

ACÀMICA

---

# ¡Bienvenidas/os a Desarrollo Web Full Stack!



# Agenda

---

DWFS en mi vida

Temas del día:

- NodeJs
- Librerías
- Crear y Exportar Módulos

Break

Actividad: Ejercicios de práctica

Cierre



# ¿Dónde estamos?



# DWFS en mi vida



# DWFS en mi vida

¡Preparen sus charlas relámpago!

En **7 minutos con 7 slides** comparte con tus compañeros/as:

En qué problemas **estás aplicando lo aprendido** en DWFS y cómo lo estás haciendo.

**O bien, en qué problemas te gustaría aplicar** DWFS y cómo lo harías.

¡Elige algún tema o proyecto que te interese y relaciónalo con lo aprendido!

# NodeJS



# Un poco de historia...

Antiguamente, JavaScript solo podía correr en un navegador, desde el lado del cliente.

Luego, en 2009 Ryan Dahl creó un entorno que no requería un navegador para ejecutar código JavaScript en una computadora. Creó NodeJS...



A portrait of Ryan Dahl, a man with short brown hair, glasses, and a light beard, wearing a brown t-shirt. He is sitting on a red cushioned bench in a room with dark wood paneling and bookshelves filled with books in the background. A desk lamp is visible on a table behind him to the left.

Ryan Dahl  
Node.js Creator  
Software Engineer @ Joyent



Node.js es un entorno multiplataforma que permite **ejecutar código JavaScript sin la necesidad de utilizar un navegador.**



NodeJs requiere ser  
descargado e instalado.

Pueden encontrar el  
instalador en la página oficial:

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

## ¿Cómo se utiliza?

Una vez bajado e instalado NodeJs, desde la consola se puede ingresar *node* para acceder a las funcionalidades de NodeJS:

```
> node
Welcome to Node.js v12.13.0.
Type ".help" for more information.
>
```



## ¿Cómo se utiliza?

Una vez bajado e instalado NodeJs, desde la consola se puede ingresar *node* para acceder a las funcionalidades de NodeJS:

```
> node
Welcome to Node.js v12.13.0.
Type ".help" for more information.
>
```

Esta misma consola funciona parecido a la consola del navegador donde pueden ejecutar código JavaScript:

```
Welcome to Node.js v12.13.0.
Type ".help" for more information.
> const saludo = "hola mundo"
> console.log(saludo)
hola mundo
```



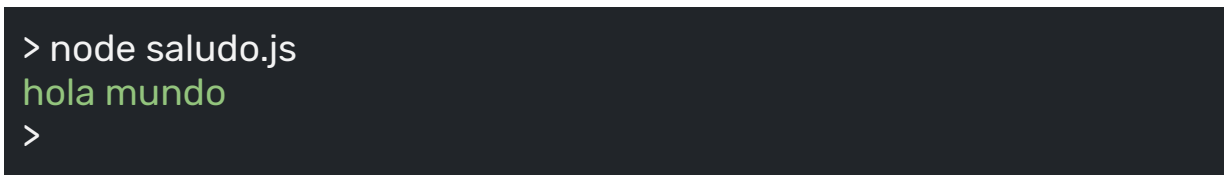
# Ejecutando un archivo JavaScript

Utilizando NodeJS se puede ejecutar un archivo JavaScript del ordenador.  
Dado el siguiente archivo `saludo.js`,

A screenshot of a code editor window with a dark theme. The title bar shows 'JS saludo.js' and a close button. The editor contains the following code:

```
JS saludo.js
1 console.log("hola mundo")
2
3
4
```

Si ejecutando el siguiente comando en la consola, van a poder ejecutar el archivo mostrando el resultado.

A screenshot of a terminal window with a dark background. It shows the command 'node saludo.js' being executed, followed by the output 'hola mundo' in green text.

