

INSTRUCCIONES

- El examen debe entregarse a través del aula virtual mediante un archivo ZIP con el nombre **Examen_Unidad4_Iniciales**. Las iniciales serán la primera letra de tu nombre, primer y segundo apellido. Por ejemplo, si tu nombre es José López Pérez, el archivo se denominará **Examen_Unidad4_JLP**.
- La nomenclatura anterior con las iniciales se empleará en todos los ejercicios con el nombre archivo que se indicará en cada caso.
- Es obligatorio crear un archivo externo para JavaScript en una carpeta llamada “js”.
- **Si los ejercicios no cumplen la nomenclatura no se corregirán.**
- Se pueden emplear apuntes, ejemplos y ejercicios de clase. Se impedirá el uso de internet. **Si se detecta conexión a la red o plagio, el examen acabará inmediatamente con la calificación de 0.**

Uso de módulos en ejercicios 1 y 2 (1,5 puntos). Se puede realizar los ejercicios 1 y 2 sin emplear módulos creando otros archivos JavaScript para realizar las pruebas o duplicar el código necesario para que funcione, aunque no puntuará este apartado.

Ejercicio 1 (2,75 puntos). Nomenclatura: El HTML estará en la carpeta raíz de la entrega y se denominará **Ejercicio 1_Iniciales.html**. El código JavaScript para probar el módulo se guardará en la carpeta “js” con el nombre **Ejercicio 1_Iniciales.js** y el módulo será **Empresa_Iniciales** (las iniciales serán la primera letra de tu nombre, primer y segundo apellido).

- a) **(2 puntos; 1 punto las propiedades privadas y 1 punto las propiedades y métodos estáticos)** Crea un módulo que exporta una clase por defecto “Empresa” que contiene las siguientes propiedades y métodos:
- Una propiedad privada de tipo String llamada “cif”.
 - Una propiedad privada de tipo String llamada “nombreFiscal”.
 - Una propiedad privada de tipo Number llamada “añoFundación”.
 - El constructor de la clase recibirá como parámetros el cif, nombreFiscal y añoFundación. Son campos privados que solo se puede acceder con un get y set.
 - Una propiedad pública estática que almacena en un Set todos los años de fundación de una empresa. El método estático “calcularAñoMinimo” devolverá el año más antiguo de fundación de alguna empresa.
- b) **(0,5 puntos)** En el archivo **Ejercicio 1_Iniciales.js**, crea un array con dos objetos de prueba “Empresa”. Recorre el array y muestra las propiedades CIF, nombreFiscal y añoFundación. Muestra también por pantalla el año mínimo de fundación de todos los objetos.
- c) **(0,25 puntos)** La información de prueba de **Ejercicio 1_Iniciales.js** se visualizará en un div o p del HTML denominado **Ejercicio 1_Iniciales.html**.

Ejercicio 2 (2,75 puntos). Nomenclatura: Se puede reutilizar el ejercicio anterior y renombrar el HTML a **Ejercicio 12_Iniciales.html**. El código JavaScript para probar el módulo se guardará en la carpeta “js” y se puede renombrar a **Ejercicio 12_Iniciales.js** con el módulo **EmpresalIniciales.js**. El módulo con la filial se denominará **FilialIniciales.js** (las iniciales serán la primera letra de tu nombre, primer y segundo apellido). Por el contrario, puedes crear archivos nuevos a partir del ejercicio 1 que se denominarán **Ejercicio 2_Iniciales.html** y **Ejercicio 2_Iniciales.js**

- a) **(1 punto)** Crea otro módulo que exporta una clase “Filial” (no será la exportación por defecto) que hereda “Empresa” y añade un campo privado de tipo objeto llamado “detalles” con la siguiente información:
- El objeto tendrá al menos dos propiedades: un String llamado “cifFilial” y un campo de tipo Date llamado “fechaCreacion”, aunque esto es irrelevante para la clase.
 - Tendrá un constructor que incluirá el nuevo campo “detalles”. Las propiedades no se añaden aquí y el constructor solo asignará el valor que recibe como parámetro.
 - El campo detalles de “Filial” tendrá un “getter” que devolverá un String concatenado el código del instituto y la fecha de creación de este objeto empleando desestructuración al menos una vez. No se hacen comprobaciones y suponemos la existencia de estas propiedades en el objeto “detalles”.
- b) **(1,5 puntos)** En el archivo **Ejercicio 12_Iniciales.js** o **Ejercicio 2_Iniciales.js** según el caso, crea un objeto con los campos “cifFilial”, “fechaCreacion” (igual a la actual) y un Symbol con descripción “Competencias” y como valor un String con una descripción de lo que hace la filial de la empresa. Clona el objeto anterior incluyendo el Symbol y recorre todos los campos del objeto clonado también considerando el Symbol. Puedes utilizar la misma etiqueta que en el HTML del ejercicio 1 para visualizar información. Si la descripción es de tipo Symbol, no hace falta formatear la salida.
- c) **(0,25 puntos)** Seguimos en el JavaScript de pruebas **Ejercicio 12_Iniciales.js** o **Ejercicio 2_Iniciales.js**. Crea una clase “Filial” y asigna uno de los objetos creados al campo “detalles”. Muestra por pantalla el getter de “detalles” para comprobar que solo muestra dos campos. Puedes utilizar la misma etiqueta que en el HTML del ejercicio 1 para visualizar información.

Ejercicio 3 (3 puntos)

Nomenclatura: El HTML estará en la carpeta raíz de la entrega y se denominará **Ejercicio 3_Iniciales.html**. El código JavaScript se guardará en la carpeta “js” con el nombre **Ejercicio 3_Iniciales.js** (las iniciales serán la primera letra de tu nombre, primer y segundo apellido).

- a) **(1 punto)** Crea un Map “clasificacion” con nombres de entidades deportivas (reales o inventadas) y un valor igual al número de puntos. Puedes añadir un numero de entradas suficiente como para poder hacer pruebas. Muestra el map por pantalla en una etiqueta div o p del HTML empleando desestructuración.
- b) **(1 punto)** Convierte el Map a un Array llamado “descenso” incluyendo aquellos equipos con menos de 42 puntos. Muestra el Array por pantalla en una etiqueta div o p del HTML incluyendo el nombre del equipo y puntos. Solo se permite recorrer el Array manualmente para mostrarlo por pantalla. El filtro será con métodos que no impliquen hacer ningún recorrido manual.
- c) **(1 punto)** Calcula la media de todos los equipos empleando algún método del objeto Array. El ejercicio no contará si se recorre el Map manualmente para calcular la media.