

## JS/Ejercicio\_01\_POO.html

```

<!DOCTYPE html>      <!-- Ejercicio_01_POO.html -->
<html lang="es">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>JS</title>
  <link rel="stylesheet" href="bootstrap.min.css">
  <script>/*

```

```

    +-----+
    |  Persona  |
    +-----+
    | + nombre: str |
    | - edad: int   |
    | + ciudad: str |
    +-----+
    | # mostrarInfo(): JSON |
    | + imprimir(): str     |
    +-----+
      ▲
      |
  -----
  |                                     |
+-----+                             +-----+
|  Alumno  |                             |  Profesor  |
+-----+                             +-----+
| - curso: str       |                   | - materia: str       |
| - nota: float      |                   | - salario: float     |
+-----+                             +-----+
| + imprimir(): str |                   | + imprimir(): str |
+-----+                             +-----+ */

```

```

class Persona {
  // Atributos
  nombre = "";
  #edad = 0; // Privado!
  ciudad = "";
  // constructor
  constructor(nombre, edad, ciudad) {
    this.nombre = nombre;
    this.#edad = edad;
    this.ciudad = ciudad;
  }
  // Método protected! (OJO, en UML se pone #)
  _mostrarInfo() {
    return {
      NOMBRE: this.nombre,
      EDAD: this.#edad,
      CIUDAD: this.ciudad,
      TIPO: this.constructor.name
    }
  }
}

```

```
        imprimir() {
            return JSON.stringify(this._mostrarInfo());
        }
        // Setter y getter
        get edad() {
            return this.#edad;
        }
        set edad (cantidad) {
            if(cantidad >0) {
                this.#edad = cantidad;
            }
        }
    }
    class Alumno extends Persona {
        curso = "";
        #nota = 0;
        constructor (nombre, edad, ciudad, curso, nota) {
            super(nombre, edad, ciudad);
            this.curso = curso;
            this.#nota = nota;
        }
        // mostrarInfo privado
        #mostrar () {
            return {
                // Para personalizar la salida JSON
                // En las clases hijas, usaremos super
                // ...super._<nombre_metodo>
                ...super._mostrarInfo(),
                CURSO: this.curso,
                NOTA: this.#nota
            }
        }

        imprimir() {
            return JSON.stringify(this.#mostrar(), null, 2);
        }
    }
    function iniciar() {
        let Alfonso =
            new Alumno("Alfonso Casillas", 26, "Sevilla", "DIWEB", 8);
        alert(`Datos Alumno: \n ${Alfonso.imprimir()}`);
    }
</script>
</head>
<body>
    <main class="m-3 p-3">
        <form action="#" method="post" class="form">
            <button type="submit" class="btn btn-success"
                onclick="iniciar()">Iniciar</button>
        </form>
    </main>
</body>
</html>
```