

PRÁCTICA 5. PHP Y LAS BASES DE DATOS

Los ejercicios de estas prácticas se dividen en dos partes. Los primeros tratan del manejo de *MySQL* a través de la consola. Por lo general en éstos se pide crear la estructura de una base de datos, poblarla con datos y luego hacer consultas sobre la base de datos. Por lo tanto se recomienda seguir este orden.

1. Diseñar sobre el papel la estructura de la base de datos.
2. Crear scripts para la definición de las tablas.
3. Crear scripts para la inicialización de las tablas.
4. Hacer operaciones sobre las tablas creadas desde la consola *MySQL*.

En la segunda parte de estos ejercicios se aprovecharán las tablas creadas en la primera parte para:

5. Hacer operaciones sobre las tablas creadas desde un programa en *PHP*.

Práctica 4.1. Crea una nueva base de datos llamada *fruteria*. Esta base de datos contiene una tabla *precios* para guardar los precios de las frutas. Ha de contener 4 campos: *id*, *fruta*, *precio_kg*, *temporada*.

Práctica 4.2. Inserta en la base de datos los siguientes precios. Como se ha comentado esto es interesante ponerlo en un script externo para poder inicializar la tabla cada vez que queramos.

- Judías - 3.50 euros/kg - temporada primavera
- Patatas - 0.40 euros/kg - temporada anual

- Tomates - 1.00 euros/kg - temporada anual
- Manzanas - 1.20 euros/kg - temporada invierno
- Uvas - 2.50 euros/kg - temporada otoño

Práctica 4.3. Usa consultas para obtener los siguientes datos:

- Muestra las frutas ordenadas por precios de menor a mayor.
- Las frutas que valen más de 1.50 euros/kg.
- Las frutas que solo se venden en otoño.
- Las frutas que están a la venta en otoño (hay que contar con las que tienen temporada anual).
- Las frutas que están a la venta todo el año y que cuestan menos de 0.50 euros/kg.

Práctica 4.4. Modifica la tabla frutas de la siguiente forma:

- Añade las frutas: melones - 0.80 euros/kg - verano, naranjas - 1.50 euros/kg - invierno.
- Elimina la fruta Manzanas.
- Cambia el precio de los melones a 0.60 euros/kg.
- Cambia la temporada de las naranjas a anual.

Práctica 4.5. En esta ocasión vamos a crear una base de datos más compleja. Llamaremos a la base de datos *baloncesto*. Contiene una tabla jugadores con los siguientes campos: *id*, *nombre*, *posicion*, *partidos*, *puntos*, *rebotes*, *asistencias*.

Práctica 4.6. Inserta en la tabla jugadores los datos de los siguientes jugadores.

- Valero - base - 24 partidos - 5.2 puntos - 1.7 rebotes - 9.8 asistencias.
- Juanfran - base - 29 partidos - 6.1 puntos - 0.8 rebotes - 5.8 asistencias.
- Montilla - escolta - 19 partidos - 11.7 puntos - 2.7 rebotes - 2.4 asistencias
- Rodriguez - escolta - 23 partidos - 17.1 puntos - 1.8 rebotes - 3.7 asistencias.

- Stipes - escolta - 31 partidos - 8.5 puntos - 3.1 rebotes - 0.9 asistencias.
- Montes - alero - 32 partidos - 13.1 puntos - 4.6 rebotes - 4.1 asistencias.
- Volkov - ala pivot - 11 partidos - 4.3 puntos - 5.6 rebotes - 1.3 asistencias.
- Suarez - ala pivot - 24 partidos - 6.9 puntos - 4.8 rebotes - 4.5 asistencias.
- Carter - ala pivot - 26 partidos - 26.1 puntos - 9.1 rebotes - 1.8 asistencias
- Graham - pivot - 17 partidos - 2.1 puntos - 8.4 rebotes - 0.2 asistencias.
- Cesar - pivot - 8 partidos - 3.1 puntos - 6.8 rebotes - 0.7 asistencias.

Práctica 4.7. Extrae estadísticas de los jugadores:

- Listas de los jugadores ordenados de más a menos en cuanto a partidos jugados, puntos, rebotes y asistencias (separar los campos a ordenar con comas).
- Listas de:
 - Jugadores que han anotado más de 12 puntos.
 - Jugadores que han cogido más de 6 rebotes.
 - Jugadores que han dado más de 5 asistencias.
- Un jugador se considera completo si es capaz de anotar más de 10 puntos, coger 4 rebotes y dar cuatro asistencias por partido. Averigua si alguno de los jugadores del equipo lo es.
- Veamos los jugadores que destacan en su posición:
 - Bases que dan más de 8 asistencias por partido.
 - Escoltas o aleros que anotan más de 15 puntos por partido.
 - Ala pivots o pivots que cogen más de 7 rebotes por partido.

