

Apellidos:
Nombre: UVUS:



Bases de datos (GE/DGME)
Quinta prueba de evaluación alternativa

24/04/2024

Ejercicio 1 (1 punto) Partiendo del archivo `ejercicio1.html` que se le proporciona, adapte **mediante Bootstrap 5** para que aplique los siguientes mecanismos:

- (0,6) Incorpore los elementos y clases necesarias para que, en lugar de disponer de simples enlaces que mueven la página hacia arriba o hacia abajo, emplee alguno de los mecanismos de navegación vistos en clase (barra, pestañas o botones), de modo que siempre estemos visualizando la opción seleccionada, quedando oculto el resto.
- (0,25) Dote a la tabla del estilo de Bootstrap, incluyendo filas de colores alternos y resaltado al pasar sobre ellas.
- (0,5) Emplee los elementos necesarios para disponer los controles de formulario en filas y columnas bien alineadas, de forma que aparezca, dependiendo del tamaño del dispositivo, de la siguiente forma:

- Para dispositivos grandes:

Nombre:	<input type="text"/>	Correo Electrónico:	<input type="text"/>
Patrón	<input type="text"/>	País:	<input type="text"/>
		<input type="button" value="Insertar"/> <input type="button" value="Actualizar"/> <input type="button" value="Eliminar"/>	

- Para dispositivos tipo tablet o similar:

Nombre:	<input type="text"/>
Correo Electrónico:	<input type="text"/>
Patrón	<input type="text"/>
País:	<input type="text"/>
<input type="button" value="Insertar"/> <input type="button" value="Actualizar"/> <input type="button" value="Eliminar"/>	

- Para dispositivos más pequeños que los anteriores:

Nombre:	<input type="text"/>
Correo Electrónico:	<input type="text"/>
Patrón	<input type="text"/>
País:	<input type="text"/>
<input type="button" value="Insertar"/> <input type="button" value="Actualizar"/> <input type="button" value="Eliminar"/>	

Además, incorpore a los controles de formulario las clases adecuadas para dotarles de la apariencia de Bootstrap.

- d) (0,15) Englobe todos los elementos de la página (incluyendo navegador) en un contenedor, de modo que quede un espacio considerable a izquierda y derecha, como hemos visto de forma general en clase. Además, mediante una clase de utilidad, separe la parte de navegación de su elemento inferior, de forma que los contenidos queden separados considerablemente de las opciones de menú/pestañas/botones. Emplee un mecanismo similar para que los controles de formulario de distintas filas aparezcan con la separación mínima predefinida por Bootstrap 5 para despejarse de su elemento superior (el de la fila anterior).
- e) (0,5) Consultando la documentación proporcionada sobre Bootstrap 5, adapte los botones para que en lugar de una serie de botones aparezcan como un grupo. Con respecto a los párrafos incluidos en la página, deberán incluirse dentro de un carrusel, con un párrafo por cada entrada del carrusel.

Ejercicio 2 (2,5 puntos) Realice los siguientes ejercicios:

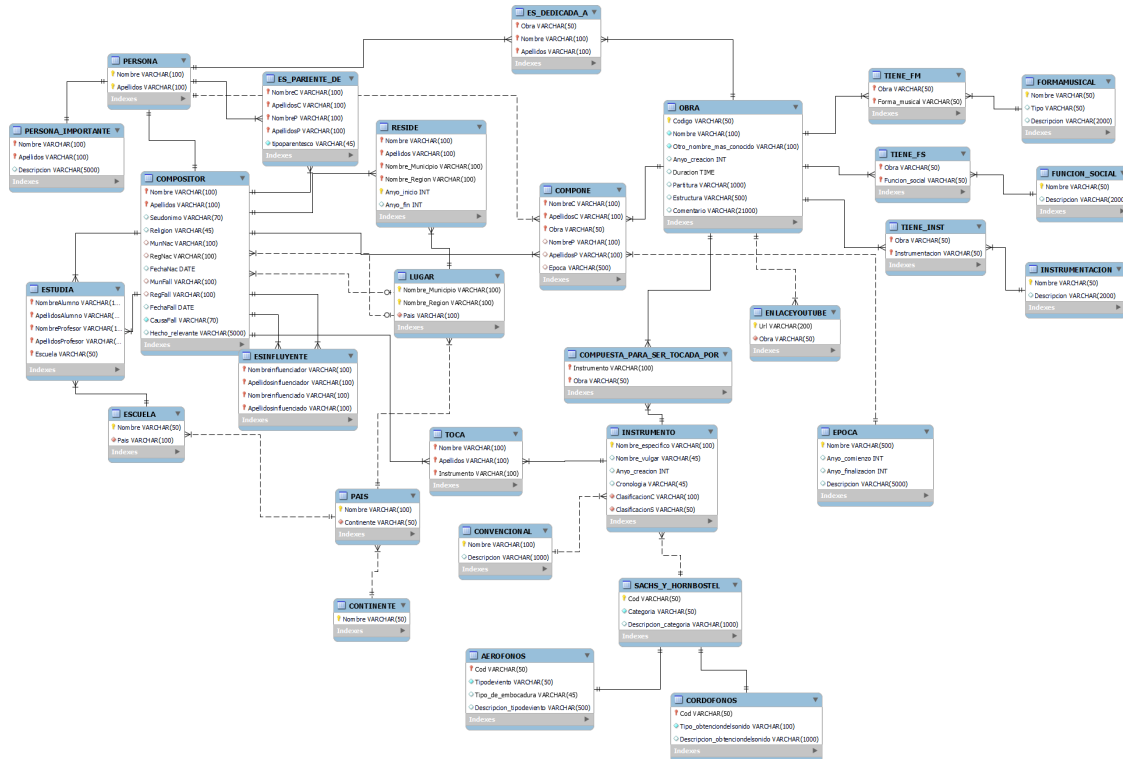
- a) (0,5) Cree un *array* con los elementos más significativos que aparecen en las cartas de las imágenes a continuación. Incluya al menos aquellos datos que se solicitan en los siguientes apartados. Añada también el continente al que pertenece el país de fabricación de cada coche.



