

ACÀMICA

# Agenda

---

Daily

Programo: Node JS

Buenas prácticas

Break

Programamos: Hobbies favoritos

Programan: Librerías & File System

Cierre



# NODE JS

Node.js es un entorno que nos permite ejecutar JavaScript en el servidor con una arquitectura orientada a eventos y basado en el motor V8

# Daily



Daily



## Sincronizando...

### Bitácora



¿Cómo te ha ido?  
¿Obstáculos?  
¿Cómo seguimos?

### Challenge



¿Cómo te ha ido?  
¿Obstáculos?  
¿Cómo seguimos?

# Node JS



# Node JS

Node JS es un entorno de ejecución para JavaScript del lado del servidor.

Instalación de [NodeJs](#) 14.17LTS



## Famous Apps Built On Node.js



NETFLIX



GROUPON



# Programo

mentores/as



## Node JS

Veamos una demostración de Node JS.

- index.js

```
//Hola Mundo  
console.log("Hola Mundo");
```

Importando el módulo nativo fs de NodeJs

- escribir.js

```
const fs = require('fs');  
fs.writeFile('test.txt', 'Hola Mundo', (error) => {  
  if (error) {  
    return console.error(error);  
  }  
  console.log("Archivo guardado exitosamente");  
});
```



# Node JS

Importando el módulo nativo de fs NodeJs  
leer.js

```
const fs = require('fs');  
fs.readFile('test.txt', 'utf8', (error, data) => {  
  if (error) {  
    return console.error(error);  
  }  
  console.log(data);  
});
```



# Buenas prácticas



# ¿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.

# Módulo User

```
class User {  
  nombre = 'Juan';  
  apellido = 'Perez';  
  email = 'jperez@hotmail.com';  
  telefono = '(5411) 4574-2299';  
  
  fullname = function() {  
    return this.nombre + ' ' + this.apellido;  
  }  
}  
  
//Exporto la clase User  
module.exports = User;
```

```
var User = require('./user_module.js')  
var person = new User();  
  
//Imprimo objeto completo  
console.log(person);  
  
//Imprimo propiedad  
console.log(person.nombre);  
  
//Ejecuto metodo  
console.log(person.fullname());
```

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 warm, creating a cozy atmosphere.

**¡BREAK!**

---

Ph. Credit: Drew Coffmann



# Programamos

todos/as





# Programamos

Programemos la siguiente estructura:

- Crear un archivo con extensión js.
- Declarar un array con hobbies favoritos.
- Muestra cada uno de los hobbies utilizando [foreach](#) para mostrar cada elemento.

# Programan

estudiantes



Actividad:  
¡Programan!

## File System

Crea un script que utilice la librería de File System con las siguientes funcionalidades:

- Crear un archivo de texto
- Inserte un texto placeholder
- Guardar archivo



Actividad:  
¡Programan!



## Calculadora

Crea un archivo `app.js` que importe un módulo llamado `calculator.js`.

Nuestro módulo debe tener la posibilidad de:

- Sumar
- Restar
- Multiplicar
- Dividir

Realizar cualquier operación desde `app.js`.

Actividad:  
¡Programan!



## Calculadora

Guarda cada ejecución de nuestro módulo en un archivo log.txt.

Ejemplo de log.txt:

$$7 \times 4 = 28$$

$$10 - 4 = 6$$

$$15 / 3 = 5$$

# Para la próxima

---

- 1) Termina el ejercicio del encuentro de hoy.
- 2) Lee la bitácora 40 y carga las dudas que tengas al Trello.
- 3) Resuelve el challenge.

En el encuentro que viene uno/a de ustedes será seleccionado para mostrar el ejercicio de hoy y otro/a mostrará cómo resolvió el challenge de la bitácora. De esta manera, ¡aprendemos todos/as de (y con) todos/as, así que vengan preparados/as!

ACÀMICA