

TRANSACCIONES

MG. MOISÉS ÁLVAREZ HUAMÁN

TRANSACCIONES

- En la actualidad las grandes organizaciones que poseen sistemas automatizados no podrían funcionar de una manera adecuada sin un procesamiento de transacciones confiables y eficientes.
- Una transacción de base de datos “comprende un grupo de operaciones que se deben procesar como una unidad de trabajo. Las transacciones se deben procesar de manera confiable, de modo que no haya ninguna pérdida de datos debido a usuarios múltiples o fallas en el sistema”
- Una transacción “se inicia por la ejecución de un programa de usuario escrito en un lenguaje de manipulación de datos de alto nivel o en un lenguaje de programación, y está delimitado por instrucciones de la forma inicio transacción y fin transacción. La transacción consiste en todas las operaciones que se ejecutan entre inicio transacción y el fin transacción” (Silberschatz, 2002).

PROPIEDADES DE LA TRANSACCIÓN

- **Atomicidad:** “significa que una transacción no se puede subdividir. Ya sea que se realice todo el trabajo en la transacción o que no se haga nada”. (Mannino) Esta propiedad establece que se realizan todas las operaciones de la transacción de la base de datos o no se efectúa ninguna.
- **Consistencia:** “significa que, si las limitaciones aplicables son ciertas antes de empezar la transacción, éstas también lo serán al terminarla”. (Mannino) Esta propiedad consiste en que los datos deben estar bien relacionados y no existan problemas de falta de información.
- **Aislamiento:** “significa que las transacciones no interfieren entre sí, excepto en formas permitidas. Una transacción nunca debe sobrescribir los cambios realizados por otra. Además, una transacción no debe interferir en otros aspectos, como no ver los cambios temporales realizados por otras transacciones” (Mannino).
- **Durabilidad:** “significa que cualquier cambio que resulte de una transacción es permanente. Ninguna falla va a borrar ningún cambio después de terminar la transacción”. (Mannino) Esta propiedad indica que, al finalizar una transacción con éxito, las modificaciones realizadas en la base de datos permanecen, incluso si se presentan fallos en el sistema.
- **Reglas ACID:** comprobar la propiedad de atomicidad (lo que se ejecuta en una transacción se ejecuta todo o nada), usando la tabla “Contrato”.

EJEMPLO

- Para esto se hace uso del comando COMMIT, el cual se utiliza para confirmar como permanentes las modificaciones realizadas en una transacción:

Indica el inicio de una secuencia de comandos

Inserción de un registro en la tabla clientes

```
BEGIN;  
INSERT INTO PUBLIC.ALUMNO VALUES  
( '20694099', 'CEI', 'Lima Huamán', 'Jacob', 'M', 'AV. PROGRESO 221', 'II', 'D' );  
COMMIT;
```

Confirma las modificaciones realizadas en una transacción

Para comprobar que los comandos anteriores se ejecutaron y almacenaron en la base de datos se hace una consulta en la tabla clientes:

```
380 SELECT * FROM ALUMNO;
```

Data Output Explain Messages Notifications

	id_alumno character (8)	id_carrera character (3)	apellidos character varying (40)	nombres character varying (40)	sexo character (1)	direccion character varying (40)	semestre character (3)	torno character (1)
10	20694084	CON	GARCÍA SANTOS	LIDIA ETHEL	F	AV. MARISCAL CASTILLA ...	VI	N
11	20694085	FAR	CASAS BALVIN	ROBERT	M	JR.CENTENARIO 123	VI	D
12	20694086	ENF	CASAS DAVILA	SARITA	F	JR.HUANUCO 123	I	D
13	20694087	AGR	VILCAPOMA MANRIQUE	WILY	M	JR.LEANDRA TORRES 123	V	D
14	20694088	MEC	PECHO DAVILA	SAUL	M	AV. LEONCIO PRADO 123	VI	D
15	20694076	ENF	BERNAL VIZCARRA	ELY	F	AV. ARTERIAL 123	IV	D
16	20694099	CEI	Lima Huamán	Jacob	M	AV. PROGRESO 221	II	D

Otra forma de comprobar la propiedad de atomicidad es la que se indica a continuación, en la cual se hace uso del comando ROLLBACK, el cual permite deshacer todas las modificaciones que se han realizado a la base de datos pero que no han sido escritas en el disco duro por la sentencia COMMIT.

