

Tema 1. Fundamentos de bases de datos

Cuestionario

1. ¿Qué es una base de datos?

- a) Un conjunto desorganizado de datos.
- b) Un sistema para procesar datos en bruto.
- c) Un conjunto organizado de datos que se almacenan y gestionan electrónicamente.
- d) Un sistema exclusivamente diseñado para redes sociales

2. ¿Cuál de las siguientes opciones describe mejor "datos"?

- a) Información procesada y útil.
- b) Hechos sin procesar.
- c) Resultados de un análisis.
- d) Datos organizados en tablas.

3. ¿Cuál es un ejemplo de información?

- a) "Cliente A realizó una compra de 25 unidades el 1 de enero de 2025".
- b) "25".
- c) "2025-01-01".
- d) "Cliente A".

4. ¿Qué tipo de base de datos organiza datos en tablas con filas y columnas?

- a) Relacional.
- b) No relacional.
- c) Jerárquica.
- d) En red.

5. ¿Cuál de las siguientes NO es una característica de bases de datos no relacionales?

- a) Uso de modelos clave-valor.
- b) Uso de SQL para consultas.
- c) Almacenamiento de datos semiestructurados.
- d) Alta escalabilidad horizontal.

6. ¿Cuál es un ejemplo de una base de datos relacional?

- a) MongoDB.
- b) Redis.
- c) PostgreSQL.
- d) Cassandra.

7. ¿Qué tipo de bases de datos está optimizada para transacciones en tiempo real?

- a) Analíticas.
- b) Transaccionales.
- c) Jerárquicas.
- d) Orientadas a objetos.

8. ¿Cuál es un ejemplo de un caso de uso para bases de datos analíticas?

- a) Procesamiento de pagos.
- b) Gestión de inventarios.
- c) Generación de dashboards.
- d) Registro de transacciones bancarias.

9. ¿Qué estructura se utiliza en las bases de datos jerárquicas?

- a) Árbol.
- b) Grafos.
- c) Tablas.
- d) Documentos JSON.

10. ¿Qué componente en una base relacional garantiza la unicidad de cada registro?

- a) Clave primaria.
- b) Clave foránea.
- c) Registro.
- d) Tabla.

11. ¿Qué componente en bases no relacionales es equivalente a una fila en una base relacional?

- a) Documento.
- b) Colección.
- c) Clave.
- d) Valor.

12. ¿Qué base de datos utiliza estructuras clave-valor?

- a) MySQL.
- b) Redis.
- c) Oracle.
- d) PostgreSQL.

13. ¿Qué significa OLTP?

- a) Procesamiento de transacciones analíticas.
- b) Procesamiento de transacciones en línea.
- c) Optimización de consultas analíticas.
- d) Organización de tablas y procesos.

14. ¿Qué es una colección en bases de datos no relacionales?

- a) Una agrupación de tablas.
- b) Una agrupación de documentos.
- c) Una clave única.
- d) Una transacción.

15. ¿Qué base de datos es más adecuada para redes sociales?

- a) Oracle.
- b) MySQL.
- c) MongoDB.
- d) PostgreSQL.

16. ¿Qué tipo de base de datos combina conceptos de programación orientada a objetos?

- a) Relacional.
- b) En red.
- c) Orientada a objetos.
- d) No relacional.

17. ¿Qué componente de bases relacionales conecta tablas mediante relaciones?

- a) Clave primaria.
- b) Clave foránea.
- c) Registro.
- d) Columna.

18. ¿Qué tipo de bases de datos utiliza JSON para almacenar datos?

- a) Relacionales.
- b) Jerárquicas.
- c) No relacionales.
- d) En red.

19. ¿Qué base de datos sería más adecuada para CAD/CAM?

- a) PostgreSQL.
- b) MongoDB.
- c) Una base orientada a objetos.
- d) Redis.

20. ¿Qué sistema se utiliza para realizar consultas en bases relacionales?

- a) JSON.
- b) SQL.
- c) BSON.
- d) API REST.

21. ¿Qué base de datos está diseñada para consultas complejas en grandes volúmenes de datos históricos?

- a) OLTP.
- b) OLAP.
- c) Redis.
- d) MongoDB.

22. ¿Qué ventaja tienen las bases no relacionales frente a las relacionales?

