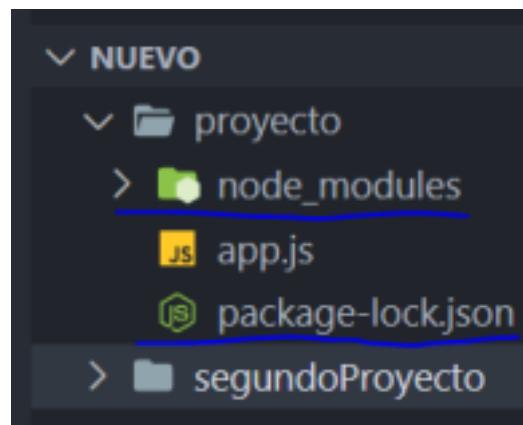


```
PROBLEMS    OUTPUT    TERMINAL    DEBUG CONSOLE

PS C:\Users\leand\Desktop\Nuevo\segundoProyecto> npm i prompt-sync
npm [WARN] saveError ENOENT: no such file or directory, open 'C:\Users\leand\Desktop\Nuevo\segundoProyecto\package.json'
npm notice created a lockfile as package-lock.json. You should commit this file.
npm [WARN] enoent ENOENT: no such file or directory, open 'C:\Users\leand\Desktop\Nuevo\segundoProyecto\package.json'
npm [WARN] segundoProyecto No description
npm [WARN] segundoProyecto No repository field.
npm [WARN] segundoProyecto No README data
npm [WARN] segundoProyecto No license field.

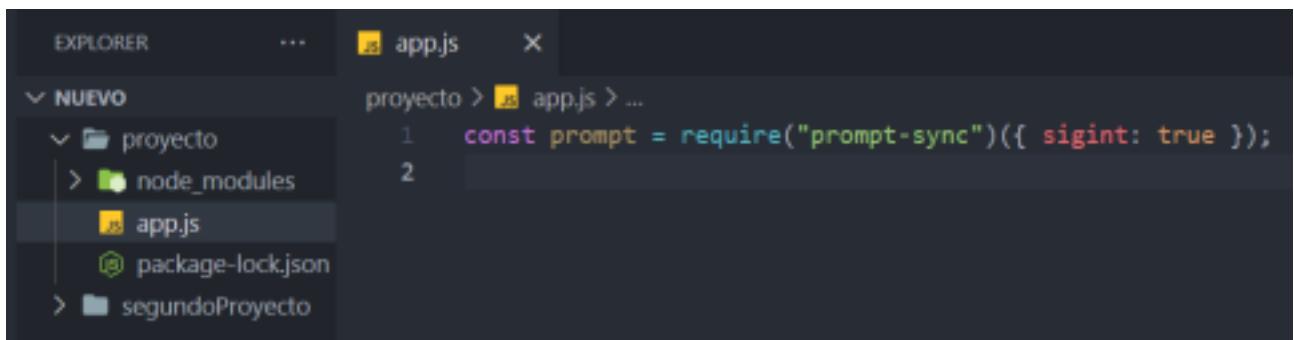
+ prompt-sync@4.2.0
added 3 packages from 3 contributors and audited 3 packages in 0.379s
found 1 moderate severity vulnerability
  run 'npm audit fix' to fix them, or 'npm audit' for details
PS C:\Users\leand\Desktop\Nuevo\segundoProyecto>
```

- e. No le damos importancia a los WARN y mensajes de error que surgen, realmente no son errores sino warnings.
- f. Para asegurarnos de que se haya instalado correctamente, en el directorio de VSCode deberíamos ver la carpeta **node_modules** y el archivo **package-lock.json**, así:



2. Una vez asegurados que tenemos correctamente instalada la herramienta, solo nos queda dejar la siguiente línea de código de configuración para poder utilizarla en nuestro archivo de JavaScript, el cual será donde escribiremos nuestro código en cuestión:

```
const prompt = require("prompt-sync")({ sigint: true });
```



Si tienes problemas con esa línea puedes intentar con:

```
const prompt = require('prompt-sync')();
```

3. Finalmente, para utilizarlo debemos simplemente llamar a la función `prompt()` la cual nos dejará ingresar datos y valores desde la consola, una vez ejecutado el código.

a. A continuación, un ejemplo sencillo de cómo utilizarla:

```
proyecto > app.js > ...
1 const prompt = require("prompt-sync")({ sigint: true });
2
3 let nombre = prompt("Ingrese su nombre: ");
4 console.log(`Hola ${nombre}!!!`);
5

PROBLEMS OUTPUT TERMINAL DEBUG CONSOLE

PS C:\Users\leand\Desktop\Nuevo\proyecto> node .\app.js
Ingrese su nombre: Leandro
Hola Leandro!!!
PS C:\Users\leand\Desktop\Nuevo\proyecto>
```

b. Vemos, como muestra la imagen, que la función `prompt()` está siendo guardada en la variable `nombre`, eso quiere decir que el dato que ingresemos por la consola al momento de su ejecución, será lo que realmente guardará la variable `nombre` (que, en efecto, es el nombre Leandro subrayado en azul que vemos en la terminal).