Indica el inicio de una secuencia de comandos

Inserción de un registro en la tabla clientes

```
BEGIN;  
INSERT INTO PUBLIC.ALUMNO VALUES  
( '20694100', 'ENF', 'ESPINOZA FLORES', 'GABRIELA', 'M', 'AV. VISTA EL MANTARO 221', 'II', 'D' );  
ROLLBACK;
```

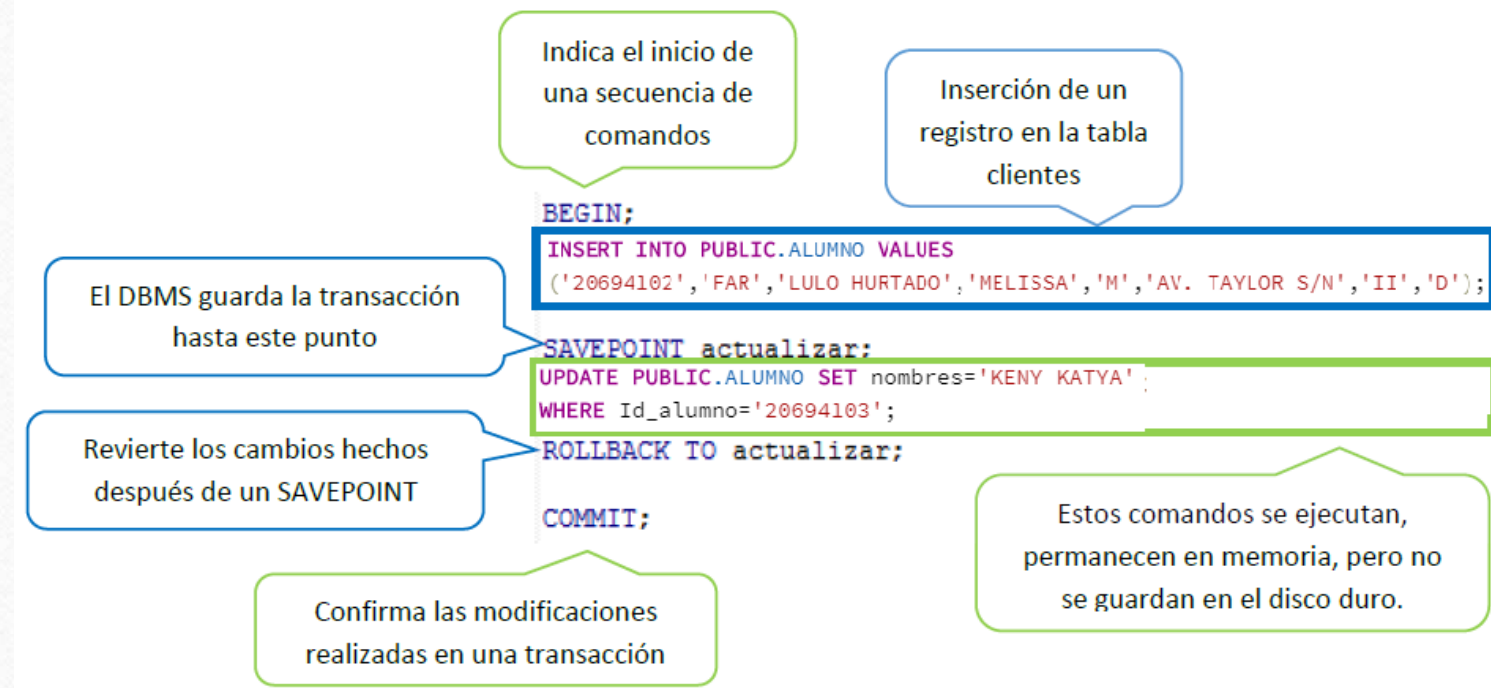
Confirma las modificaciones realizadas en una transacción

Para comprobar que los comandos anteriores se ejecutaron y no se almacenaron en la base de datos se hace una consulta en la tabla clientes:

```
SELECT * FROM ALUMNO;
```

Otros de los comandos que se pueden usar para la comprobación de la propiedad de atomicidad son: **SAVEPOINT** y **ROLLBACK TO**.

SAVEPOINT le indica al DBMS la ubicación de un punto de retorno en una transacción en caso de que la transacción sea cancelada. **ROLLBACK TO** revierte los cambios hechos después de




```
SELECT * FROM ALUMNO;
```

- Con lo cual se evidencia que las instrucciones que se encuentran dentro del bloque `SAVEPOINT <actualizar>` y `ROLLBACK TO <actualizar>` no han efectuado ningún cambio, porque no han sido escritas en el disco duro, y en cambio la información que se ubica fuera del bloque `SAVEPOINT <actualizar>` y `ROLLBACK TO <actualizar>` si ha sido almacenada en la base de datos.

● Reglas ACID: comprobar la propiedad de aislamiento (los cambios en una transacción no terminada no se ven en otra sesión), usando la tabla “Contrato”.
Para comprobar que PostgreSQL cumple con la propiedad de aislamiento se abren dos ventanas para ejecutar consultas SQL.
