

TEST

50% de la nota. Un fallo resta una correcta. Hay también preguntas cortas que restan. Puede haber más de una respuesta correcta (Si la respuesta correcta es seleccionar 2 y seleccionas 1 está mal y resta 1).

1. Cuando se modifica la base de datos, se anota en el registro:
 - a) Antes
 - b) Después
 - c) Simultáneamente para cumplir atomicidad y durabilidad
 - d) Simultáneamente para cumplir consistencia y durabilidad
2. Una transacción está comprometida cuando
 - a) realiza la última actualización en la BD
 - b) realiza la última operación de escritura
 - c) todo su registro histórico está en memoria intermedia
 - d) todo su registro histórico está en memoria estable
3. ¿Es mejor un interbloqueo o un estado inconsistente?
4. El protocolo de bloqueo en 2 fases evita interbloqueos? Si los evita como lo hace?
5. Una transacción modifica la BD si hace actualización en
 - a) exclusivamente memoria intermedia
 - b) exclusivamente no volátil
 - c) exclusivamente estable
 - d) en memoria intermedia, estable o no volátil
6. ¿Qué contiene un plan de ejecución de una consulta?
 - a) Operaciones sobre la BD
 - b) Algoritmos en cada operación
 - c) Sintaxis SQL óptima
 - d) Coordinación de operaciones (materialización o encauzamiento)
7. Qué afirmación es incorrecta sobre organización de registros en bloques:
 - a) Los archivos se organizan como secuencias de registros
 - b) Los archivos de la BD se dividen en bloques
 - c) Ningún registro es mayor que un bloque
 - d) Un bloque contiene exactamente un registro

8. ¿Cuándo se puede quedar espacio de bloque sin usar por registros?
- Con longitud fija
 - Con longitud variable
 - Ambos
 - Nunca
9. ¿Para qué búsquedas son mejores los índices ordenados y para qué búsquedas son mejores las asociaciones?
10. En qué esquema de acceso se utilizan los árboles B +? ¿Qué tamaño puede tener cada nodo?
11. Hay índices que no tengan todos los valores de la clave de búsqueda?
- Si, denominados densos
 - Si, denominados dispersos
 - Si, denominados secundarios
 - No
12. El optimizador encuentra siempre el plan menos costoso?
- Siempre
 - A veces
 - Nunca
13. Si se pudiera elegir libremente. ¿Qué enfoque de conexión de primitivas es preferible?
- Secuenciable
 - Concurrente
 - Encauzamiento
 - Materialización
14. Que expresiones del álgebra relacional son equivalentes:
- Mismo conjunto de tuplas en cualquier orden
 - Tardan lo mismo
 - Mismo conjunto de tuplas en mismo orden
 - Mismas primitivas de evaluación
15. ¿Cuál es el coste más importante de una consulta?
- Transferencia de memoria no volátil.
 - Transferencia de memoria intermedia
 - Acceso de memoria no volátil
 - Acceso de memoria intermedia
16. ¿Cómo mejorar el coste de una consulta?
- Adelantando la proyección y la selección.
 - Retrasando la proyección y la selección
 - Adelantando la reunión
 - Retrasando la reunión

17. Nivel de RAID que requiere de más discos para almacenar la misma información en memoria estable.

JDBC

50% de la nota.

1. ¿Qué es JDBC? Pasos de una conexión. Muestra el ciclo completo.
2. Cómo implementarías la inserción de un gran número de filas en una tabla usando JDBC. Describe los pasos. Supón que los valores están almacenados en un fichero csv.
3. Tipos de Statement y diferencias entre ellos. Método para ejecutarlos: ¿Cuándo y cómo?
4. ¿Qué es el ResultSet? Mostrar todos los resultados en una JTable. Hazlo en pseudocódigo/código.
5. Desde el punto de vista de la seguridad: Aspectos que hay que valorar al desarrollar un programa que accede a una BD.

Corrección (Pueden estar mal)

1. Cuando se modifica la base de datos, se anota en el registro:

- e) Antes
- f) Después
- g) Simultáneamente para cumplir atomicidad y durabilidad
- h) Simultáneamente para cumplir consistencia y durabilidad

2. Una transacción está comprometida cuando

- e) realiza la última actualización en la BD
- f) realiza la última operación de escritura
- g) todo su registro histórico está en memoria intermedia
- h) todo su registro histórico está en memoria estable

3. ¿Es mejor un interbloqueo o un estado inconsistente?