c. A su vez, también vemos que dentro de los paréntesis de la función se encuentra el string “**Ingresé su nombre:** “. El mismo se repite en la terminal a la izquierda del nombre subrayado en azul, eso nos indica que lo que ponemos entre paréntesis simulará “el título” de lo que queramos que el usuario ingrese por medio de nuestra función `prompt()`

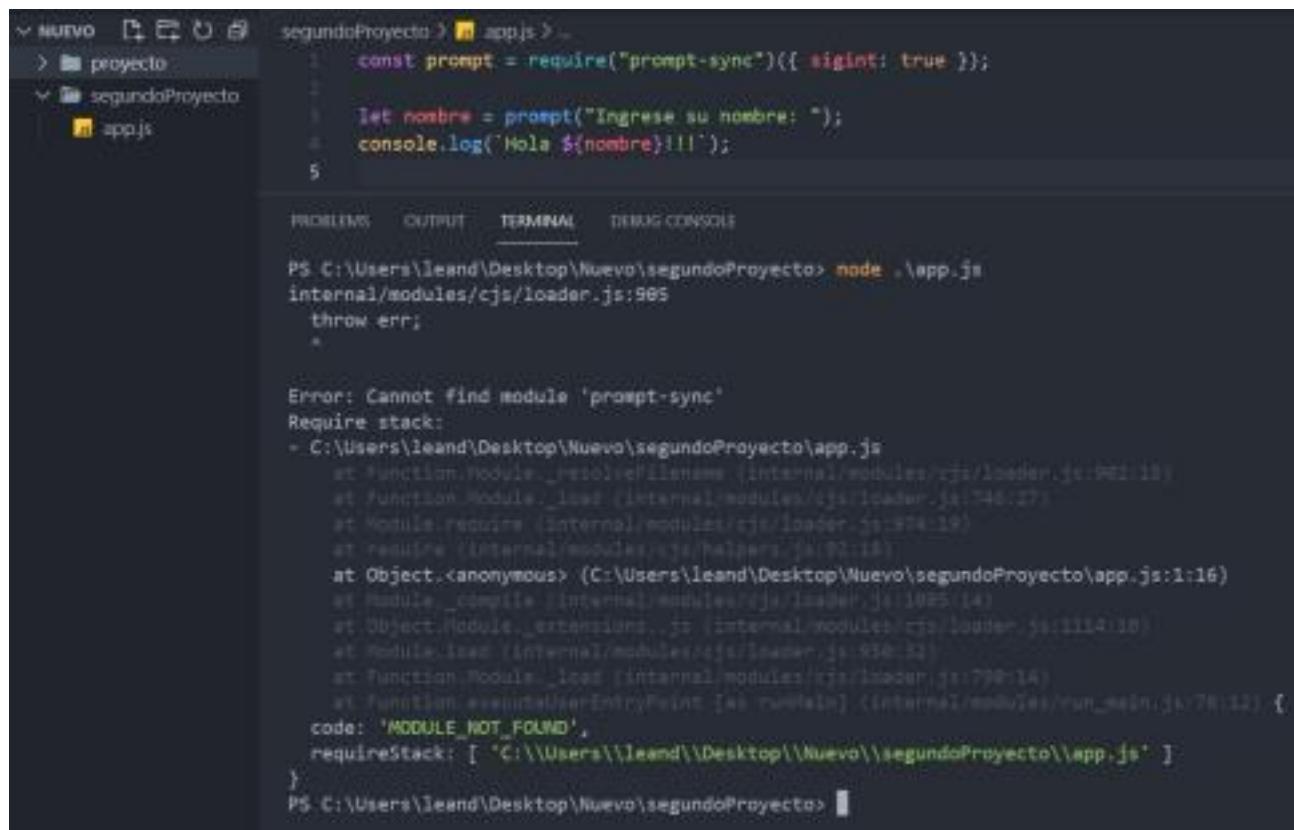
d. Finalmente vemos cómo se ejecuta el **console.log()**, el cual contiene el string Hola concatenado (por medio de literal templates) con nuestra variable nombre, que aloja el contenido de lo que la función prompt() retorna, en este caso el nombre Leandro.

