

1. Creación de la Base de Datos y Tablas

Objetivo: Diseñar la estructura de una base de datos para una tienda de artículos deportivos.

Tareas:

1. **Crear la base de datos** llamada `tienda_deportiva`.
 2. **Definir las tablas** con los siguientes campos:
 - **Categorías**(`id_categoria` , `nombre` , `descripcion`)
 - **Productos**(`id_producto` , `nombre` , `descripcion` , `precio` , `stock` , `id_categoria` , `marca` , `fecha_creacion`)
 - **Clientes**(`id_cliente` , `nombre` , `apellidos` , `email` , `telefono` , `fecha_registro` , `puntos_fidelidad`)
 - **Pedidos**(`id_pedido` , `id_cliente` , `fecha_pedido` , `estado` , `total` , `metodo_pago`)
 - **Detalles_Pedido**(`id_detalle` , `id_pedido` , `id_producto` , `cantidad` , `precio_unitario`)
 3. **Establecer las relaciones** entre tablas mediante claves foráneas.
-

2. Consultas Simples

Objetivo: Realizar consultas básicas para extraer información específica.

Ejercicios:

1. Mostrar **todos los productos** de la tienda.
 2. Listar **nombre y precio** de los productos con **stock mayor a 10 unidades**.
 3. Contar el **número total de clientes** registrados.
 4. Mostrar los **pedidos realizados hoy**.
 5. Ordenar los productos por **precio (de mayor a menor)**.
-

3. Consultas con Composición Interna (INNER JOIN)

Objetivo: Combinar datos de varias tablas relacionadas.

Ejercicios:

1. Mostrar **productos con su categoría** correspondiente.
 2. Listar **detalles de pedidos** incluyendo **nombre del producto** comprado.
 3. Mostrar **pedidos con información del cliente** (nombre, apellidos).
 4. Identificar **productos vendidos en pedidos ya entregados**.
 5. Encontrar **clientes que han comprado productos de la categoría "running"**.
-

4. Consultas con Composición Externa (OUTER JOIN)

Objetivo: Incluir registros aunque no tengan coincidencias en otras tablas.

Ejercicios:

1. Mostrar **todas las categorías**, incluso las **sin productos asociados**.
 2. Listar **clientes que nunca han realizado un pedido**.
 3. Identificar **productos que nunca han sido vendidos**.
 4. Mostrar **todos los pedidos**, incluyendo clientes (aunque no debería haber pedidos sin cliente).
 5. Listar **productos y sus categorías**, mostrando también **productos sin categoría asignada**.
-

5. Consultas Resumen (GROUP BY)

Objetivo: Agrupar datos para obtener estadísticas.

Ejercicios:

1. Calcular el **total de ventas por categoría** (solo pedidos entregados).
 2. Contar el **número de pedidos por cliente** y su **total gastado**.
 3. Calcular el **promedio de gasto por pedido** (entregados).
 4. Identificar los **5 productos más vendidos** (por cantidad).
 5. Generar un **informe de ventas mensuales**.
-

6. Consultas con Subconsultas

Objetivo: Usar consultas anidadas para filtrar datos complejos.

Ejercicios:

1. Listar **productos con precio superior al promedio**.
2. Identificar **clientes que han gastado más que el promedio**.

3. Mostrar **productos nunca vendidos** (usando `NOT IN`).
 4. Encontrar **categorías con menos productos que la media**.
 5. Listar **pedidos con valor superior al 90% de los pedidos**.
-

7. Consultas con Múltiples Selecciones (UNION, INTERSECT, EXCEPT)

Objetivo: Combinar resultados de varias consultas.

Ejercicios:

1. Combinar **productos de las categorías "running" y "fitness"** (usando `UNION`).
 2. Identificar **clientes que han comprado tanto en "running" como "fitness"** (simular `INTERSECT`).
 3. Mostrar **productos de "running" que no están en "fitness"** (simular `EXCEPT`).
 4. Combinar **productos con stock bajo y productos con precio alto** en un solo listado (`UNION ALL`).
-

8. Optimización de Consultas

Objetivo: Mejorar el rendimiento de las consultas.

Ejercicios:

1. **Crear índices** en campos de búsqueda frecuente (`nombre` , `id_categoria` , `fecha_pedido`).
2. Reescribir una consulta usando `EXISTS` en lugar de `IN` para mejorar rendimiento.
3. **Evitar `SELECT *`** y especificar solo columnas necesarias.
4. Comparar dos versiones de una consulta (con `JOIN` vs. subconsulta) y analizar cuál es más eficiente.
5. Usar `EXPLAIN ANALYZE` para evaluar el rendimiento de una consulta compleja.