Práctica 4.8. La temporada siguiente hay bajas y fichajes. Actualiza la información de la base de datos de forma acorde.

- Juanfran y Carter son baja

- Se ha fichado a Jofre - alero, Lehman - ala pivot y a Stevenson - Pivot. Sus estadísticas iniciales están a 0.

Práctica 4.9. Usando la base de datos *fruteria*, hacer una página web que muestre una lista de los productos que están a la venta, "*listar_fruta.php*".

Práctica 4.10. Hacer una página web que muestre las estadísticas del equipo de baloncesto. Ha de estar compuesta por las siguientes páginas.

- equipo.php: listado de los nombres de los jugadores ordenados por posiciones de juego. Cada nombre tiene un enlace a ficha.php
- ficha.php: la ficha de un jugador con sus estadísticas personales.
- partidos.php: muestra un listado de los jugadores ordenados por partidos jugados.
- puntos.php: muestra un listado de los jugadores ordenados de por puntos anotados por partido.
- rebotes.php: muestra un listado de los jugadores ordenados de por rebotes capturados por partido.
- asistencias.php: muestra un listado de los jugadores ordenados de por asistencias dadas por partido.

Práctica 4.11. Estaría bien que se pudieran gestionar las fichas desde la propia web.

Para ello hay que agregar los formularios:

- nueva_ficha.php: formulario para añadir un jugador y sus datos.
- guardar_ficha.php: el script que guarda los datos que provienen del script anterior.
- borrar_ficha.php: el script que borra la ficha de un jugador.

Práctica 4.12. Mejorar la aplicación 4.9 para que se puedan gestionar datos.

- Añadir un formulario que permita añadir frutas, "*nueva_fruta.php*". Y una página que lo inserte en la base de datos, "*guardar_fruta.php*".
- Programar también un sistema para que se puedan borrar frutas del catálogo, "*borrar_fruta.php*".

- Poner un último formulario que permita modificar el precio de una fruta. Primero se han de leer los datos actuales, mostrar en el formulario (*editar_fruta.php*), modificar y guardarlos en la base de datos mediante un UPDATE (*mod_fruta.php*).

Práctica 4.13. Este ejercicio tiene una dificultad notablemente mayor que los anteriores y pone en juego varios de los conocimientos adquiridos hasta el momento.

Se podría añadir una mejora a la página del equipo de baloncesto que consistiera en un formulario para subir el acta de un partido, y que la aplicación recogiera ese acta, y actualizara las estadísticas de los jugadores.

El acta es un archivo que contiene una línea por cada jugador que ha participado con los siguientes datos (los jugadores que no figuran en el acta es porque no han salido al campo):

nombre del jugador, puntos, rebotes, asistencias;

Un fichero de acta podría ser el siguiente:

<pre>Valero, 17, 6, 4; Juanfran, 8, 1, 11; Montilla, 12, 4, 2; Volkov, 7, 7, 0; Carter, 2, 4, 0; Stipes, 23, 0, 3; Cesar, 9, 2, 2; Montes, 11, 3, 1;</pre>
--

Este proceso es algo complicado. Vamos a detallar los pasos que vamos a seguir.

- Un formulario "*subir_acta.php*" que permite subir un archivo con un acta.
- El destino de la página "*subir_acta.php*" es la página "*recoger_acta.php*". Ésta realiza todo el trabajo.
 - Primero guarda el archivo temporal en una carpeta actas.
 - Segundo abre el archivo con el acta y lee los datos y los guarda en variables.
 - Por cada jugador que aparece en el acta lee los datos de la base de datos.
 - Calcula la estadística actualizada para cada categoría. Por ejemplo:

<?php

```
// Código anterior...
$partidos_ahora = $partidos_antes + 1;
$puntos_antes = $puntos_por_partido_antes * $partidos_antes;
$puntos_ahora = $puntos_antes + $puntos_ultimo_partido;
$puntos_por_partido_ahora = $puntos_ahora / $partidos_ahora;
// Código posterior...
?>
```

- Finalmente actualiza los datos de cada jugador en la base de datos.

Es altamente recomendable usar arrays asociativos para almacenar los datos y extraer el código que se usa varias veces a funciones para que el código quede más limpio y ordenado.