- a) Mejor manejo de datos estructurados.
- b) Escalabilidad horizontal.
- c) Mayor control de integridad referencial.
- d) Simplicidad en la modelación.

23. ¿Qué modelo de base de datos utiliza nodos con múltiples relaciones?

- a) Jerárquico.
- b) En red.
- c) Relacional.
- d) Transaccional.

24. ¿Cuál de las siguientes es una característica de las bases de datos relacionales?

- a) Manejan datos en grafos.
- b) Están optimizadas para datos clave-valor.
- c) Garantizan la integridad referencial.
- d) Usan JSON nativamente.

25. ¿Qué tipo de datos es ideal para una base de datos jerárquica?

- a) Relaciones complejas.
- b) Datos estructurados en árbol.
- c) Datos en JSON.
- d) Análisis histórico.

26. ¿Qué es un registro en bases de datos relacionales?

- a) Una fila en una tabla.
- b) Una columna en una tabla.
- c) Una clave primaria.
- d) Un atributo.

27. ¿Cuál es el objetivo principal de una base de datos analítica?

- a) Registrar transacciones.
- b) Generar informes y análisis.
- c) Proporcionar escalabilidad horizontal.
- d) Gestionar catálogos de productos.

28. ¿Cuál de las siguientes NO es una base de datos relacional?

- a) MySQL.
- b) PostgreSQL.
- c) MongoDB.
- d) Oracle.

29. ¿Qué tipo de bases de datos permite almacenar datos estructurados, semiestructurados y no estructurados?

- a) Relacionales.
- b) No relacionales.
- c) Jerárquicas.
- d) En red.

30. ¿Qué tipo de clave conecta tablas en bases de datos relacionales?

- a) Clave primaria.
- b) Clave compuesta.
- c) Clave foránea.
- d) Clave única.

31. ¿Qué tipo de dato se utiliza para almacenar valores de texto de longitud fija?

- a) VARCHAR
- b) CHAR
- c) TEXT
- d) BLOB

32. ¿Cuál de los siguientes tipos de datos es adecuado para almacenar números decimales con precisión exacta?

- a) FLOAT
- b) INTEGER
- c) DECIMAL
- d) BOOLEAN

33. ¿Qué tipo de dato sería más apropiado para almacenar una imagen?

- a) VARCHAR
- b) BLOB
- c) BOOLEAN
- d) TEXT

34. ¿Qué tipo de dato se usa comúnmente para almacenar valores de verdadero o falso?

- a) BOOLEAN
- b) INTEGER
- c) CHAR
- d) DATE

35. ¿Cuál de los siguientes no es un tipo de dato numérico?

- a) FLOAT
- b) INT
- c) TEXT
- d) DECIMAL

36. Si necesitas almacenar una fecha y una hora, ¿qué tipo de dato es el más adecuado?

- a) DATE
- b) TIME
- c) DATETIME
- d) TIMESTAMP

37. ¿Qué diferencia principal hay entre VARCHAR y TEXT?

- a) TEXT tiene una longitud fija, mientras que VARCHAR es variable.
- b) TEXT puede almacenar más datos que VARCHAR.
- c) VARCHAR es para datos binarios, mientras que TEXT es para texto.
- d) TEXT no permite índices, mientras que VARCHAR sí.

38. ¿Qué tipo de dato utilizarías para almacenar un número de identificación muy grande?

- a) SMALLINT
- b) INT
- c) BIGINT
- d) FLOAT

39. ¿Qué tipo de dato sería más adecuado para almacenar un precio, como 19.99?

- a) FLOAT
- b) DECIMAL
- c) BOOLEAN
- d) INT

40. ¿Qué tipo de dato se recomienda para almacenar datos estructurados en formato JSON?

- a) TEXT
- b) BLOB
- c) JSON
- d) VARCHAR

SOLUCIONES

1 - C	11 - A	21 - B	31 - B
2 - B	12 - B	22 - B	32 - C
3 - A	13 - B	23 - B	33 - B
4 - A	14 - B	24 - C	34 - A
5 - B	15 - C	25 - B	35 - C
6 - C	16 - C	26 - A	36 - C
7 - B	17 - B	27 - B	37 - B
8 - C	18 - C	28 - C	38 - C
9 - A	19 - C	29 - B	39 - B
10 - A	20 - B	30 - C	40 - C

Tema 2. Modelado de bases de datos

Cuestionario

1. ¿Qué es la normalización en bases de datos?

