

Conceptos de SQL para entrevista.

1. DML: Lenguaje de manipulación de datos. Inserta, actualiza y elimina datos.
2. DDL: Lenguaje de definición de datos crea, modifica y elimina objetos.
3. Aggregate Functions: Se utilizan para realizar cálculos en un conjunto de datos.
4. Group by: Cláusula para agrupar filas de una tabla en función de una o varias columnas.
5. Order by: Cláusula para ordenar los resultados de una consulta en función de una o varias columnas.
6. Function: Bloque de código que realiza una tarea específica.
7. Stored Procedures: Programas almacenados, se pueden invocar desde una aplicación o la propia base de datos.
8. Union: Operación, combina los resultados de dos o más consultas en una única tabla de resultados.
9. Triggers: Objetos, se activan automáticamente en respuesta a ciertos eventos.
10. CTE's: Expresiones, se utilizan para crear una tabla temporal que se puede utilizar en una consulta.
11. Transactions: Secuencia de operaciones, se ejecutan como una única unidad lógica de trabajo y se pueden deshacer.

Preguntas de nivel junior.

1. SQL: Lenguaje de programación utilizado para interactuar con bases de datos relacionales.
2. CREATE: Declaración, crea objetos en la base de datos.
3. DROP: Declaración, elimina objetos de la base de datos.
4. Identity fields: Columnas especiales, generan valores únicos para cada nueva fila. Se conocen como campos de identidad, campos de autoincremento o secuencias.
5. SELECT: Declaración, recuperar datos de una o varias tablas en la base de datos.
6. WHERE: Cláusula, filtrar los datos recuperados por SELECT o para especificar una condición en una declaración DELETE o UPDATE.
7. Alias: Nombre alternativo, se puede asignar a una tabla o columna en una declaración SELECT.
8. Null definition: Valor desconocido o ausente en una columna. Se utiliza para indicar que un valor no se ha introducido o no se conoce.
9. Joins: Declaración, combina datos de dos o más tablas en una sola tabla de resultados.
 - Inner join: Filas que tienen coincidencias en ambas tablas.
 - Left join: Filas de la tabla izquierda y las filas coincidentes de la tabla derecha.
 - Right join: Filas de la tabla derecha y las filas coincidentes de la tabla izquierda.
10. INSERT: Declaración, inserta nuevos datos en una tabla existente.
11. UPDATE: Declaración, actualiza los datos existentes en una tabla.
12. DELETE: Declaración, elimina filas de una tabla existente.
13. Normalization: Proceso, organiza una base de datos, reduce la redundancia y evita la inconsistencia en los datos. Diferentes niveles (1NF, 2NF, 3NF).
14. Schema: Estructura lógica de una base de datos, que describe tablas, columnas, relaciones, restricciones, índices, vistas, procedimientos almacenados, entre otros.
15. Relational database: Base de datos que almacena los datos en tablas relacionales.
16. Non-relational database: Base de datos que no utiliza tablas relacionales. Utiliza modelos de datos alternativos, como documentos, grafos, claves-valor, etc. se conocen como NoSQL.
17. Table: Estructura de datos que consiste en filas y columnas. Cada fila representa un registro y cada columna representa un atributo o una propiedad.
18. Stored procedures: Conjunto de instrucciones, se puede llamar desde otras aplicaciones o procedimientos almacenados.
19. Permission administration: Proceso de definir y gestionar los permisos de acceso a los objetos. Los permisos de acceso se pueden asignar a diferentes usuarios.

Preguntas de nivel middle.

1. ALTER: Instrucción, modificar la estructura de los objetos de la base de datos.
2. LIKE y Wildcards: LIKE es un operador, compara una cadena con un patrón de texto. Los Wildcards son caracteres utilizados con el operador LIKE para hacer coincidir patrones de texto complejos.
3. Subqueries: Es una consulta dentro de otra consulta. Recuperar datos de una tabla en función de los resultados de otra consulta.
4. Normalización vs denormalización: Son dos técnicas para la gestión de bases de datos. La normalización se reduce la redundancia y mejora la integridad de los datos. La denormalización mejora el rendimiento de las consultas al recuperar datos de varias tablas en una sola consulta.
5. Null handling: Representa la ausencia de un valor en una columna de la base de datos.
6. Constraints: Son reglas para las columnas de una tabla para garantizar que los datos cumplan ciertas condiciones (claves primarias, las claves foráneas, los valores únicos y las comprobaciones).
7. IN vs EXISTS: Operadores, IN compara una columna con una lista de valores, EXISTS compara una columna con el resultado de una subconsulta.
8. Temporary tables: Se utilizan para almacenar datos de forma temporal. Se utilizan para realizar cálculos intermedios o para almacenar resultados de consultas.