Para mantener una de las cuatro propiedades ACID, la de la consistencia en la base de datos, es preferible que se produzca un interbloqueo (deadlock) para posteriormente solucionarlo

4. El protocolo de bloqueo en 2 fases evita interbloqueos? Si los evita como lo hace?

El protocolo en 2 fases no está diseñado para prevenir el deadlock, por tanto no lo haría

5. Una transacción modifica la BD si hace actualización en

- e) exclusivamente memoria intermedia

- f) exclusivamente no volátil
- g) exclusivamente estable
- h) en memoria intermedia, estable o no volátil

6. ¿Qué contiene un plan de ejecución de una consulta?

- e) Operaciones sobre la BD
- f) Algoritmos en cada operación
- g) Sintaxis SQL óptima
- h) Coordinación de operaciones (materialización o encauzamiento)

7. Qué afirmación es incorrecta sobre organización de registros en bloques:

- e) Los archivos se organizan como secuencias de registros
- f) Los archivos de la BD se dividen en bloques
- g) Ningún registro es mayor que un bloque
- h) Un bloque contiene exactamente un registro

8. ¿Cuándo se puede quedar espacio de bloque sin usar por registros?

- e) Con longitud fija
- f) Con longitud variable
- g) Ambos
- h) Nunca

9. ¿Para qué búsquedas son mejores los índices ordenados y para qué búsquedas son mejores las asociaciones?

Los índices ordenados son mejores para las búsquedas por rango, mientras que las asociaciones son útiles para las búsquedas por comparación

10. En qué esquema de acceso se utilizan los árboles B +? ¿Qué tamaño puede tener cada nodo?

Se usan en acceso ordenado, y tienen $n-1$ claves de búsqueda y n parámetros

11. Hay índices que no tengan todos los valores de la clave de búsqueda?

- e) Si, denominados densos
- f) Si, denominados dispersos
- g) Si, denominados secundarios
- h) No

12. El optimizador encuentra siempre el plan menos costoso?

- d) Siempre
- e) A veces
- f) Nunca

13. Si se pudiera elegir libremente. ¿Qué enfoque de conexión de primitivas es preferible?

- e) Secuenciable
- f) Concurrente
- g) Encauzamiento

h) Materialización

14. Que expresiones del álgebra relacional son equivalentes:

e) Mismo conjunto de tuplas en cualquier orden

f) Tardan lo mismo

g) Mismo conjunto de tuplas en mismo orden

h) Mismas primitivas de evaluación

15. ¿Cuál es el coste más importante de una consulta?

e) Transferencia de memoria no volátil.

f) Transferencia de memoria intermedia

g) Acceso de memoria no volátil

h) Acceso de memoria intermedia

16. ¿Cómo mejorar el coste de una consulta?

e) Adelantando la proyección y la selección.

f) Retrasando la proyección y la selección

g) Adelantando la reunión

h) Retrasando la reunión

17. Nivel de RAID que requiere de más discos para almacenar la misma información en memoria estable.

Para la misma cantidad de información el nivel que requiere más discos es el Raid 1, pues este utiliza un disco más por disco ocupado, es decir, por cada disco hay una copia de este

JDBC

50% de la nota.

6. ¿Qué es JDBC? Pasos de una conexión. Muestra el ciclo completo.

JDBC es una api que permite la conexión a una base de datos en java, pudiendo interactuar con ella en el código.

Para la conexión a JDBC se requieren los siguientes pasos:

1: Importamos el paquete java.sql.* (para importar todo)

2: Creamos y registramos el controlador

3: Creamos la conexión Connection conn = DriverManager.getConnection(...)

4: Ejecutaremos las transacciones y statements necesarios, como por ejemplo Statement st = conn.createStatement (...)

5: Cerramos la conexión con conn.close();

7. Cómo implementarías la inserción de un gran número de filas en una tabla usando JDBC. Describe los pasos. Supón que los valores están almacenados en un fichero csv.

Para la inserción de un gran número de filas a una tabla es útil utilizar los prepared statement, puesto que no especifican en la sentencia ningún valor, sino que ponen '?'