- a) Un proceso para eliminar datos duplicados y garantizar integridad.
- b) Una técnica para mejorar el rendimiento de las consultas SQL.
- c) Un modelo de bases de datos para datos no estructurados.
- d) Un método para realizar copias de seguridad.

2. ¿Qué nivel de normalización exige que los valores sean atómicos y sin grupos repetidos?

- a) Segunda forma normal (2FN).
- b) Primera forma normal (1FN).
- c) Tercera forma normal (3FN).
- d) Cuarta forma normal (4FN).

3. En un diagrama entidad-relación (ER), ¿qué representa un rectángulo?

- a) Una entidad.
- b) Un atributo.
- c) Una relación.
- d) Una clave primaria.

4. ¿Qué es una clave primaria?

- a) Un atributo que referencia a otra tabla.
- b) Un identificador único para cada registro en una tabla.
- c) Un atributo opcional en una tabla.
- d) Un conjunto de datos duplicados.

5. ¿Qué tipo de relación se establece cuando un registro de una tabla puede estar asociado a múltiples registros de otra tabla, pero un registro de la segunda tabla solo puede estar relacionado con uno de la primera?

- a) Uno a uno (1:1).
- b) Uno a muchos (1:N).
- c) Muchos a muchos (N:M).
- d) Jerárquico.

6. ¿Qué representa un rombo en un diagrama ER?

- a) Una entidad.
- b) Un atributo.
- c) Una relación.
- d) Un conjunto de claves foráneas.

7. ¿Cuál es el propósito de las claves foráneas?

- a) Asegurar que los datos sean únicos en una tabla.
- b) Establecer relaciones entre tablas.
- c) Eliminar datos redundantes en una base de datos.
- d) Representar atributos calculados.

8. En un modelo relacional, ¿qué describe una columna de una tabla?

- a) Una entidad.
- b) Un registro.
- c) Un atributo.
- d) Una clave compuesta.

9. ¿Cuál es un ejemplo de un atributo derivado?

- a) Fecha de nacimiento.
- b) Edad calculada a partir de la fecha de nacimiento.
- c) Nombre completo.
- d) Dirección de correo electrónico.

10. ¿Qué término se utiliza para describir una relación que conecta una tabla consigo misma?

- a) Relación autorreferencial.
- b) Relación jerárquica.
- c) Relación uno a muchos.
- d) Relación recursiva.

11. ¿Qué se necesita para que una base de datos alcance la tercera forma normal (3FN)?

- a) Que elimine valores atómicos en los datos.
- b) Que elimine dependencias transitivas entre los atributos.
- c) Que no tenga claves foráneas.
- d) Que utilice únicamente relaciones de tipo 1:1.

12. ¿Qué tipo de atributo suele usarse para representar una clave primaria?

- a) Atributo opcional.
- b) Atributo derivado.
- c) Atributo único.
- d) Atributo compuesto.

13. ¿Cuál de las siguientes relaciones no es válida en un modelo relacional?

- a) Uno a muchos (1:N).
- b) Muchos a muchos (N:M).
- c) Uno a uno (1:1).
- d) Ninguna de las anteriores.

14. En un modelo entidad-relación, ¿qué se utiliza para representar un atributo opcional?

- a) Un rectángulo con líneas dobles.
- b) Un rombo con un subrayado.
- c) Un óvalo con una línea discontinua.
- d) Un óvalo con una línea continua.

15. ¿Qué es una tabla intermedia en un modelo relacional?

- a) Una tabla que no tiene claves primarias.
- b) Una tabla que se utiliza para resolver relaciones muchos a muchos (N:M).
- c) Una tabla que almacena solo datos históricos.
- d) Una tabla que contiene solo atributos calculados.

16. ¿Cuál es el propósito de una cardinalidad en un modelo entidad-relación?

- a) Determinar el tipo de datos que puede tener un atributo.
- b) Especificar el número máximo de registros que puede tener una tabla.
- c) Indicar el número de instancias de una entidad que se relacionan con otra.
- d) Identificar claves únicas en una tabla.

