

# {JS}

**Clase 16**

# ◀ Índice ▶

## Clases en JS

Definición de clases

---

Métodos privados

---

Métodos estáticos

---

Getters y setters

---

Herencia

# **“Classes” en Javascript**

- Introducidas en ECMAScript 2015 o ES6
- Recordemos que JS se basa en prototipos
- En prototipos existe la “herencia” o delegación
- Sugar syntax...
- Al final no dejan de ser funciones constructoras
- Ventajas de la POO

# Definición de clases

- Recordemos las funciones constructoras
- Declaración de una “instancia” de función constructora
- ¿Dónde viven los métodos en un objeto de función constructora?
- Añadámoslo al prototipo de usuario
- Comprobemos donde vive ahora la función

# Nuestra primera clase

- Convirtámoslo en clase
- ¿Qué es el constructor?
- Declaremos nuestro primer objeto
- Declaremos el primer método saludar
- ¿Dónde vive?
- Arreglémoslo
- ¿Dónde vive ahora el método?



# Métodos o propiedades privados

- ¿Cómo hacíamos atributos privados en funciones constructoras?
- ¿Cómo lo hacemos con clases?
- Declaremos nuestra primera propiedad privada
- Peculiaridad de Chrome para devs
- La importancia de la visibilidad o no de propiedades
- Funciona para propiedades y métodos

# Métodos estáticos

- ¿Qué son?
- Ya lo hemos usado
- ¿Dónde viven?
- Hagamos un ejemplo
- Sirve tanto para métodos como para propiedades

# Getters y Setters

- ¿Qué son?
- Su importancia
- Su uso, validación por ejemplo
- Ejemplo, edad de un alumno

# Herencia

- ¿Qué es?
- Su importancia
- Ejemplo de uso
- Llamar al constructor del padre
- Sobreescibir métodos, saludar



**◀ Despedida ▶**

Email

**bienvenidosaez@gmail.com**

Instagram

**@bienvenidosaez**

Youtube

**youtube.com/bienvenidosaez**