

Quiz Final de Bases de Datos

Este quiz contiene 80 preguntas de Verdadero o Falso para evaluar tus conocimientos sobre los conceptos fundamentales de las bases de datos, basándose en los temas de los apuntes proporcionados.

Sección 1: SQL y Conceptos Generales

1. **F:** SQL es un lenguaje de programación imperativo, similar a Java.
2. **F:** SQL es considerado software libre y de código abierto.
3. **V:** Una base de datos es una colección de datos lógicamente coherente con un significado implícito.
4. **F:** CREATE TABLE es una sentencia del sublenguaje DML (Lenguaje de Manipulación de Datos).
5. **V:** GRANT y REVOKE son sentencias pertenecientes al DCL (Lenguaje de Control de Datos).
6. **V PERO SI TIENE EXT FALSA:** El álgebra relacional y el cálculo relacional de tuplas tienen la misma expresividad que SQL.
7. **F:** Una vista es una tabla física que almacena el resultado de una consulta para mejorar el rendimiento.
8. **V / F:** Los triggers son procedimientos que se ejecutan automáticamente en respuesta a eventos DML.
9. **F:** PostgreSQL y MySQL son implementaciones del estándar SQL que son software propietario.
10. **F:** La principal función de una base de datos es únicamente almacenar datos, sin importar sus relaciones.

Sección 2: Tablas y Claves

11. **V:** Una tabla en una base de datos relacional puede existir sin una clave primaria (PK).
12. **F:** Si una tabla no tiene clave primaria, no puede tener registros duplicados.
13. **F:** Una clave primaria (PK) identifica unívocamente a una columna dentro de una tabla.
14. **V:** Una clave primaria puede estar compuesta por más de una columna.
15. **F:** Los valores de una clave primaria pueden ser modificados (UPDATE) fácilmente para mantener la flexibilidad.
16. **F:** Una clave foránea (FK) siempre debe hacer referencia a una clave primaria en otra tabla.
17. **V / F PARA MI PUEDE:** Una columna que es clave foránea (FK) nunca puede contener valores NULL.
18. **V / F PARA MI PERMITE NULLS:** La restricción UNIQUE asegura que todos los

valores en una columna sean únicos, sin permitir NULLs.

- 19. **V:** Según las heurísticas de diseño, si dos tablas usan la misma PK (mismo dominio y semántica), deberían fusionarse.
- 20. **F:** Es una buena práctica mezclar en la misma tabla datos ingresados por usuarios y datos generados por el sistema (logs).

Sección 3: El Valor NULL

- 21. **F:** En SQL, NULL es equivalente a FALSE en expresiones lógicas.
- 22. **F:** La operación $5 + \text{NULL}$ da como resultado 5, ya que el NULL se interpreta como 0.
- 23. **F:** La comparación $\text{NULL} = \text{NULL}$ devuelve TRUE.
- 24. **V:** La función de agregación $\text{COUNT}(\ast)$ cuenta las filas que contienen valores NULL.
- 25. **V:** La función de agregación $\text{AVG}()$ ignora los valores NULL para calcular el promedio.
- 26. **F:** La expresión NULL IS NOT NULL devuelve TRUE.
- 27. **F:** COALESCE es la única función en SQL para manejar valores NULL.
- 28. **F:** Una clave primaria (PK) puede contener valores NULL.
- 29. **V:** En una clave compuesta, la estrategia FULL MATCH exige que todas las columnas de la FK sean NULL o todas tengan valor.
- 30. **F:** La concatenación de una cadena con NULL (ej: 'hola' || NULL) da como resultado la cadena original ('hola').

Sección 4: ACID y Transacciones

- 31. **V:** La propiedad de Atomicidad (Atomicity) asegura que si una transacción falla, la base de datos vuelve a un estado anterior válido.
- 32. **V:** La Consistencia (Consistency) garantiza que una transacción solo puede llevar la base de datos de un estado válido a otro.
- 33. **F:** El Aislamiento (Isolation) se encarga de que las transacciones no tengan fallos de sistema.
- 34. **V:** La Durabilidad (Durability) asegura que una vez que una transacción ha sido confirmada, sus cambios persistirán incluso si hay fallos del sistema.
- 35. **F:** El nivel de aislamiento SERIALIZABLE es el más rápido porque permite el mayor grado de paralelismo.
- 36. **V:** En el nivel SERIALIZABLE, la ejecución de transacciones concurrentes es equivalente a alguna ejecución en serie de las mismas.
- 37. **F:** Los deadlocks (bloqueos mutuos) no pueden ocurrir en el nivel de aislamiento SERIALIZABLE.
- 38. **F:** Una transacción en SQL solo puede contener una única operación (un solo

INSERT o UPDATE).

