

DESARROLLO WEB EN ENTORNO CLIENTE	
UT1. Javascript	Nota
Apellidos:	
Nombre:	
Fecha:	

NOTA: Debes crear una carpeta con un único index.html y un único .js donde tendrás todo el código debidamente ordenado y comentado y las llamadas a las funciones de los apartados a, b, c y d.

1. Dado un vector de objetos literales con el siguiente formato: {nombreCompleto:"Ana Pérez Soler", DNI: "21147465T", edad: 21}:
 - a. Crea una función que reciba como parámetro el array original y devuelva un array con los identificadores de los usuarios. Para ello debes coger la primera letra (minúsculas y sin tildes) del nombre, las tres primeras letras del primer apellido (minúsculas y sin tildes), las tres primeras del segundo apellido (minúsculas y sin tildes) y los tres últimos dígitos del DNI. Usa la función map.(3p)
 NOTA: Se entiende que los nombres y los apellidos no son compuestos y todos tienen tres o más caracteres.
 - b. Crea una función que reciba como parámetro el array original y devuelva un array con los DNI que sean válidos. NOTA: Los DNI pueden tener la letra en mayúscula o minúscula pero no tienen un separador entre número y letra. Usa la función filter y map. (3p)
 - c. Crea una función que reciba como parámetro el array original y devuelva un array con los objetos ordenados por edad de mayor a menor. NOTA: Usar la función sort. (2p)
 - d. Crea una función que reciba como parámetro el array original y devuelva un array con las edades que sean números casi primos (es un número que solo es divisible por sí mismo, la unidad y por un solo número que no sea ni la unidad ni si mismo). Usa la función filter y map. (2p)

Ejemplo.

Entrada:

```
[{nombre:"Ana Pérez Soler", DNI: "21147465T", edad: 4},
 {nombre:"Álvaro García Castro", DNI: "24545535G", edad: 32},
 {nombre:"Paloma González Máxim", DNI: "45678912L", edad: 9},
 {nombre:"Alberto Alguacil Alcalde", DNI: "79753330W", edad: 17}
]
```

Salida:

A: ["apersol465", "agarcas535", "pgonmax912", "aalgalc330"]

B: ["21147465T", "24545535G", "79753330W"]

C: [{nombre:"Álvaro García Castro", DNI: "24545535G", edad: 32},{nombre:"Alberto Alguacil Alcalde", DNI: "79753330W", edad: 17},{nombre:"Paloma González Máxim", DNI: "45678912L", edad: 9},{nombre:"Ana Pérez Soler", DNI: "21147465T", edad: 4}]

D: [9,4]

¿Cómo saber si un DNI es válido?

Para calcular la letra del DNI o carácter de verificación solo debes de realizar los siguientes pasos:

1. Dividir la parte numérica del DNI entre el número 23.
2. Tomamos el resto de dicha división y buscamos en la siguiente tabla cual carácter es el que le corresponde:

RESTO	0	1	2	3	4	5	6	7	8	9	10	11
LETRA	T	R	W	A	G	M	Y	F	P	D	X	B

RESTO	12	13	14	15	16	17	18	19	20	21	22
LETRA	N	J	Z	S	Q	V	H	L	C	K	E

A continuación te mostraremos un ejemplo:

Imaginemos que un cliente rellena el formulario de registro de una tienda online con el número de DNI 12151135G. Para verificar si el DNI es válido solo debemos dividir 12151135 entre 23. Al realizar dicha operación tendremos que el resto de la división es 5. Echando una pequeña ojeada a la tabla que te presentamos más arriba, podemos ver que el carácter de verificación para el número suministrado es M y no G, lo cual indica que el número de DNI no es válido.

Tabla de caracteres:

['T', 'R', 'W', 'A', 'G', 'M', 'Y', 'F', 'P', 'D', 'X', 'B', 'N', 'J', 'Z', 'S', 'Q', 'V', 'H', 'L', 'C', 'K', 'E']

DNI válidos: 21147465T, 24545535G, 79753330W

DNI inválidos: 45678912L, 12345678P

¿Cómo saber si un número es casi primo?

Un número casi primo: es un número que solo es divisible por sí mismo, la unidad y por un solo número que no sea ni la unidad ni si mismo.

Ejemplo:

- 2 no es un número casi primo, porque es divisible por 1 y por 2, pero no por otro número.
- 4 es un número casi primo, porque es divisible por 1, por 2 y por 4.
- 9 es un número casi primo, porque es divisible por 1, por 3 y por 9.
- 8 no es un número casi primo, porque es divisible por 1, por 8 y por 2, pero además también es divisible por 4.