Errores Comunes (Leer atentamente)

4. Podemos encontrarnos con los siguientes errores comunes:

a. Que estemos queriendo acceder a la función prompt() **sin haberla instalado**.

En este caso vemos cómo estamos trabajando en la carpeta **segundoProyecto** y tenemos la línea de código de configuración de prompt, pero **no tenemos instalada la herramienta**.



The screenshot shows a VS Code interface with a file tree on the left containing a 'segundoProyecto' folder with a 'app.js' file. The code in 'app.js' is:

```
const prompt = require("prompt-sync")({ sigint: true });

let nombre = prompt("Ingrese su nombre: ");
console.log(`Hola ${nombre}!!!`);
```

The 'TERMINAL' tab at the bottom shows the command `node .\app.js` being run, which results in an error:

```
PS C:\Users\leand\Desktop\Nuevo\segundoProyecto> node .\app.js
internal/modules/cjs/loader.js:985
  throw err;
  ^

Error: Cannot find module 'prompt-sync'
Require stack:
- C:\Users\leand\Desktop\Nuevo\segundoProyecto\app.js
  at Function.Module._resolveFilename (internal/modules/cjs/loader.js:981:11)
  at Function.Module._load (internal/modules/cjs/loader.js:920:27)
  at Module.require (internal/modules/cjs/loader.js:974:19)
  at require (internal/modules/cjs/helpers.js:92:18)
  at Object.<anonymous> (C:\Users\leand\Desktop\Nuevo\segundoProyecto\app.js:1:16)
  at module._compile (internal/modules/cjs/loader.js:1023:14)
  at Object.Module._extensions..js (internal/modules/cjs/loader.js:1054:10)
  at Module.load (internal/modules/cjs/loader.js:936:32)
  at Function.Module._load (internal/modules/cjs/loader.js:778:14)
  at Function.executeUserEntryPoint [as runMain] (internal/modules/run_main.js:67:12) {
    code: 'MODULE_NOT_FOUND',
    requireStack: [ 'C:\Users\leand\Desktop\Nuevo\segundoProyecto\app.js' ]
}
PS C:\Users\leand\Desktop\Nuevo\segundoProyecto>
```

b. Que hayamos olvidado de **agregar la línea de configuración de prompt**.

The screenshot shows a terminal window with the following details:

- File Explorer:** Shows a project named "Nuevo" containing a folder "proyecto" with files "node_modules", "app.js", and "package-lock.json".
- Terminal:** The command "node ./app.js" is run, resulting in the following output:

```
PS C:\Users\leand\Desktop\Nuevo\proyecto> node ./app.js
C:\Users\leand\Desktop\Nuevo\proyecto\app.js:1
let nombre = prompt("Ingrésese su nombre: ");
^

ReferenceError: prompt is not defined
    at Object.<anonymous> (C:\Users\leand\Desktop\Nuevo\proyecto\app.js:1:14)
    at Module._compile (internal/modules/cjs/loader.js:1065:14)
    at Object.Module._extensions..js (internal/modules/cjs/loader.js:1094:18)
    at Module.load (internal/modules/cjs/loader.js:936:32)
    at Function.Module._load (internal/modules/cjs/loader.js:770:36)
    at Function.executeUserEntryPoint [as mainModule] (internal/modules/run_main.js:70:62)
    at internal/main/run_main_module.js:17:47
PS C:\Users\leand\Desktop\Nuevo\proyecto>
```

Sucede cuando por más que tengamos instalada la herramienta, no agregamos la línea:

```
const prompt = require("prompt-sync")({ sigint: true });
```

c. Por último, puede surgir que, como consecuencia de un error, no quede instalado prompt-sync (si no tenemos la carpeta node_modules y el archivo package-lock.json es porque no se instaló)

Una razón por la que esto podría ocurrir es porque **no tenemos accesos de administrador en nuestra computadora**. En ese caso deberán solucionarlo ya que esta, como otras herramientas a lo largo de la carrera, **será sumamente necesaria**. En ese sentido, es imperativo que puedan instalar o desinstalar herramientas.

Estos son los errores más comunes a los que nos enfrentaremos. Por supuesto, siempre es una buena práctica **leer el error**, ya que, en la mayoría de los casos, **nos va a indicar en su mensaje cuál es el problema en cuestión que estamos teniendo**, y así poder solucionarlo o buscar la manera de hacerlo (ya sea mediante esta herramienta, algún colega, o incluso Google; quizás nuestra mejor herramienta y amigo)

Si no entendiste completamente el paso a paso, te dejo un video mas explicativo de como instalarlo (Pon los subtítulos porque está en portugués)

Video tutorial instalación Prompt Sync:

<https://www.youtube.com/watch?v=1TzCYVTC9tc&t=7s>