



# **“PREGUNTAS SOBRE TRANSACCIONES, BLOQUEOS Y NIVELES DE AISLAMIENTO”**

---

## **TALLER DE BASE DE DATOS**

### **INGENIERIA EN SISTEMAS COMPUTACIONALES**

PRESENTA:

**JOSE ANTONIO GONZALEZ CARDENAS**

**LEONARDO MARTINEZ GONZALEZ**

---





## CONTENIDO

¿QUÉ SON LAS TRANSACCIONES? .....	3
¿CÓMO APLICO LAS TRANSACCIONES A MI PROYECTO? .....	3
¿TIPOS DE BLOQUEOS?.....	3
¿NIVELES DE AISLAMIENTO Y CÓMO FUNCIONAN?.....	4



## ¿QUÉ SON LAS TRANSACCIONES?

Una transacción es un conjunto de operaciones (consultas, actualizaciones, inserciones, eliminaciones) que se ejecutan como una única unidad lógica de trabajo en una base de datos. Las transacciones aseguran que todas las operaciones dentro de una misma transacción se realicen correctamente. Si alguna de las operaciones falla, la transacción se puede revertir (rollback) y la base de datos se queda en su estado original, garantizando la integridad de los datos.

## ¿CÓMO APLICO LAS TRANSACCIONES A MI PROYECTO?

Para aplicar transacciones en tu proyecto, debes usar las instrucciones adecuadas en el sistema de gestión de bases de datos que estés utilizando, como SQL Server, MySQL, PostgreSQL, etc.

Por ejemplo, en SQL Server, las transacciones se manejan de la siguiente manera:

**BEGIN TRANSACTION; -- Inicia la transacción**

**-- Operaciones de la transacción (INSERT, UPDATE, DELETE, etc.)**

**INSERT INTO Tabla (columna1, columna2) VALUES (valor1, valor2);**

**-- Si todas las operaciones son exitosas, confirmar la transacción**

**COMMIT TRANSACTION;**

**-- Si hay un error, revertir todos los cambios realizados en la transacción**

**ROLLBACK TRANSACTION;**

Las transacciones se aplican de acuerdo con el flujo de operaciones dentro de la base de datos, asegurando que todos los cambios sean coherentes.

## ¿TIPOS DE BLOQUEOS?

En SQL Server, los bloqueos controlan cómo las transacciones acceden a los datos y evitan conflictos entre ellas. Los tipos de bloqueos más comunes son:

- **Shared Lock (S):** Se utiliza para operaciones de lectura (SELECT). Permite que otros procesos también realicen lecturas, pero impide escrituras mientras el bloqueo esté activo.

- **Exclusive Lock (X):** Se utiliza para operaciones de escritura (INSERT, UPDATE, DELETE). Bloquea el acceso a la fila o tabla para otros procesos hasta que se libere el bloqueo.
- **Update Lock (U):** Es un tipo de bloqueo intermedio. Se utiliza cuando un proceso desea actualizar una fila, y es un bloqueo preventivo para evitar que otro proceso realice una escritura simultánea en el mismo recurso.
- **Intent Locks (IS, IX):** Son bloqueos que se aplican a niveles más altos de la jerarquía (como las tablas) para indicar que hay un bloqueo más específico (como en una fila) a un nivel inferior.

Los bloqueos pueden ocurrir de manera optimista o pesimista:

- **Bloqueo optimista:** Permite que las transacciones lean y escriban sin bloquear los recursos, pero verifica si hubo conflictos al final.
- **Bloqueo pesimista:** Restringe el acceso a los recursos, bloqueándolos para que no haya interferencia entre transacciones.

## ¿NIVELES DE AISLAMIENTO Y CÓMO FUNCIONAN?

Los niveles de aislamiento definen el grado de visibilidad que tienen las transacciones sobre los datos modificados por otras transacciones. Estos niveles controlan el comportamiento de las transacciones en cuanto a:

- **Lecturas sucias (dirty reads):** Leer datos que aún no han sido confirmados.
- **Lecturas no repetibles (non-repeatable reads):** Leer datos que cambian durante la transacción.
- **Inserciones fantasmas (phantom reads):** Leer un conjunto de datos que cambia durante la transacción (por ejemplo, filas nuevas insertadas por otra transacción).

Los niveles de aislamiento en SQL Server son:

1. **READ UNCOMMITTED:** Permite lecturas sucias (lectura de datos no confirmados). No bloquea las lecturas, pero puede dar lugar a datos inconsistentes.



SET TRANSACTION ISOLATION LEVEL READ UNCOMMITTED;

- 2. READ COMMITTED (predeterminado):** No permite lecturas sucias. Solo lee datos confirmados. Pero puede permitir lecturas no repetibles (si otro proceso modifica los datos entre lecturas).

SET TRANSACTION ISOLATION LEVEL READ COMMITTED;

- 3. REPEATABLE READ:** Evita lecturas no repetibles bloqueando los registros leídos para que no cambien durante la transacción. Puede generar bloqueos más agresivos, lo que podría afectar el rendimiento.

SET TRANSACTION ISOLATION LEVEL REPEATABLE READ;

- 4. SERIALIZABLE:** Es el nivel más alto de aislamiento. Bloquea un rango de registros, evitando lecturas no repetibles y filas fantasma. Garantiza que una transacción vea los mismos datos durante toda su ejecución.

SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;

Resumen de cómo funcionan los niveles de aislamiento:

Nivel de Aislamiento	Lecturas Sucias	Lecturas No Repetibles	Inserciones Fantasma
<b>READ UNCOMMITTED</b>	Permitidas	Permitidas	Permitidas
<b>READ COMMITTED</b>	No permitidas	Permitidas	Permitidas
<b>REPEATABLE READ</b>	No permitidas	No permitidas	Permitidas
<b>SERIALIZABLE</b>	No permitidas	No permitidas	No permitidas