17. ¿Qué representa una línea doble entre dos entidades en un diagrama ER?

- a) Una relación opcional.
- b) Una relación obligatoria.
- c) Una relación autorreferencial.
- d) Una clave primaria compuesta.

SOLUCIONES**1 - A****2 - B****3 - A****4 - B****5 - B****6 - C****7 - B****8 - C****9 - B****10 - D****11 - B****12 - C****13 - D****14 - C****15 - B****16 - C****17 - B**

Tema 3. Introducción a Sistemas de Gestión de Bases de Datos (SGBD)

Cuestionario

1. ¿Qué es un Sistema de Gestión de Bases de Datos (SGBD)?

- a) Un software que permite gestionar bases de datos de manera eficiente.
- b) Un lenguaje de programación para crear bases de datos.
- c) Un hardware para almacenar datos de forma física.
- d) Un protocolo de comunicación entre servidores.

2. ¿Cuál es una de las principales funciones de un SGBD?

- a) Diseñar interfaces gráficas de usuario.
- b) Garantizar la integridad y seguridad de los datos.
- c) Convertir datos no estructurados en gráficos.
- d) Ejecutar código en múltiples lenguajes de programación.

3. ¿Cuál de los siguientes es un ejemplo de SGBD relacional?

- a) MongoDB
- b) Cassandra
- c) MySQL
- d) Redis

4. ¿Cómo interactúa el SGBD con las aplicaciones?

- a) Envía datos directamente a los usuarios sin intermediarios.
- b) Procesa consultas y devuelve resultados a las aplicaciones.
- c) Almacena datos temporalmente en la memoria RAM.
- d) No interactúa con aplicaciones, solo con administradores.

5. ¿Qué lenguaje suelen utilizar los SGBD relacionales para manejar datos?

- a) HTML
- b) Python
- c) SQL
- d) JavaScript

6. ¿Cuál de los siguientes NO es un SGBD relacional?

- a) PostgreSQL
- b) Oracle
- c) MongoDB
- d) MySQL

7. ¿Qué diferencia principal tienen los sistemas no relacionales frente a los relacionales?

- a) Los sistemas no relacionales no utilizan tablas para almacenar datos.
- b) Los sistemas no relacionales solo manejan datos estructurados.
- c) Los sistemas no relacionales no permiten escalar.
- d) Los sistemas no relacionales son exclusivos para pequeñas aplicaciones.

8. ¿Cuál de los siguientes sistemas utiliza un modelo clave-valor?

- a) Cassandra
- b) MongoDB
- c) Redis
- d) PostgreSQL

9. ¿Qué característica define mejor a los sistemas relacionales?

- a) Almacenamiento en documentos JSON.
- b) Alta consistencia y cumplimiento de las propiedades ACID.
- c) Escalabilidad horizontal por diseño.
- d) Ausencia de un esquema fijo.

10. ¿Para qué tipo de aplicaciones es más adecuado un sistema no relacional?

- a) Sistemas financieros que requieren alta consistencia.
- b) Redes sociales con grandes volúmenes de datos no estructurados.
- c) Aplicaciones empresariales con relaciones complejas entre datos.
- d) Sistemas de facturación con tablas estructuradas.

SOLUCIONES

1 - A

2 - B

3 - C

4 - B

5 - C

6 - C

7 - A

8 - C

9 - B

10 - B

Tema 4. Introducción a SQL

Cuestionario

1. ¿Qué comando se utiliza para recuperar datos de una tabla en SQL?

- a) INSERT
- b) SELECT
- c) UPDATE
- d) DELETE

2. ¿Qué comando se usa para agregar un nuevo registro en una tabla?

- a) SELECT
- b) INSERT
- c) UPDATE
- d) DELETE

3. ¿Cuál es la función del comando UPDATE en SQL?

- a) Insertar datos en una tabla
- b) Eliminar registros de una tabla
- c) Modificar datos existentes en una tabla
- d) Crear una nueva tabla

4. ¿Para qué se usa el comando DELETE en SQL?

- a) Para borrar una tabla completa
- b) Para eliminar registros específicos de una tabla
- c) Para actualizar datos
- d) Para agregar una nueva columna

5. ¿Cómo se pueden filtrar los resultados de una consulta en SQL?

