

## PRÁCTICA 2.04 Objetos predefinidos

### Normas de entrega

- En cuanto al **código**:
  - en la **presentación interna**, importan los **comentarios**, la claridad del código, la significación de los nombres elegidos; todo esto debe permitir considerar al programa como **autodocumentado**. No será necesario explicar que es un **if** un **for** pero sí su funcionalidad. Hay que comentar las cosas destacadas y, sobre todo, las **funciones** y **clases** empleadas. La ausencia de comentarios será penalizada,
  - en la **presentación externa**, importan las leyendas aclaratorias, información en pantalla y avisos de ejecución que favorezcan el uso de la aplicación,
  - si no se especifica lo contrario, la información resultante del programa debe aparecer en la consola del navegador **console.log(información)**,
  - los ejercicios deben realizarse usando **JavaScript ES6** y usar el modo estricto (**use strict**) No se podrá utilizar *jQuery* ni cualquier otra biblioteca (si no se especifica lo contrario en el enunciado),
  - para el nombre de **variables**, **constantes** y **funciones** se utilizará *lowerCamelCase*,
- En cuanto a la **entrega** de los archivos que componen los ejercicios:
  - todos los ejercicios en **una carpeta** (creando las **subcarpetas** necesarias para documentación anexa como imágenes o estilos) cuyo nombre queda a discreción del discente,
  - el nombre de los ficheros necesarios para resolver el ejercicio será el número de ejercicio que contenga,
  - el código contendrá ejemplos de ejecución, si procede, y
  - la carpeta será comprimida en formato **ZIP** y será subida a **Aules** de forma puntual.

### Ejercicio 1 - toCani

Crear una función que reciba una cadena como parámetro y la devuelva transformada en lenguaje Cani. Por ejemplo, si se la pasa la cadena "una cadena cani es como esta" obtendremos "UnA KaDeNa kAnI Es kOmO EsTaHHH".

Para ello hay que:

- alternar el uso de MAYÚSCULAS y minúsculas,
- sustituir la letra C por la K y
- añadir entre dos y cinco letras H al final de la cadena.

### Ejercicio 2 - Temporizador

Diseña una función temporizador que reciba como dos parámetros: minutos y segundos. Cada segundo mostrará por consola el tiempo que le queda al temporizador hasta llegar a cero. En el caso que sólo reciba un parámetro, considerará que son los segundos desde donde comenzará la cuenta atrás. La función se llamará de este modo:

```
function temporizador(minutos, segundos);
```

Por ejemplo **temporizador(2,50)**; indica 2 minutos y 50 segundos. Si alguno de los valores que recibe como parámetros son negativos, de un tipo inesperado o los segundos superan el número 59, la función debe lanzar un error ~~insultando~~ informando al usuario.

La salida del temporizador debe estar debidamente formateada.

Para establecer el intervalo de repetición de la función emplear **setInterval** del siguiente modo

```
const id = setInterval(actualizarReloj, 1000);
```

Cuando el temporizador llegue a cero deberemos parar las repeticiones, usar:

```
clearInterval(id);
```

### Ejercicio 3 - DNI

Realiza una función que cada cinco segundos (mediante setInterval) solicite un número de DNI sin la letra (llamando a prompt) hasta que alguien le escriba la cadena -1. En ese momento el programa debe mostrar los DNIS con las letras de todos los DNI introducidos por consola. Para saber cómo calcular la letra del DNI puedes verlo [aquí](#).

Ejemplo resultado:

Has introducido 4 DNI's	<a href="#">ejercicio3.js:33</a>
23456767X	<a href="#">ejercicio3.js:35</a>
23456762M	<a href="#">ejercicio3.js:35</a>
23456769N	<a href="#">ejercicio3.js:35</a>
23456765P	<a href="#">ejercicio3.js:35</a>

#### Ejercicio 4 - Fechas

Escribe un programa que pasados tres segundos muestre **una vez** la fecha actual del sistema por consola debidamente formateada y en castellano.

Ejemplo resultado:

```
Sábado, 14 de Septiembre de 2024
```

#### Ejercicio 5 - Más DNI

Crea una función que reciba una letra mayúscula como parámetro y calcule cuantos DNI de entre 48357001 a 48360000 tienen esa letra (reutiliza el código del Ejercicio 3).

Tras esto mostrará por consola el número de documentos encontrados y el listado debidamente formateados de una sola vez sin usar un array.

Ejemplo resultado:

```
Hay 43 DNI entre 48357001 y 48357999
48357012H 48357035H 48357058H 48357081H 48357104H 48357127H 48357150H
48357173H 48357196H 48357219H 48357242H 48357265H 48357288H 48357311H
48357334H 48357357H 48357380H 48357403H 48357426H 48357449H 48357472H
48357495H 48357518H 48357541H 48357564H 48357587H 48357610H 48357633H
48357656H 48357679H 48357702H 48357725H 48357748H 48357771H 48357794H
48357817H 48357840H 48357863H 48357886H 48357909H 48357932H 48357955H
48357978H
```

#### Ejercicio 6 - Números

Escribe una función que calcule cuántos números son a la vez primos y palíndromos (capicua) hasta 100000 y los meta todos en un array. Luego muestra el listado por consola debidamente formateado.