Suponiendo que se encuentra en un fichero csv, el proceso a seguir sería el siguiente:

- 1: conectamos la base de datos de la manera previamente descrita
- 2: leemos el archivo csv
- 3: creamos el prepared statement "(por ejemplo insert into tabla values (?, ?, ?, ?)"
- 4: iteramos sobre el archivo csv y separando las columnas de los valores para pasarlos al prepared statement.
- 5: ejecutamos el prepared statement, hacemos `executeUpdate()` y cerramos la conexión

8. Tipos de Statement y diferencias entre ellos. Método para ejecutarlos: ¿Cuándo y cómo?

Existen tres principales tipos de statement:

Por un lado tenemos la sentencia básica, el Statement, que es una sentencia de sql básica. Un ejemplo podría ser el siguiente : "Select * from empleados". Esta no es muy útil, menos cuando queremos leer de la tabla, como en el ejemplo anterior, que nos devolvería toda la información de los empleados. Se ejecuta con `executeQuery` si es una consulta o `executeUpdate` si es una modificación.

Por otra parte tenemos PreparedStatement y nos permite compilarla una vez pero ejecutarla varias veces cambiando los valores "?" con `setString` o `setInt` en cada ejecución lo que nos permite, a diferencia del anterior que solo permite una compilación por ejecución, leer muchas tuplas de datos en la misma compilación, útil para cuando hay que leer muchos datos. Se declara con PreparedStatement ejecutandolo con `executeUpdate` o `executeQuery` como el anterior.

Y por último el CallableStatement que se utiliza para crear sentencias de sql que son instanciables como funciones o procedimientos. Se crea con CallableStatement y siempre se ejecuta con `execute()`.

9. ¿Qué es el ResultSet? Mostrar todos los resultados en una JTable. Hazlo en pseudocódigo/código.

ResultSet se refiere al objeto de JDBC que contiene los resultados de una consulta de sql.

Así sería el código de resultados en una JTable:

```
//Creamos el resultset suponiendo que hay una sentencia
ResultSet rs = stm.executeQuery()

//Cogemos información de los meta datos
ResultSetMetaData meta = rs.getMetaData()
int numeroColumnas = meta.getColumnCount()

//Creamos la tabla
DefaultTableModel tableModel = new DefaultTableModel();
```

```
//Iteramos a través del result set, como podría ser el siguiente
while(rs.next){
    //Creamos un obj para almacenar los datos de la fila
    Object[] row = new Object[numeroColumnas]
    for(int i = 0; i < numeroColumnas; i++){
        row[i] = rs.getObject(i)
    }
    tableModel.addRow(row)
}
```

```
JTable table = new JTable(tableModel)
```

10 . Desde el punto de vista de la seguridad: Aspectos que hay que valorar al desarrollar un programa que accede a una BD.

Desde el punto de vista de la seguridad, es importante evitar la inyección de sql. Este proceso consiste en que si tenemos por ejemplo una consulta de la manera `Select * from usuarios where username = 'usuario' and password = 'contraseña'`, si alguien introdujese en los dos campos, tanto de contraseña como de usuario, una sentencia como `' or '1' = '1'`, la sentencia si tenemos un statement quedaría de la siguiente manera `Select * from usuarios where username = " or '1' = '1' and password = " or '1' = '1'` devolviendo siempre todos los usuarios, y teniendo un grave problema de seguridad. Para evitar esto usaremos los prepared statement, que evitan el problema anterior.

Por otra parte tenemos los privilegios. Las cuentas que pueden acceder a la base de datos tienen que tener los privilegios mínimos para la correcta realización de sus tareas, para así evitar males innecesarios. Una manera de captar esto es con la implementación de los roles en la base de datos con `grant role`, limitando así las acciones de cada rol al espacio necesario.

Y por último cabe destacar las trazas de autoría, es decir, “apuntar” los cambios realizados en la base de datos y el usuario que lo hizo para que si existiese algún problema de seguridad se pudiesen revertir los daños causados y quitarle el rol con `revoke role` a la persona que llevó a cabo ese daño, quitando de esa manera los privilegios de los que gozaba anteriormente.

Estos solo son algunos de los aspectos a desarrollar, puesto que hay muchos factores a tener en cuenta.