- a) Con el comando WHERE
- b) Con el comando ORDER BY
- c) Con el comando INSERT
- d) Con el comando GROUP BY

6. ¿Qué comando permite ordenar los resultados de una consulta?

- a) WHERE
- b) ORDER BY
- c) GROUP BY
- d) DELETE

7. ¿Qué palabra clave se utiliza para agrupar registros en SQL?

- a) ORDER BY
- b) GROUP BY
- c) INSERT
- d) UPDATE

8. ¿Cuál de las siguientes funciones permite contar el número de registros en una consulta?

- a) COUNT
- b) SUM
- c) AVG
- d) ORDER BY

9. ¿Qué función se usa para calcular la suma de valores en una columna?

- a) COUNT
- b) SUM
- c) AVG
- d) DELETE

10. ¿Qué función devuelve el promedio de los valores en una columna?

- a) COUNT
- b) SUM
- c) AVG
- d) UPDATE

11. Si quieres seleccionar todos los registros de una tabla llamada "clientes", ¿cuál sería la consulta correcta?

- a) SELECT clientes FROM *;
- b) SELECT ALL FROM clientes;
- c) SELECT * FROM clientes;
- d) SELECT FROM clientes ALL;

12. ¿Qué cláusula se usa para especificar la condición de filtrado en una consulta SQL?

- a) ORDER BY
- b) GROUP BY
- c) WHERE
- d) INSERT

13. ¿Cuál de las siguientes opciones es una consulta SQL válida?

- a) INSERT INTO usuarios (nombre, edad) VALUES ('Ana', 25);
- b) INSERT usuarios SET nombre='Ana', edad=25;
- c) ADD INTO usuarios (nombre, edad) VALUES ('Ana', 25);
- d) PUSH INTO usuarios (nombre, edad) VALUES ('Ana', 25);

14. ¿Cuál es el resultado de la siguiente consulta?

SELECT COUNT(*) FROM empleados;

- a) La suma de los salarios de todos los empleados
- b) El número total de empleados en la tabla
- c) La media de los salarios de los empleados
- d) Un listado con todos los empleados

15. ¿Qué consulta en SQL se usaría para actualizar el salario de un empleado con id=5 a 3000?

- a) UPDATE empleados SET salario=3000 WHERE id=5;
- b) MODIFY empleados SET salario=3000 WHERE id=5;
- c) CHANGE empleados SET salario=3000 WHERE id=5;
- d) SET empleados UPDATE salario=3000 WHERE id=5;

16. ¿Cuál es el propósito de la cláusula GROUP BY en SQL?

- a) Ordenar los datos de la consulta
- b) Filtrar los resultados según una condición
- c) Agrupar filas que tienen valores similares en columnas específicas
- d) Insertar registros en una tabla

17. ¿Qué consulta se usaría para eliminar todos los registros de la tabla "productos"?

- a) DELETE FROM productos;
- b) REMOVE FROM productos;
- c) DROP FROM productos;
- d) ERASE FROM productos;

18. ¿Qué sucede si se ejecuta la consulta DELETE FROM empleados WHERE id=10;?

- a) Se eliminan todos los empleados
- b) Se elimina el empleado con id=10
- c) Se elimina la columna id
- d) Se borra la tabla empleados

19. ¿Cómo se pueden obtener los empleados cuyo salario sea mayor a 2000?

- a) SELECT * FROM empleados WHERE salario > 2000;
- b) SELECT * FROM empleados ORDER BY salario > 2000;
- c) SELECT * FROM empleados GROUP BY salario > 2000;
- d) SELECT * FROM empleados FILTER salario > 2000;

20. ¿Qué comando se usa para insertar múltiples filas en una tabla en una sola consulta?

- a) INSERT INTO tabla VALUES (valor1), (valor2), (valor3);
- b) INSERT tabla VALUES (valor1), (valor2), (valor3);
- c) ADD INTO tabla VALUES (valor1), (valor2), (valor3);
- d) PUSH INTO tabla VALUES (valor1), (valor2), (valor3);

SOLUCIONES**1 - B****11 - C****2 - B****12 - C****3 - C****13 - A****4 - B****14 - B****5 - A****15 - A****6 - B****16 - C****7 - B****17 - A****8 - A****18 - B****9 - B****19 - A****10 - C****20 - A**

Tema 4. Introducción a SQL

Ejercicios prácticos

Ejercicio Práctico: Base de Datos para una Librería

La librería necesita una base de datos para gestionar los libros en su almacén.

