

Apellidos: .....

Nombre: ..... UVUS: .....



Bases de datos (GE/DGME)

31/05/2024

Sexta prueba de evaluación alternativa

---

**Ejercicio 1 (1,5 puntos)** Crear el esqueleto básico de una página HTML `ej01.html`, con el cuerpo vacío, e incorporar el código dinámico necesario para que, al cargar dicha página, realice lo siguiente:

- (0,25) Imprimir en la pantalla una cabecera de primer nivel indicando *Ejercicio 1*, solicitar al usuario una cadena de texto y calcular su número de caracteres.
- (0,625) Si dicho número es menor o igual a 7, deberá imprimir por consola todas las subcadenas de la misma que partan del primer carácter. Ej: para la cadena palabra imprimiría *p*, *pa*, *pal*, *pala*, *palab*, *palabr* y *palabra*.
- (0,625) En caso contrario, deberá calcular el producto de los números impares desde el 1 hasta el tamaño de la cadena (ambos *inclusive*). A continuación, pedirá confirmación al usuario para mostrar el resultado en un mensaje emergente. En caso de rechazar mostrarlo, el mensaje deberá escribir directamente a la salida de la página, de modo que se visualice en el navegador.

**Ejercicio 2 (1,5 puntos)** Partiendo del archivo `ej02.html` que se le proporciona, incorporar el código dinámico en cliente para hacer lo siguiente:

1. (0,625) Definir una función que reciba como parámetros dos números y un tipo de operación y que, según este tipo sea “sum”, “res”, “mul” o “div”, la función devuelva el resultado de realizar, respectivamente, la adición, sustracción, producto y división entre los números recibidos, y siempre tomando como primer operando el mayor de los números recibidos y como segundo operando el menor de ellos. En caso de recibir una operación que no sea ninguna de las 4 indicadas, se deberá devolver un -1.
2. (0,625) Asignar un manejador de evento al botón, de manera que, al pulsarse el mismo, se calcule la operación indicada por el desplegable sobre los números dados por las entradas de número proporcionadas en el formulario, y se muestre en el div “resultado”.
3. (0,25) Asigne el mismo manejador de evento anterior pero ahora al evento de cambiar la operación.

**Ejercicio 3 (2,5 puntos)** Crear el esqueleto básico de una página HTML `ej03.html`, con el cuerpo vacío. Teniendo en cuenta la información proporcionada por Wikipedia sobre las galardonadas con el balón de oro femenino:

Año	Pos.	Jugadora	Club (es)	Puntos
2018	1st	 <b>Ada Hegerberg</b>	 <b>Lyon</b>	<b>136</b>
	2nd	 Pernille Harder	 Wolfsburg	130
	3rd	 Dzszenifer Marozsán	 Lyon	86
2019	1st	 <b>Megan Rapinoe</b>	 <b>Reign FC</b>	<b>230</b>
	2nd	 Lucy Bronze	 Lyon	94
	3rd	 Alex Morgan	 Orlando Pride	68
2021	1st	 <b>Alexia Putellas</b>	 <b>Fútbol Club Barcelona</b>	<b>186</b>
	2nd	 Jennifer Hermoso	 Fútbol Club Barcelona	84
	3rd	 Sam Kerr	 Chelsea Football Club	46
2022	1st	 <b>Alexia Putellas</b>	 <b>Fútbol Club Barcelona</b>	<b>178</b>
	2nd	 Beath Mead	 Arsenal Fútbol Club	152
	3rd	 Sam Kerr	 Chelsea Football Club	74
2023	1st	 <b>Aitana Bonmatí</b>	 <b>Fútbol Club Barcelona</b>	<b>266</b>
	2nd	 Sam Kerr	 Chelsea Football Club	87
	3rd	 Salma Paralluelo	 Fútbol Club Barcelona	49

, se pide incorporar el código dinámico necesario para realizar lo siguiente:

- (0,5) Crear una estructura de JavaScript adecuada que almacene, a nivel de cada año, la información de las jugadoras que obtuvieron el primer, segundo y tercer lugar en las votaciones por el balón de oro. Para cada jugadora en una de estas posiciones, se deberá almacenar su nombre, su país, su club y el número de puntos obtenidos. Para ahorrar tiempo, no es necesario que almacene todos los años, pero al menos incluya varios de ellos (los que considere oportuno).
- (1) Crear una función que reciba una estructura del tipo anterior y el nombre de una jugadora y que devuelva un objeto con varios campos: 1) indicando si la jugadora ha estado nominada al balón de oro; 2) indicando si lo ha ganado; 3) array con los años en que ha estado nominada y el número de puntos obtenidos; 4) promedio de puntos obtenidos en sus nominaciones.
- (1) Llamar a la función anterior, pasándole la estructura creada previamente y el nombre de una jugadora indicada por el usuario (mediante el método que prefiera), y recuperar en una constante el valor devuelto por la llamada. A continuación, generar en la página los siguientes elementos, haciendo uso de funciones de manipulación del DOM: 1) El nombre de la jugadora, como cabecera de segundo nivel, y si ha ganado algún balón de oro, este nombre deberá aparecer en letra **negrita** y color de fondo **dorado** o similar; 2) Una lista no ordenada con los años en los que ha estado nominada.