39. **F:** La propiedad ACID garantiza que no habrá fallas en el hardware del servidor.

40. **F:** COMMIT y ROLLBACK son sentencias del DML.

Sección 5: Integridad Referencial y Dependencias

41. **V / F:** Es posible insertar una fila en una tabla "hija" (ej: Localidad) con una clave foránea que no existe en la tabla "padre" (ej: Provincia).

42. **V / F:** Se puede borrar una fila en una tabla "padre" (Provincia) sin borrar sus filas asociadas en la tabla "hija" (Localidad) si se define ON DELETE CASCADE.

43. **V / F:** Si la FK en la tabla Localidad es anulable (NULLABLE), se puede insertar una localidad sin una provincia asociada.

44. **V / F:** Es posible cambiar la provincia de una localidad existente a otra provincia que sí existe en la tabla Provincia.

45. **V / F:** La integridad referencial es una regla opcional y no es importante para la consistencia de los datos.

46. **V / F:** Una dependencia funcional $X \rightarrow Y$ significa que el valor de Y determina el valor de X.

47. **V / F:** Si $DNI \rightarrow Nombre$, significa que para un mismo DNI siempre corresponde el mismo Nombre.

48. **V / F:** Una clave candidata es un conjunto de atributos que determina funcionalmente a todos los demás atributos, y no es mínima.

49. **V / F:** Toda tabla en 3FN también está en FNBC (Forma Normal de Boyce-Codd).

50. **V / F:** Una tabla está en 1FN si no contiene atributos multivaluados.

Sección 6: Normalización e Índices

51. **V / F:** Una tabla está en 2FN si todos sus atributos no primos dependen de forma completa de la clave primaria.

52. **V / F:** Una tabla está en 3FN si, para toda dependencia funcional no trivial $X \rightarrow Y$, X es una superclave o Y es un atributo primo.

53. **V / F:** Estar en 3FN pero no en FNBC siempre se considera un error grave de diseño.

54. **V / F:** La normalización ayuda a reducir la redundancia de datos.

55. **V / F:** Un índice clustered define el orden físico en que los datos se almacenan en el disco.

56. **V / F:** Una tabla puede tener múltiples índices clustered.

57. **V / F:** Un índice non-clustered tiene un ordenamiento independiente al de los datos en el archivo.

58. **V / F:** Un índice denso tiene una entrada por cada valor de la clave en la tabla.

59. **V / F:** Un índice clustered primario siempre es denso.

60. **V / F:** Los índices aceleran las operaciones de UPDATE y DELETE en todos los casos.

Sección 7: Optimización y NoSQL

61. **V / F:** El optimizador de consultas de un DBMS siempre elige el plan de ejecución más rápido posible.
62. **V / F:** Una heurística común del optimizador es resolver las selecciones (WHERE) lo más tarde posible.
63. **V / F:** Un histograma de "igual ancho" divide los datos en rangos que tienen la misma cantidad de filas.
64. **V / F:** El plan de ejecución de una consulta SQL se genera después de que el motor ejecuta la consulta.
65. **V / F:** El Block Nested Loop Join es siempre el método de JOIN más eficiente.
66. **V / F:** Las bases de datos NoSQL son ideales para modelos de datos que requieren joins complejos y frecuentes.
67. **V / F:** Las bases de datos NoSQL suelen ofrecer garantías ACID más débiles, como el modelo BASE.
68. **V / F:** MongoDB es un ejemplo de una base de datos NoSQL basada en grafos.
69. **V / F:** Redis es un ejemplo de una base de datos NoSQL de tipo clave-valor.
70. **V / F:** El esquema en las bases de datos NoSQL es generalmente flexible o dinámico, a diferencia del esquema rígido de SQL.

Sección 8: Temas Avanzados

71. **V / F:** El logging en una base de datos tiene como objetivo principal registrar los inicios de sesión de los usuarios.
72. **V / F:** El cache manager es responsable de traer bloques de datos desde la memoria RAM al disco.
73. **V / F:** Un "dirty bit" en una página de caché indica que la página ha sido modificada y no coincide con la versión en disco.
74. **V / F:** XML, a diferencia de JSON, solo puede almacenar datos y no metadatos.
75. **V / F:** JSON es generalmente más liviano y tiene una estructura más simple que XML.
76. **V / F:** Big Data se refiere exclusivamente a un gran volumen de datos, sin importar la velocidad o variedad.
77. **V / F:** La ley argentina de protección de datos personales prohíbe almacenar datos sensibles, incluso con el consentimiento del titular.
78. **V / F:** UTF-8 es una codificación de caracteres de longitud fija, donde cada carácter ocupa siempre 8 bits.
79. **V / F:** Collation define las reglas de comparación de cadenas, por ejemplo, si 'a'

es igual a 'A'.