9. **HAVING vs WHERE:** WHERE, filtra los registros antes de que se realice la agrupación, HAVING filtra los resultados después de la agrupación.
10. **Keys (Primary, foreign):** Son restricciones que se aplican a las columnas de una tabla. La clave primaria, identifica de forma única cada registro, la clave foránea establece una relación entre dos tablas.
11. **Stored procedures types:**
- **Simples:** No devuelven ningún valor.
 - **Con valores de retorno:** Devuelven un solo valor.
 - **Con parámetros de entrada:** Aceptan uno o más parámetros de entrada.
 - **Con parámetros de salida:** Devuelven uno o más valores de salida.
 - **Con parámetros de entrada/salida:** Aceptan y devuelven uno o más parámetros de entrada y valores de salida.
12. **Handling data validation:** Es un proceso en la gestión de bases de datos. Algunos son utilizar:
- **Restricciones:** Garantiza que los datos cumplan ciertas condiciones (clave primaria, clave foránea, de valor único o de comprobación).
 - **Cláusula WHERE:** Filtra los registros que cumplen ciertas condiciones, asegura que los registros cumplen con los criterios de validación.
 - **Procedimientos almacenados:** Automatiza la validación de datos, garantiza que sean precisos y estén completos.
 - **Transacciones:** Agrupan varias operaciones en una sola unidad lógica. Aseguran que todas las operaciones se completen con éxito antes de que se guarden los datos.

Preguntas de nivel senior.

1. **Cursor:** Objeto, recorre filas de una tabla en una consulta y realizar operaciones en ellas.
2. **Delete vs Truncate:** Delete elimina filas específicas de una tabla, Truncate las elimina todas. Además, Delete está registrada en el registro de transacciones, Truncate no lo está.
3. **Currency:** en SQL, Representar valores monetarios en aplicaciones financieras.
4. **Transaction's log:** Registra todas las operaciones realizadas en una transacción para garantizar que se puedan deshacer.
5. **ACID:** Acrónimo, describe las propiedades de una transacción. Las propiedades son Atomicidad, Consistencia, Aislamiento y Durabilidad.
6. **Profiling:** Técnica, mide y analiza el rendimiento de una consulta o una base de datos en su conjunto, con el fin de identificar cuellos de botella y mejorar la eficiencia.
7. **Data:** Información almacenada en una base de datos, que puede ser estructurada o no estructurada.
8. **Data unicity:** Propiedad que asegura que no hay duplicados en los datos.
9. **Data dictionaries:** Objetos que contienen información sobre las estructuras de datos y los metadatos (tablas, índices, columnas y restricciones).
10. **Data types handling:** Capacidad de SQL para manejar diferentes tipos de datos.
11. **Key's:** Una clave es una o varias columnas que se utilizan para identificar de manera única cada fila.
 - **Clave primaria:** Identifica de manera única cada fila, Puede ser una clave natural o una clave artificial.
 - **Clave natural:** Columna o conjunto de columnas que existen en la tabla y tienen un valor único en cada fila.
 - **Clave artificial:** Se crea específicamente para ser la clave primaria (secuencia, identificador único).
12. **Grouping sets:** Cláusula, agrupa datos por múltiples columnas y produce varios niveles de resumen en una sola consulta.
13. **Merge:** Operación, combina los datos de dos tablas en una sola tabla.
14. **Indexes:** Estructura de datos, acelera la búsqueda de datos en una tabla.
15. **Partitions:** Técnica, divide una tabla en secciones más pequeñas llamadas particiones, se pueden basar en valores de columna, rangos de valores o en función del tiempo.
16. **Views vs Materialized views:**
 - **Vista:** Consulta predefinida que se almacena y se puede invocar como si fuera una tabla.
 - **Vista materializada:** Copia de una vista que se almacena físicamente en la base de datos y se actualiza periódicamente para reflejar los cambios en los datos subyacentes.