- a) Crea una base de datos con el nombre "librería"
- b) Define una tabla llamada libros con los siguientes campos:
 - id_libro (Identificador único del libro, clave primaria)
 - titulo (Título del libro)
 - autor (Nombre del autor)
 - genero (Categoría del libro, ej. "Ficción", "Historia", "Ciencia")
 - precio (Precio del libro)
 - stock (Cantidad disponible en almacén)
 - fecha_publicacion (Fecha en que se publicó el libro)

Escribe la consulta SQL para crear esta tabla y decide qué tipo de datos es adecuado para cada columna.

- c) Inserta al menos cinco libros en la tabla. Escribe una consulta INSERT INTO para agregar libros a la tabla.
- d) Consulta para mostrar todos los libros almacenados.
- e) Consulta para buscar un libro por título
- f) Consulta para saber el número total de libros.
- g) Mostrar libros de género específico
- h) Ordenar libros por precio (de mayor a menor)
- i) Mostrar cuántos libros hay por cada género ordenados alfabéticamente.

Desafío adicional

- j) Agrega una nueva columna a la tabla para incluir la editorial del libro y actualiza los datos de los libros insertados.

Soluciones Base de Datos para una Librería**a) Crea una base de datos con el nombre “librería”**

```
CREATE DATABASE libreria;  
USE libreria;  (Para usar la base de datos librería)
```

b) Define una tabla llamada libros:

```
CREATE TABLE libros (  
id_libro INT AUTO_INCREMENT PRIMARY KEY, -- Clave primaria autoincremental  
titulo VARCHAR(255) NOT NULL, -- Título del libro, obligatorio  
autor VARCHAR(255) NOT NULL, -- Autor del libro, obligatorio  
genero VARCHAR(100), -- Género del libro, opcional  
precio DECIMAL(10,2), -- Precio con dos decimales  
stock INT DEFAULT 0, -- Cantidad en almacén, por defecto 0  
fecha_publicacion DATE -- Fecha en formato AAAA-MM-DD  
);
```

c) Insertar al menos 5 registros en la tabla:

```
INSERT INTO libros (titulo, autor, genero, precio, stock, fecha_publicacion) VALUES  
( 'Cien años de soledad', 'Gabriel García Márquez', 'Ficción', 19.99, 10, '1967-06-05'),  
( '1984', 'George Orwell', 'Ciencia Ficción', 14.50, 15, '1949-06-08'),  
( 'El nombre del viento', 'Patrick Rothfuss', 'Fantasía', 25.00, 5, '2007-03-27'),  
( 'Sapiens', 'Yuval Noah Harari', 'Historia', 21.99, 8, '2011-02-04'),  
( 'El código Da Vinci', 'Dan Brown', 'Thriller', 17.95, 12, '2003-03-18')  
);
```

d) Mostrar todos los libros de la base de datos:

```
SELECT * FROM libros;
```

e) Buscar libro por título:

```
SELECT * FROM libros WHERE titulo = 'Cien años de soledad';
```

f) Conocer número total de libros:

```
SELECT COUNT(*) FROM libros;
```

Esta consulta devuelve el número total de títulos. Para saber el número total de libros que tenemos en el almacén (en stock) utilizaríamos la siguiente consulta:

```
SELECT SUM(stock) FROM libros;
```

g) Mostrar libros de género específico:

```
SELECT * FROM libros WHERE genero = 'Historia';
```

h) Ordenar los libros por precio de mayor a menor:

```
SELECT * FROM libros ORDER BY precio DESC;
```

i) Mostrar cuantos libros hay por cada género ordenados alfabéticamente:

```
SELECT genero, COUNT(*) AS cantidad_libros
FROM libros
GROUP BY genero
ORDER BY genero;
```

*AS sirve para dar nombre al campo de la columna calculada que nos muestra el conteo de libros para cada género.