80. **V / F:** Una baja lógica (usar un flag como activo=0) es la práctica recomendada sobre la eliminación física (DELETE).

Respuestas Correctas

1. **F:** Es declarativo.
2. **F:** SQL es un estándar, algunas implementaciones son de código abierto (MySQL) y otras no (SQL Server).
3. **V**
4. **F:** Es DDL (Lenguaje de Definición de Datos).
5. **V**
6. **F:** SQL es más expresivo (agregación, recursión, etc.).
7. **F:** Es una tabla virtual, no almacena datos físicamente.
8. **V**
9. **F:** Son software libre.
10. **F:** También gestiona la integridad y las relaciones entre ellos.
11. **V:** Aunque no es una buena práctica.
12. **F:** Sin PK, sí puede haber registros duplicados.
13. **F:** Identifica a un registro (fila).
14. **V**
15. **F:** Una PK debe ser inmutable.
16. **F:** Puede hacer referencia a una clave UNIQUE.
17. **F:** Puede ser NULL si la columna está definida como anulable.
18. **F:** La restricción UNIQUE permite múltiples valores NULL.
19. **V**
20. **F:** Se recomienda separarlos.
21. **F:** Es un valor desconocido, no es ni TRUE ni FALSE.
22. **F:** El resultado es NULL.
23. **F:** Devuelve NULL (o desconocido).
24. **V**
25. **V**
26. **F:** Devuelve FALSE.
27. **F:** Existen otras como ISNULL, NVL, etc.
28. **F:** Nunca.
29. **V**
30. **F:** El resultado es NULL.
31. **V**
32. **V**
33. **F:** Garantiza que las transacciones concurrentes no interfieran entre sí.

- 34. **V**
- 35. **F:** Es el más restrictivo y puede ser el más lento.
- 36. **V**
- 37. **F:** Los deadlocks pueden ocurrir.
- 38. **F:** Puede contener múltiples operaciones.
- 39. **F:** Se refiere a fallas lógicas y de sistema, no de hardware.
- 40. **F:** Son TCL (Lenguaje de Control de Transacciones).
- 41. **F:** Violaría la integridad referencial.
- 42. **F:** ON DELETE CASCADE borraría las filas asociadas.
- 43. **V**
- 44. **V**
- 45. **F:** Es fundamental para la consistencia.
- 46. **F:** X determina a Y.
- 47. **V**
- 48. **F:** Debe ser mínima.
- 49. **F:** Al revés: toda tabla en FNBC está en 3FN.
- 50. **V**
- 51. **V**
- 52. **V**
- 53. **F:** Puede ser una decisión de diseño para mejorar la expresividad.
- 54. **V**
- 55. **V**
- 56. **F:** Solo puede haber uno.
- 57. **V**
- 58. **V**
- 59. **F:** Es no denso.
- 60. **F:** Las ralentizan, ya que el índice también debe actualizarse.
- 61. **F:** Elige el que estima que tiene el menor coste, que no siempre es el más rápido en la realidad.
- 62. **F:** Al revés, las resuelve lo antes posible para reducir el conjunto de datos.
- 63. **F:** Eso es un histograma de "igual altura". El de igual ancho tiene rangos de la misma longitud (ej: 1-10, 11-20).
- 64. **F:** Se genera antes de la ejecución.
- 65. **F:** Suele ser ineficiente para tablas grandes.
- 66. **F:** SQL es mejor para eso. NoSQL evita los joins complejos.
- 67. **V**
- 68. **F:** Es basada en documentos. Neo4j es basada en grafos.
- 69. **V**
- 70. **V**

- 71. **F:** Su objetivo es la recuperación ante fallos.
- 72. **F:** Trae bloques desde el disco a la caché en memoria.
- 73. **V**
- 74. **F:** XML puede almacenar metadatos en sus atributos y estructura.
- 75. **V**
- 76. **F:** También se define por su velocidad y variedad (las 3 V).
- 77. **F:** Prohíbe almacenarlos sin consentimiento y sin medidas de seguridad adecuadas.
- 78. **F:** Es de longitud variable.
- 79. **V**
- 80. **F:** Se prefiere la eliminación física o el archivado.