- b) (0,5) Recorra el *array* de coches (como el creado en el apartado anterior), y **muestre** el nombre, el país de fabricación y la potencia (en CV) de cada coche (y pruebe que, efectivamente, funciona como debería).
- c) (0,75) Cree una función que reciba un *array* de coches (como el creado en el apartado anterior) y un número natural, y devuelva un *array* con todos aquellos coches cuyo número de cilindros sea **estrictamente mayor** que el número que se ha pasado por parámetros. Llame a la función con algunos valores e imprima por pantalla el *array* de resultados recibido.
- d) (0,75) Modifique la función anterior para que reciba, además del número natural, una *cadena* que indique el continente, y que devuelva el *array* que cumpla ambas condiciones. Antes de devolver el *array*, se debe **mostrar por pantalla** el promedio de cubicación de los coches seleccionados. Llame a la función e imprima el nombre del primer coche que cumpla las condiciones.

Ejercicio 3 (2,5 puntos) Realice los siguientes ejercicios. En caso de no indicar ningún método, realizarlo con el que resulte más adecuado para la ocasión.

- a) (1,25) Cree una función que se conecte a la base de datos *Compositores* a través de PHP (para ello, estará disponible el archivo *Compositores.sql*) y devuelva un *array* con todas las épocas descritas en la base de datos.



- b) (1,25) Haga una consulta a la base de datos *Compositores* y muestre una *lista* de épocas, mostrando entre paréntesis los años entre los que se desarrolló y un párrafo que contenga la información relativa al mismo (puede usar, si quiere, la función definida en el apartado anterior). Un ejemplo sería:

- Renacimiento (1450 - 1600):

La música renacentista se caracteriza por una suave sonoridad que deriva de la aceptación de la tercera como intervalo armónico consonante (uniéndose en esta categoría a quintas y octavas, ya admitidas en la Edad Media) y del progresivo aumento del número de voces, todas de igual importancia y regidas por las reglas del contrapunto: independencia de las voces, preparación y resolución de las disonancias, uso de terceras y sextas paralelas, exclusión de las quintas y octavas paralelas, etc. El prototipo de obra musical renacentista es una pieza vocal de textura polifónica, frecuentemente imitativa, escrita para entre tres y seis voces de carácter cantabile; cada línea melódica o voz podía ser interpretada indistintamente con voces reales o con instrumentos. Si bien el rango de cada línea supera apenas la octava, la extensión general del conjunto rebasa ampliamente las dos octavas, evitándose el cruce entre las voces (que forzaba a que estas fueran heterogéneas y contrastantes en la polifonía medieval)...

Ejercicio 4 (3 puntos) Partiendo de la misma base de datos anterior, cree una copia del formulario del ejercicio 1 (para no invertir tiempo en formatear los elementos básicos) y modifíquela para que contenga los datos necesarios para gestionar las épocas de la base de datos. Cree un programa básico en PHP que reciba los datos del formulario (sin usar el método por defecto) y realice distinto procesamiento según el botón pulsado:

- a) En el caso de la inserción, comprobar que se han recibido al menos el nombre y año de comienzo y posteriormente insertar, mediante una consulta preparada con parámetros nombrados, los datos en la tabla correspondiente.
- b) En el caso de la eliminación, comprobar que se ha recibido el nombre, y proceder al borrado correspondiente del registro, mediante una consulta preparada con parámetros anónimos.
- c) En el caso de la actualización, deberá recibir todos los datos del formulario y actualizar tanto los años como la descripción, para el nombre de época indicado. Realícelo mediante consulta preparada pero con el método que prefiera, aunque se valorará que sea mediante un mecanismo distinto de los anteriores.
- d) En el caso de que se produzca un error por no estar el servidor disponible, deberá enviarse un mensaje personalizado al usuario indicando tal circunstancia. Para cualquier otro error durante el proceso, deberá almacenarse en un array asociativo tanto el código de error como el mensaje recibido de la base de datos, e imprimirse esta estructura por pantalla.
- e) En cualquiera de las acciones anteriores, de no recibirse los datos requeridos para la misma, no se realizará ninguna operación.

Notas:

- Escriba su nombre, apellidos y UVUS en la hoja del examen.
- Incluya los archivos que vaya a entregar en una carpeta control5_uvus y comprímala. La carpeta generada será lo que deba subir al servidor **FileZilla**.