Solución desafío adicional:**j) Modificar la tabla y agregar columna con campo editorial. Agregar la editorial a los registros de la tabla:**

```
ALTER TABLE libros ADD COLUMN editorial VARCHAR(100);
```

```
UPDATE libros SET editorial = 'Penguin Random House' WHERE titulo = 'Cien años de soledad';
```

```
UPDATE libros SET editorial = 'Secker & Warburg' WHERE titulo = '1984';
```

```
UPDATE libros SET editorial = 'DAW Books' WHERE titulo = 'El nombre del viento';
```

```
UPDATE libros SET editorial = 'Debate' WHERE titulo = 'Sapiens';
```

```
UPDATE libros SET editorial = 'Planeta' WHERE titulo = 'El código Da Vinci';
```

*ALTER TABLE es el comando que se utiliza para modificar la tabla, en este caso para agregar una nueva columna con el comando ADD COLUMN.

Ejercicio Práctico: Base de Datos para una tienda de tecnología

Una tienda de tecnología necesita una base de datos para gestionar sus productos en almacén. La base de datos debe contener información sobre los productos, como su nombre, categoría, precio, stock disponible y fecha de incorporación.

- a) Crea una base de datos con el nombre "tecnología".
- b) Define una tabla llamada productos.
- c) Inserta al menos cinco productos en la tabla.
- d) Consulta para contar número total de productos.
- e) Consulta para calcular el numero total de stock en la tienda de todos los productos.
- f) Ordenar los productos por precio de mayor a menor.
- g) Listar productos por orden de entrada.
- h) Filtrar productos por categoría específica.
- i) Mostrar productos cuyo stock es menor a 10.

Desafío adicional

- j) Mostrar un listado solo con los nombres de los productos y su stock.

*Pista: investiga como seleccionar columnas de tablas en lugar de todos los campos.

Soluciones Base de Datos para una tienda de tecnología**a) Crea una base de datos con el nombre “tecnología”.**

```
CREATE DATABASE tecnologia;
```

b) Define una tabla llamada productos.

```
USE tecnologia;
```

```
CREATE TABLE productos (  
id_producto INT AUTO_INCREMENT PRIMARY KEY,  
nombre VARCHAR(255) NOT NULL,  
categoria VARCHAR(100) NOT NULL,  
precio DECIMAL(10,2) NOT NULL,  
stock INT DEFAULT 0,  
fecha_ingreso DATE  
);
```

c) Inserta al menos cinco productos en la tabla.

```
INSERT INTO productos (nombre, categoria, precio, stock, fecha_ingreso) VALUES  
( 'Laptop Dell XPS', 'Computadoras', 1200.99, 5, '2024-01-15'),  
( 'iPhone 14', 'Smartphones', 999.99, 10, '2024-02-01'),  
( 'Teclado mecánico', 'Accesorios', 79.50, 15, '2024-01-20'),  
( 'Monitor LG 27"', 'Monitores', 250.00, 7, '2024-01-25'),  
( 'Auriculares Sony WH-1000XM5', 'Audio', 350.99, 8, '2024-02-05'  
);
```

d) Consulta para contar número total de productos.

```
SELECT COUNT(*) FROM productos;
```

e) Consulta para calcular el numero total de stock en la tienda de todos los productos.

```
SELECT SUM(stock) FROM productos;
```

f) Ordenar los productos por precio de mayor a menor.

```
SELECT * FROM productos ORDER BY precio DESC;
```

g) Listar productos por orden de entrada.

```
SELECT * FROM productos ORDER BY fecha_ingreso DESC;
```

h) Filtrar productos por categoría específica.

```
SELECT * FROM productos WHERE categoria = 'Smartphones';
```

i) Mostrar productos cuyo stock es menor a 10.

```
SELECT * FROM productos WHERE stock < 10;
```

Desafío adicional

j) Mostrar un listado solo con los nombres de los productos y su stock.

```
SELECT nombre, stock FROM productos;
```

Tema 5. SQL Avanzado – Integración de Datos y Optimización

Cuestionario

1. ¿Cuál es la principal ventaja de utilizar JOIN en SQL?

- a) Permite combinar datos de dos o más tablas en una consulta.
- b) Reduce el número de registros en una tabla.
- c) Borra automáticamente los datos duplicados.
- d) Optimiza el rendimiento de la base de datos.

2. ¿Qué tipo de JOIN devuelve solo las coincidencias entre dos tablas?

