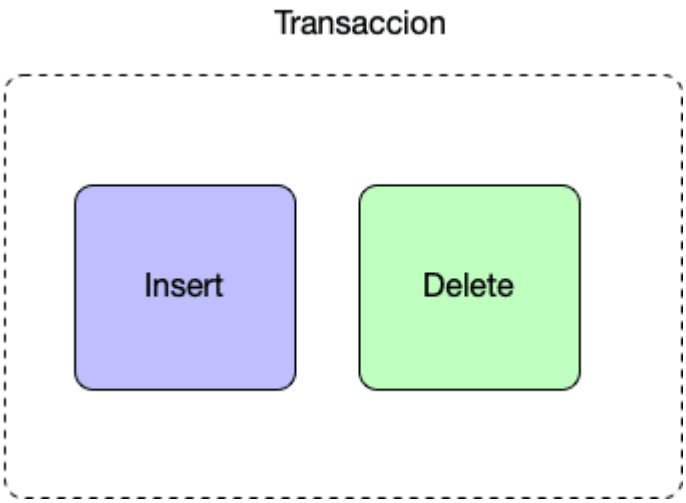
Las Transacciones ACID son uno de los términos más comunes cuando trabajamos con transacciones . Sin embargo muchas personas tienen dudas de que implica el acrónimo ACID . Vamos a hablar un poco de ello . El concepto de Transacción en el lenguaje común esta ligado a una operación que se realiza en varios pasos y que es indivisible.



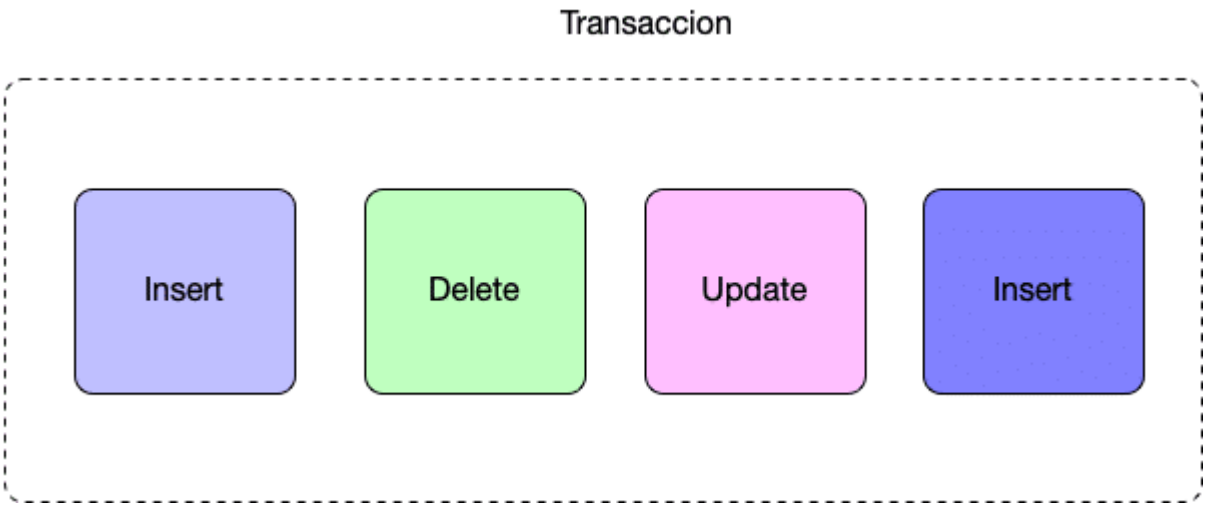
Por ejemplo la compra de un producto. Cuando nosotros compramos un producto realizamos un pago de un importe (100 euros) y a cambio recibimos el producto . Se trata de una operación que incluye dos pasos y estos no se pueden dividir o se ejecutan ambos o no se ejecuta ninguno. Ese es el concepto de transacción a grosso modo.

Transacciones ACID y Bases de Datos

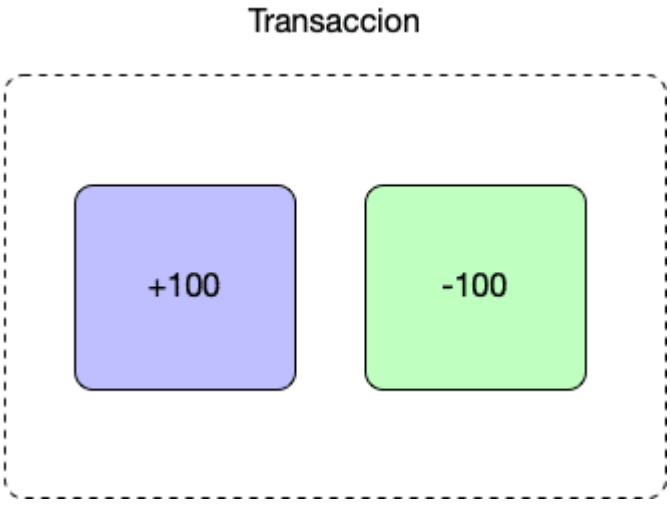
A nivel de base de datos que es lo que nos interesa a los desarrolladores una Transacción debe cumplir con el acrónimo ACID (Atomic,Consistent, Isolated, Durable) . Vamos a explicar estos términos uno a uno para entenderlos mejor.

