Transactions

CMPE 2400

Databases

Purpose

- transactions are used to guarantee that no changes will occur within a database unless all intended changes within a set of SQL statements have been completed successfully.

Banking Example

- If you deposit a cheque into an account, there are two distinct actions that must take place for the overall transaction to be successful.
  - The account from which the cheque was written must be debited for the amount of the cheque.
  - The account into which the cheque is being deposited must be credited for the amount of the cheque.

Inventing Money?

- If there is an error in the debiting process, and the crediting process still proceeds, then money has been created out of thin air.
- If there is an error in the crediting process after the debiting process has successfully completed, then money has been spontaneously destroyed.

Chaos

- If the system is not bulletproofed against such errors, then confidence in our electronic banking system is destroyed as it causes an uncertainty in the value of our currency.
- The banks would eventually fail under a lack of confidence as clientele flees to protect themselves and their hard-earned savings.

Transactions to the Rescue

- We need a system that will be able to ensure that if either the debiting action fails or the crediting action fails, then no changes will be made at all.

transactions

- In SQL, transactions were made especially for the purpose of preserving data integrity.
- In order for any changes to be permanently made within the database, all changes proposed within a transaction must be able to complete successfully.

transactions

- Up until this point, we have used autocommit transactions
- A database can be set to autocommit or require explicit commits
- Autocommit means that every SQL statement has been a transaction unto itself with two possible outcomes:
  - The statement completes successfully, and is thus pushed into the database.
  - The statement fails and is automatically prevented from making any changes to the database.
    - You may witness this happening by attempting to insert a row into a table with a value that has an illegal data type for the target column.
    - The row will not be partially inserted. Thus, either the *entire* statement is correct, or it *entirely* fails.

transaction

- Now, we will explicitly group several SQL statements into a single transaction.
- transactions may be used in global space with normal queries, but more often, we will see them used inside stored procedures.
- Data validation checks are not normally included within a transaction as they are not performing any changes to the stored data (unless you are using them to validate before performing an explicit commit)

Syntax

```
--pre-DML check on state of data
begin transaction
    begin try
    --DML operations
    end try
    begin catch
    rollback
    end catch
--within the transaction check on state of data
--commit if satisfied
--rollback if not satisfied
```