

PRÁCTICA 2.01 Empezando con la sintaxis

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*,
 - para la resolución de los ejercicios hay que usar **funciones**.
- 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 - Estás delgado

Marcos y Juan quieren comparar su IMC (Índice de Masa Corporal), que se calcula con la siguiente fórmula:

$$\text{IMC} = \text{peso}(\text{kg}) / \text{altura}(\text{m})^2 = \text{masa} / (\text{altura} * \text{altura})$$

- guarda las masas y alturas de Marcos y Juan en variables (pon los valores que quieras).
- calcula ambos IMC y preséntalos por consola con dos decimales.
- crea una variable booleana que contenga información sobre si Marcos tiene un IMC mayor que Juan.
- muestra un texto en la consola que contenga la variable del paso 3. Algo así como:
¿Tiene Marcos un IMC menor que el de Juan?: true

Ejercicio 2 - Juego mejor que tú

Juan y Miguel juegan al baloncesto en equipos diferentes. En los últimos tres partidos el equipo de Juan consiguió 89, 120 y 103 puntos, mientras que el de Miguel consiguió 116, 94 y 123. Elige la estructura de datos que quieras para almacenar esta información: array (u objeto JSON).

Diseña y programa funciones para que realicen las siguientes tareas:

- Imprimir una cadena que indique los puntos de un equipo, se le pasará el array y el nombre del equipo.
- Función calcular la puntuación media de puntos (los puntos se le pasan como parámetro),
- Otra para mostrar qué equipo tiene mejor media de puntuación e imprime el ganador en la consola. Incluye en la salida también la media de puntuación del equipo. No olvides tener en cuenta el posible caso de un empate (misma puntuación media). Como parámetros se entregan los puntos y nombre de cada equipo.

Ejemplo resultado:

```
Los puntos del equipo de Juan son: 89 - 120 - 103
La puntuación media del equipo de Juan es: 104.00
Los puntos del equipo de Miguel son: 116 - 94 - 123
La puntuación media del equipo de Miguel es: 111.00
El equipo ganador es el equipo de Miguel y la puntuación media es: 111.00
```

- María también juega al baloncesto y su equipo consiguió 97, 134 y 105 puntos. Rediseña las funciones anteriores para que acepten también a María.

Ejercicio 3 - Números

Crea una función que reciba dos números por parámetro. El primer número indicará cuantas veces se debe imprimir el segundo por pantalla, pero en cada iteración muestra el valor anterior multiplicado por 2. Por ejemplo, si recibe 4 y 6 la salida será: 6 12 24 48.

Ejercicio 4 - Facturas

Crea una función que reciba tres parámetros (nombre de producto, precio e impuesto en porcentaje sobre 100). La función hará lo siguiente:

- los parámetros deberán tener un valor por defecto por si no los recibe que deben ser: "Producto genérico", 100 y 21,
- si los números no son válidos (**isNaN()**) muestra un error. Si son válidos, muestra por consola el nombre del producto y el precio final contando impuestos.

Ejemplo resultado:

```
Producto: Producto Genérico - Precio: 121
facturas.js:15 Error. Número(s) no valido(s).
facturas.js:16 Producto: Pelota - Precio: 60.5
facturas.js:17 Producto: TV FHD - Precio: 1058.75
```

Ejercicio 5 - Propinas

Juan y su familia se fueron de vacaciones a EE.UU. y comieron en tres restaurantes distintos. Pagaron 124, 48 y 268 \$. Allí todo el mundo deja propina, así que Juan quiere crear una calculadora de propinas.

Ha pensado que le gustaría dar un 20 % de propina si la factura es menor de 50 \$, 15 % si la factura está entre 50 \$ - 200 \$ y un 10 % si la factura es de más de 200 \$.

Al final, a Juan le encantaría tener tres *arrays*:

- el primero contiene el valor de todas las facturas. Este no se calcula, hay que crear el *array* poniendo a mano los valores,
- el segundo contiene las tres propinas (que se rellenan de forma automática),
- el tercero contiene las tres cantidades finales que tiene que pagar (que se rellena de forma automática).

Crea una función que reciba el primer *array* como parámetro y que muestre el resultado por pantalla debidamente formateado.

- consola el nombre del producto y el precio final contando impuestos.

Ejemplo resultado:

FACTURA	PROPINA	TOTAL
124.00	18.60	142.60
48.00	9.60	57.60
268.00	26.80	294.80