Atomic : Una transacción debe ser atómica y por lo tanto indivisible. Esto hace referencia a la compra que hemos comentado anteriormente . Pero también implica que no

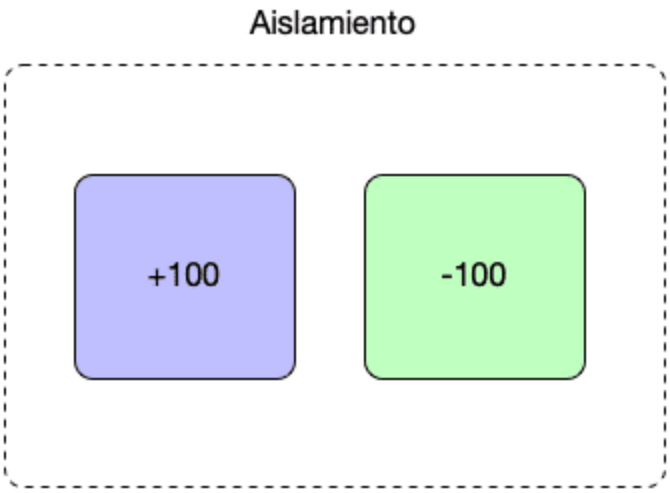
necesariamente tienen que ser solo 2 pasos sino que pueden ser 10 tranquilamente . Lo que aquí implica es que no se pueden dividir es decir si yo compro 10 cosas o las compro todas y pago o no compro ninguna y no pago nada. Ese es el concepto de Atómico o indivisible.

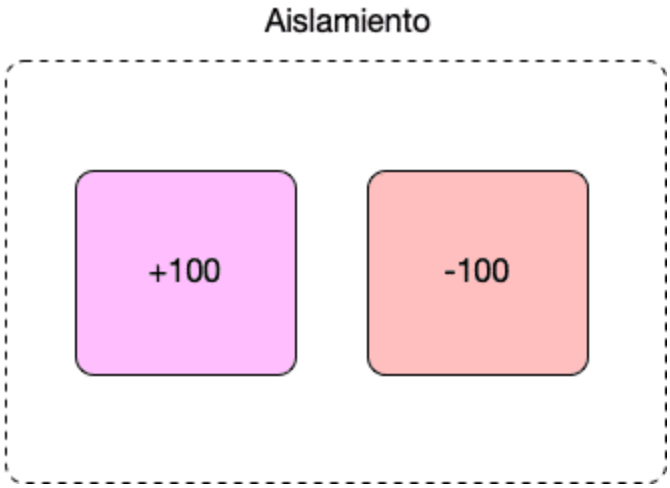


Consistent: Una transacción debe ser consistente en cuanto a la gestión de la información es decir si yo compro 3 productos que cuestan en total 100 eruros . No puede ser que me lleguen 2 productos o no puede ser que me lleguen 3 pero que pague 200 euros. La información que se ejecuta dentro de una transacción debe guardar consistencia.

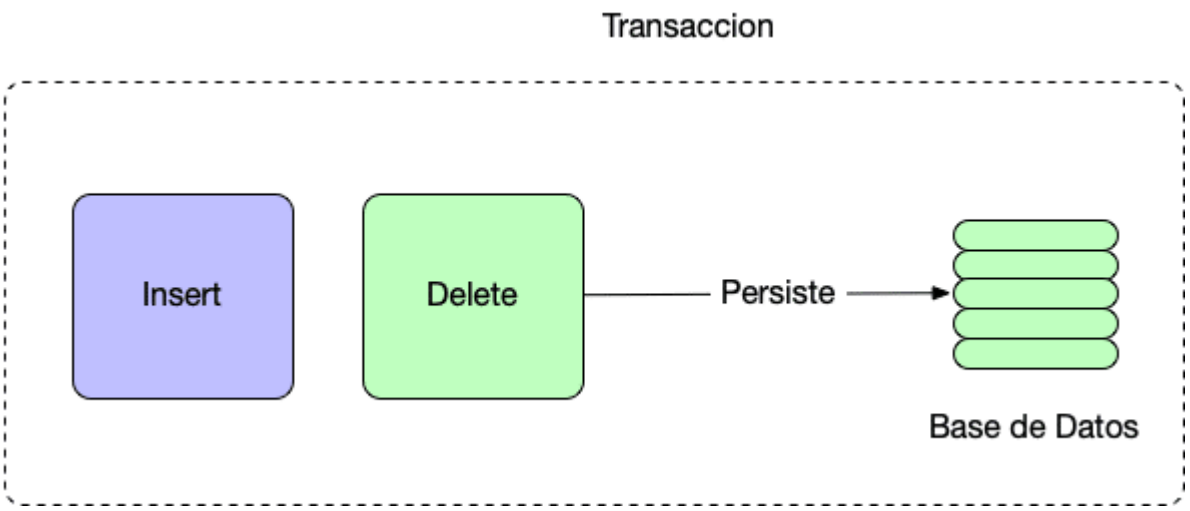


Isolated: Una transacción aparte de ser atómica esta aislada del resto y se ejecuta de forma independiente es decir no le afectan otros cambios que se produzcan en la base de datos.





Durable : Los cambios persisten con el tiempo es decir no son cambios “temporales” sino que cuando la transacción termina han sido persistidos en la base de datos y se se produce en ese momento un fallo , los datos ya han sido salvados a disco.



Otros Artículos relacionados

- JDBC, Java try with resources
- JDBC Prepared Statement y su manejo
- JDBC Batch y rendimiento