Q 6)

a.) The lock acquisition phase in 2PL is very crucial for accessing the data during transactions. The transaction requests and obtains all the necessary locks on the data items it needs to access and modify during this phase. This phase shall make sure that until the first transaction is finished, additional transactions cannot access or modify the same data items. If this is not done correctly, then the transaction may produce inconsistent or inaccurate data and break the isolation principle of the ACID properties for transactions.

b.) Prematurely releasing of the locks in 2PL can result in lost-updates scenario. This happens when the data item is updated by one transaction, but also the same data item is updated by another transaction before the first transaction has done a COMMIT on its modifications. The original transaction's modifications are thereby undone, and the data is now inconsistent. To avoid this situation, we can utilize a strict two-phase locking, which shall keep all locks in place until the transaction has either done a COMMIT or a ROLLBACK.

c.) Withouttheuseoflocks,transactionmanagementcanleadtodirtyreadswhichisa problem. When a transaction reads a piece of data that has already been updated by another transaction but hasn't been committed, this happens. It may produce inconsistent or inaccurate outcomes when the initial transaction bases its operations on the updated (but uncommitted) data. We can use optimistic concurrency control to resolve these types of problem, which entails allowing transactions to read and write data without restriction while checking for conflicts at commit time. All transactions commit only if they do not conflict with one another, so if a conflict is found, one of the transactions is rolled back & resumed.