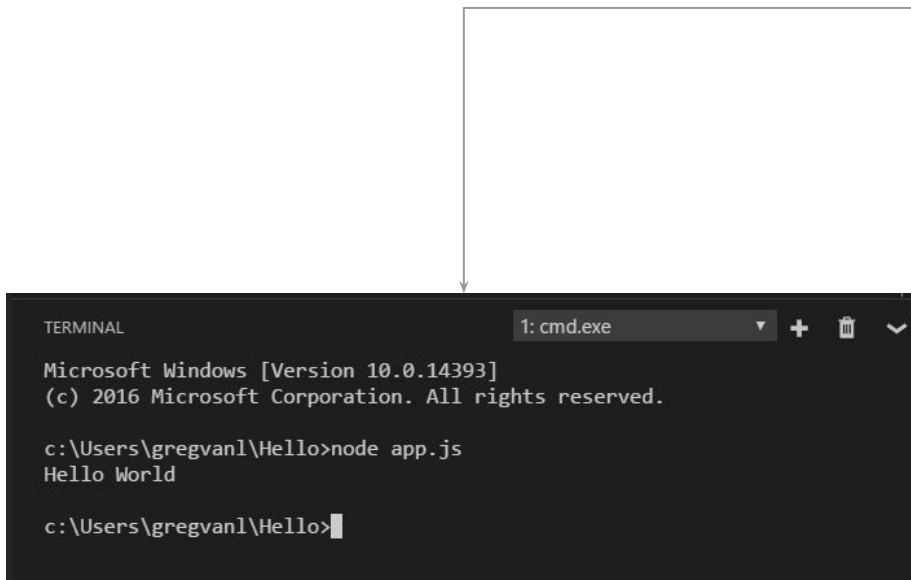
```
> node saludo.js
hola mundo
>
```

# Buena práctica



# Visual Studio Code Terminal

Si utilizan Visual Studio Code, desde el menú Vista, pueden acceder a una terminal (o consola) mismo desde el IDE.



```
TERMINAL
1: cmd.exe
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

c:\Users\gregvanl\Hello>node app.js
Hello World

c:\Users\gregvanl\Hello>
```





# Importando Librerías o Módulos



# ¿Cómo importar una librería o Módulo?

Para importar una librería o módulo se utiliza *require*.

Se puede pasar por parámetro tanto un nombre (pre-cargado en NodeJs) como una ruta de un archivo JavaScript.

```
const fs = require("fs") //Librería para interactuar con archivos del sistema operativo
console.log(fs.readdirSync()); //método que muestra los archivos en la carpeta actual
[ "saludo.js" ]
```

# Crear una librería o Módulo

libreria.js

```
const suma = (valor1, valor2) => valor1 + valor2;  
  
module.exports = { suma }; //module.exports se utiliza para exportar como módulo
```

script.js

```
const lib = require("./libreria") //obtiene libreria.js como módulo y lo almacena en lib  
const total = lib.suma(5, 4); //utiliza la función suma de lib y lo almacena  
console.log("Resultado de la suma: " + total)
```

Consola

```
> node script.js  
Resultado de la suma: 9  
>
```

# Buena práctica



# ¿Cuándo crear una librería propia?

Por lo general, vamos a crear librerías o módulos propios en las siguientes situaciones:

- **Tenemos el mismo código que se utiliza en más de un proyecto**
- **Queremos separar las responsabilidades o funcionalidades en distintos módulos**

El motivo más común es éste último ya que en aplicaciones grandes, lo ideal es tener todo bien organizado.

# Actividad práctica



# Práctica 1

1. Crear un archivo JavaScript.
2. Declarar un array con hobbies favoritos.
3. Muestra cada uno de los hobbies utilizando foreach para mostrar cada elemento



## Práctica 2.a

Crea un script que utilice la librería de [File System](#) y tenga las siguientes funcionalidades:

1. Crear un archivo de texto
2. Inserte un texto placeholder
3. Guardar archivo



## Práctica 2.b

Crear un segundo script que:

1. Lea el archivo generado
2. Pase todo su contenido a mayúsculas
3. Guarde el archivo

A close-up photograph of a white ceramic cup filled with a latte. The surface of the milk is decorated with intricate latte art, featuring a central heart shape surrounded by concentric, wavy lines. The cup is placed on a matching white saucer. In the background, a white napkin and a silver fork are visible, though they are out of focus. The overall lighting is soft and even, highlighting the textures of the coffee and the smooth surface of the cup.

**¡BREAK!**

---



# Actividad práctica



# Primera librería

1. Crear un archivo `app.js` que importe un módulo llamado `calculator.js`
2. Nuestro módulo debe tener la posibilidad de sumar, restar, multiplicar y dividir
3. Realizar cualquier operación desde `app.js`

# Guardar

Guardar cada ejecución de nuestro módulo en un archivo `log.txt`.

Ejemplo de `log.txt`:

$$7 \times 4 = 28$$

$$10 - 4 = 6$$

$$15 / 3 = 5$$

# Recursos



# Si te quedaste con ganas de más...

- <https://nodejs.org/en/>
- <https://codeburst.io/the-only-nodejs-introduction-youll-ever-need-d969a47ef219>
- <https://momentjs.com/>



# Para la próxima

---

- Investigar sobre NPM
- Ver plataforma módulo 1.4



ACÀMICA