**Ejercicio 4 (4,5 puntos)** Cargue en su servidor local MySQL la BD `musicales.sql`<sup>1</sup> Pruebe el archivo `ej04.php` (pruébelo en su forma original directamente en el navegador, llamando a `php` sin parámetros y luego pasándole parámetros de prueba desde el navegador). Partiendo del mismo, realice las adaptaciones necesarias para responder a las siguientes necesidades:

- a) (0,25) Disponer de una página HTML mínima que permita lanzar la consulta desde un botón de la misma, sin dinámica en cliente, de modo que su pulsación sea equivalente a haber lanzado directamente el `php` desde el navegador, y sin pasarle parámetros.
- b) (0,5) Disponer de una segunda versión de dicha página, pero reemplazando el botón anterior por uno de otro tipo que llame a PHP de forma asíncrona mediante una petición GET, sin pasarle parámetros. Recibida la respuesta de PHP, deberá plasmarla tal cuál en el cuerpo de la página HTML, al final del cuerpo del documento o en algún contenedor creado a tal efecto, lo que prefiera.
- c) (1) Desarrollar una función que reciba un número  $k$  (con valor por defecto 0) y realice una petición al servidor POST pero ahora enviándole el dato  $k$ , usando el método que prefiera. Adapte el código PHP para que pueda recibir cualquiera de los métodos vistos en clase. La respuesta recibida de la petición anterior se deberá tratar igual que en el apartado previo. Crear una nueva variante de la página HTML anterior para incluir un control de número con valores entre 0 y 10 de modo que, cada vez que su valor cambie o que dejemos de pulsar una tecla, su manejador de evento acceda a su valor y se lo pase a la función anteriormente creada.
- d) (1) Crear una nueva variante del apartado anterior pero que cambie tanto la manera de devolver los datos por parte del servidor como su tratamiento posterior en el cliente. Específicamente PHP ya no deberá generar HTML sino que deberá devolver los datos obtenidos de la consulta como JSON, de modo que en el cliente recibamos un array de objetos, cada uno representando una fila de la consulta. De vuelta en el cliente, se deberá generar una serie de párrafos al final del documento, con un texto que describa la información contenida sobre el correspondiente nominado. Además, deberá garantizarse que, al cargar la página, se visualice por primera vez el resultado de la consulta, enviando el valor 0 al servidor.
- e) (1,75) Crear una nueva versión de este ejercicio de modo que el servidor, en lugar de lo que enviaba anteriormente, ahora envíe como JSON un nuevo objeto con varios campos: 1) el primer registro devuelto por la consulta (en caso de que se devuelva alguno, un valor nulo en otro caso); 2) el número total de registros obtenido; 3) un valor lógico indicando si la consulta devolvió algún valor; 4) la lista completa de registros obtenidos. (*Continúa en página siguiente*)

---

<sup>1</sup>Agradecimientos a Magdalena Jáñez Vaz, que realizó durante el curso 2021/2022 el genial trabajo de bases de datos que hemos tomado como punto de partida para plantear este ejercicio.

Adapte el cliente para que la petición al servidor emplee una variante distinta a la que haya usado en apartados anteriores. Ahora, al procesar la respuesta del servidor, tendrá que hacer lo siguiente:

- Si el valor lógico recibido en la estructura anterior generada por el servidor es falso, y por tanto no se ha recibido ningún valor, se deberá mostrar al usuario un mensaje emergente, avisando de tal circunstancia.
- En caso contrario, se deberán generar, como en el caso de un apartado anterior, un párrafo por cada registro recibido. Además, se deberá almacenar en variables o constantes auxiliares el número de nominaciones y el número de premios del primer registro recibido.
- Además, para cada párrafo que vaya generando en el apartado anterior, se deberá asignar un manejador de evento tal que, al pasar sobre él, realice lo siguiente: comprobar si la diferencia entre su número de nominaciones y su número de premios es mayor o igual que la diferencia entre los mismos en el caso del primer registro recibido. En caso de que así sea, en dicho manejador se deberá aplicar al párrafo una clase que lo destaque del resto (por ejemplo, poniéndolo en cursiva y en un color que lo enfatice). En caso contrario, deberá aplicarle otra clase que haga lo contrario (por ejemplo, lo ponga en un gris que tienda a apagarlo).

## Notas:

- Escriba su nombre, apellidos y UVUS en la primera hoja del enunciado del examen.
- Incluya los archivos que vaya a entregar en una carpeta control6\_uvus y comprímala. La carpeta generada será lo que deba subir al servidor FileZilla.