

Apellidos:

Nombre: UVUS:



Bases de datos (GE/DGME)

11/06/2025

Final Primera Convocatoria 24/25 - Parte 2 - Control 6

Ejercicio 1 (4,75 puntos) Crear el esqueleto básico de una página HTML `ej1.html`, con el cuerpo vacío, e incorpore código dinámico en cliente para realizar lo siguiente (no tiene por qué estar todo en un mismo bloque de código):

- (0,5) Recorrer los número naturales del 1 al 10. Para cada número par, imprimir a la página el número directamente, sin más. Por cada número impar, mostrarlo en mensaje emergente. En cualquiera de los casos, añadir su cuadrado a un array.
- (0,5) Crear una estructura de datos que almacene información de los distintos cursos que tiene su grado o doble grado. De cada uno de ellos, nos interesará la lista de asignaturas, y de cada una de ellas su nombre y una lista de temas tratados en la misma. Nota: para ahorrar tiempo, no incorpore todos los cursos, asignaturas y temas, pero sí al menos varios cursos, en alguno varias asignaturas, y en alguna varios temas, lo mínimo para que se vea el tipo de estructura que crearía.
- (1,25) Crear una función `buscaTema`, que reciba únicamente una cadena de texto y busque en la estructura global anterior el primer tema cuyo nombre incluya la cadena de texto recibida como parámetro. Deberá devolver un objeto con un campo booleano que indique si se ha encontrado el tema y, en caso de haberlo hecho, incluye tres campos adicionales que indiquen el curso, el nombre de la asignatura y el tema.
- (1,75) Crear una función que no reciba nada y haga lo siguiente:
 - (0,25) Solicitar al usuario una cadena de texto y almacenarla en una constante.
 - (0,25) Llamar a la función del apartado anterior, pasándole la cadena de texto, y recuperar en una variable el valor devuelto por la llamada.
 - (1,25) Comprobar si el tema buscado devolvió el resultado esperado:
 - (0,5) Si no lo hizo, generar un párrafo al final del cuerpo del documento, con un texto que indique que encontró la cadena, incorporando explícitamente la misma dentro del texto del párrafo mediante interpolación. Poner mediante funciones de manipulación de estilo el texto en negrita y con un fondo amarillo.
 - (0,75) Si se encontró un tema incluyendo la cadena, crear en un contenedor de salida (previamente creado de forma estática en HTML) una cabecera de segundo nivel con la asignatura, y un párrafo después de dicha cabecera con el nombre del tema encontrado. Además, deberá crear una cabecera de primer nivel con el curso, explicitando que vaya justo antes del contenedor.
- (0,75) Asociar Crear un botón con un manejador asignado dinámicamente de forma tal que, al pulsar sobre el botón, realice lo siguiente: 1) llamar a la función anterior; 2) aplicar a todo párrafo de la página una clase que cambie el color del texto y le ponga un pequeño borde azul.

Ejercicio 2 (5,25 puntos) Cargue en su servidor local MySQL la BD `Cine.sql` (diseño en `Diagrama.png`).¹ Partiendo del mismo, realice las aportaciones necesarias en dinámica de cliente y servidor para responder a las siguientes necesidades:

- a) (1,25) Crear un código de servidor `ej2a.php` que se conecte a la base de datos `Cine` y lance una consulta (del tipo que desee) que devuelva todos los datos de las producciones cuyo presupuesto sea mayor al recibido en una petición GET (si no se recibe, ese dato debería ser 0). Deberá imprimir tal cual a la salida la información devuelta en el array de resultados de la consulta. Disponer de una página HTML `ej2a.html` que realice mediante JS una petición GET mínima al archivo PHP recién creado, sin pasarle parámetros. Recibida la respuesta de PHP, deberá tomar el texto e imprimirlo sin más en la propia página actual.
- b) (1,25) En una segunda página HTML `ej2b.html`, desarrollar en JS una función que tome el valor de un control de entrada de texto, creado de forma estática, y realice una petición POST a un nuevo PHP `ej2b.php`, creado como copia del anterior, pero ahora enviándole como JSON el dato de dicho control (solo debe aceptar este tipo de petición, POST recibido como JSON). La salida de PHP debe ser la misma que en el apartado anterior, pero ahora JS deberá plasmar dicha salida de PHP en un contenedor de la página, no imprimiendo directamente. Deberá lanzarse esta función cuando cambie el valor del control, así como en la carga inicial de la página.
- c) (1,25) Crear una variante del apartado anterior adaptado para que el servidor acepte cualquier método visto en clase, y para que devolvamos ahora JSON. Así, PHP ya no deberá generar HTML sino devolver los datos obtenidos de la consulta tal cual como un JSON. En el cliente, recibiremos ahora un array de objetos, cada uno representando una fila de la consulta. Deberemos generar una tabla con las tres primeras producciones devueltas, con su nombre, fecha de estreno y presupuesto.
- d) (1,5) Crear una nueva versión de este ejercicio de modo que el servidor, en lugar de lo que enviaba anteriormente, ahora realice lo siguiente: en caso de que haya o no resultados, devolver un objeto JSON con un campo lógico indicando que no tuvo éxito; en caso de que haya exactamente un resultado, devolver el primer resultado de la consulta; en caso de haber más de uno, devolver tal cual el array anterior.

Adapte el cliente para que haga lo siguiente:

- Si no tuvo éxito la consulta, indicarlo en un mensaje emergente.
- Si hay un resultado, devolverlo en la tabla, como en el apartado anterior.
- Si hay más de un resultado,

Notas:

- Si tiene cualquier duda, levante la mano y pregunte lo que necesite al profesorado presente en el aula.
- Escriba su nombre, apellidos y UVUS en la hoja del examen y folios usados.
- Incluya los archivos que vaya a entregar en una carpeta `final1_control6_uvus` y comprímala como `zip`, `rar`, `7z` o similar. El archivo generado con dicha extensión será lo que deba subir al servidor `FileZilla`, a la carpeta del servidor que se le indicará en la pizarra del examen.

¹Agradecemos al excelente trabajo de una antigua alumna, *Rosa María Carrillo García*.