- a) LEFT JOIN
- b) RIGHT JOIN
- c) INNER JOIN
- d) FULL OUTER JOIN

3. ¿Cuál de las siguientes opciones describe correctamente LEFT JOIN?

- a) Devuelve solo los registros coincidentes entre ambas tablas.
- b) Devuelve todos los registros de la tabla izquierda y solo las coincidencias de la tabla derecha.
- c) Devuelve solo los registros de la tabla derecha.
- d) Devuelve todos los registros de ambas tablas.

4. ¿Cuál de las siguientes consultas es correcta para unir la tabla clientes con alquileres, relacionándolas por id_cliente?

- a) `SELECT * FROM clientes FULL JOIN alquileres ON clientes.id_cliente = alquileres.id_cliente;`
- b) `SELECT * FROM clientes INNER JOIN alquileres ON clientes.id_cliente = alquileres.id_cliente;`
- c) `SELECT * FROM clientes MERGE alquileres USING id_cliente;`
- d) `SELECT * FROM clientes RIGHT OUTER JOIN alquileres WHERE clientes.id_cliente = alquileres.id_cliente;`

5. ¿Para qué se utiliza la cláusula GROUP BY en SQL?

- a) Para ordenar los resultados.
- b) Para eliminar registros duplicados.
- c) Para agrupar filas que tienen valores similares en columnas específicas.
- d) Para eliminar registros que no coinciden en ambas tablas.

6. ¿Cuál de las siguientes consultas cuenta cuántos coches hay actualmente alquilados?

- a) `SELECT COUNT(id_coche) FROM alquileres;`
- b) `SELECT COUNT(DISTINCT id_coche) FROM alquileres WHERE CURDATE() BETWEEN fecha_inicio AND fecha_fin;`
- c) `SELECT COUNT(*) FROM alquileres WHERE fecha_inicio > CURDATE();`
- d) `SELECT DISTINCT COUNT(id_coche) FROM alquileres WHERE fecha_inicio <= CURDATE() AND fecha_fin >= CURDATE();`

7. ¿Qué diferencia hay entre UNION e INTERSECT en SQL?

- a) UNION devuelve todos los registros de ambas consultas y INTERSECT solo los registros comunes.
- b) UNION devuelve solo registros duplicados y INTERSECT elimina duplicados.
- c) INTERSECT se usa solo en bases de datos NoSQL.
- d) UNION solo funciona con JOIN.

8. ¿Cuál es la diferencia entre MINUS y LEFT JOIN con condición WHERE ... IS NULL?

- a) MINUS elimina registros duplicados y LEFT JOIN no.
- b) MINUS es más rápido que LEFT JOIN.
- c) MINUS devuelve las filas que están en la primera consulta pero no en la segunda, mientras que LEFT JOIN con WHERE ... IS NULL muestra registros que no tienen coincidencias en la segunda tabla.
- d) No hay ninguna diferencia.

9. ¿Cómo listar todos los productos de una tienda, mostrando incluso los que nunca han sido vendidos?

- a) `SELECT * FROM productos INNER JOIN ventas ON productos.id_producto = ventas.id_producto;`
- b) `SELECT * FROM productos LEFT JOIN ventas ON productos.id_producto = ventas.id_producto;`
- c) `SELECT * FROM productos RIGHT JOIN ventas ON productos.id_producto = ventas.id_producto;`
- d) `SELECT * FROM productos WHERE id_producto NOT IN (SELECT id_producto FROM ventas);`

10. ¿Cuál de estas consultas muestra el nombre del cliente y el modelo del coche alquilado?

- a)

`SELECT clientes.nombre, coches.modelo FROM clientes JOIN alquileres ON clientes.id_cliente = alquileres.id_cliente JOIN coches ON alquileres.id_coche = coches.id_coche;`

- b)

`SELECT clientes.nombre, coches.modelo FROM clientes, coches, alquileres WHERE clientes.id_cliente = alquileres.id_cliente AND alquileres.id_coche = coches.id_coche;`

- c) Ambas son correctas
- d) Ninguna es correcta

SOLUCIONES

1 - A

2 - C

3 - B

4 - B

5 - C

6 - B

7 - A

8 - C

9 - B

10 - C