1. What is the name of a single logical operation on the data to satisfy ACID property?

Transaction. A Transaction can either be implicit such as an INSERT or UPDATE or explicit when defined to BEGIN TRANSACTION …. COMMIT/ROLLBACK TRANSACTION.

1. Which ACID property does the following DDLs satisfy?

CREATE TABLE Customer (CustomerID int PRIMARY KEY, CustomerName varchar(100) NOT NULL)

Consistency is and ACID property which states that database must be valid according to all defined rules, including constraints, cascades, triggers, and any combination thereof. In this case the DDL above creates a table with PK constraint, non-nullable columns. All these rules must be valid according to ACID property( specifically Consistentency)

1. Which ACID property ensures the integrity of data reads?

Isolation. Isolation ensures that concurrent execution of transactions leaves the database in the same state that would have been obtained if the transactions were executed sequentially.

1. Failure to write data to non-volatile memory violates which property?
   1. Atomicity
   2. Consistency
   3. Isolation
   4. Durability
2. State the reasons why concurrency control needed?

Concurrency is need because DBMS are designed to support multiple users and processes operating various set of tasks such as Inserting records, Updating records while other processes read from those records at the same time. If concurrency is uncontrolled, the following issues with be present: dirty reds, lost update problem, non-repeatable read, and the phantom read problem.

1. What is the difference between a local transaction and a distributed transaction?

A local transaction occurs within a database. Distributed transaction involves two or more databases, and often time distributed across the network(s).

1. When should you use the SAVE TRANSACTION statement?

A save point marks a specified point within the transaction so that all **updates that follow** can be canceled without canceling the entire transaction

1. Discuss the difference between row-level and page-level locking.

Locking means that the transaction marks the data that it accesses so that the DBMS knows not to allow other transactions to modify it until the first transaction succeeds or fails.

Row versioning provides each reading transaction the prior, unmodified version of data that is being modified by another active transaction. This allows readers to operate without acquiring locks, i.e., writing transactions do not block reading transactions, and readers do not block writers.

1. Can a user explicitly influence the locking behavior of the system?

YES, by either changing the isolation level using the SET TRANSACTION ISOLATION LEVEL statement or by using hints. For example, for databases in “READ COMMITED” Isolation Level, NOLOCK hint allows the user to read non-committed data which essentially skips the waiting of exclusive placed by the writer process