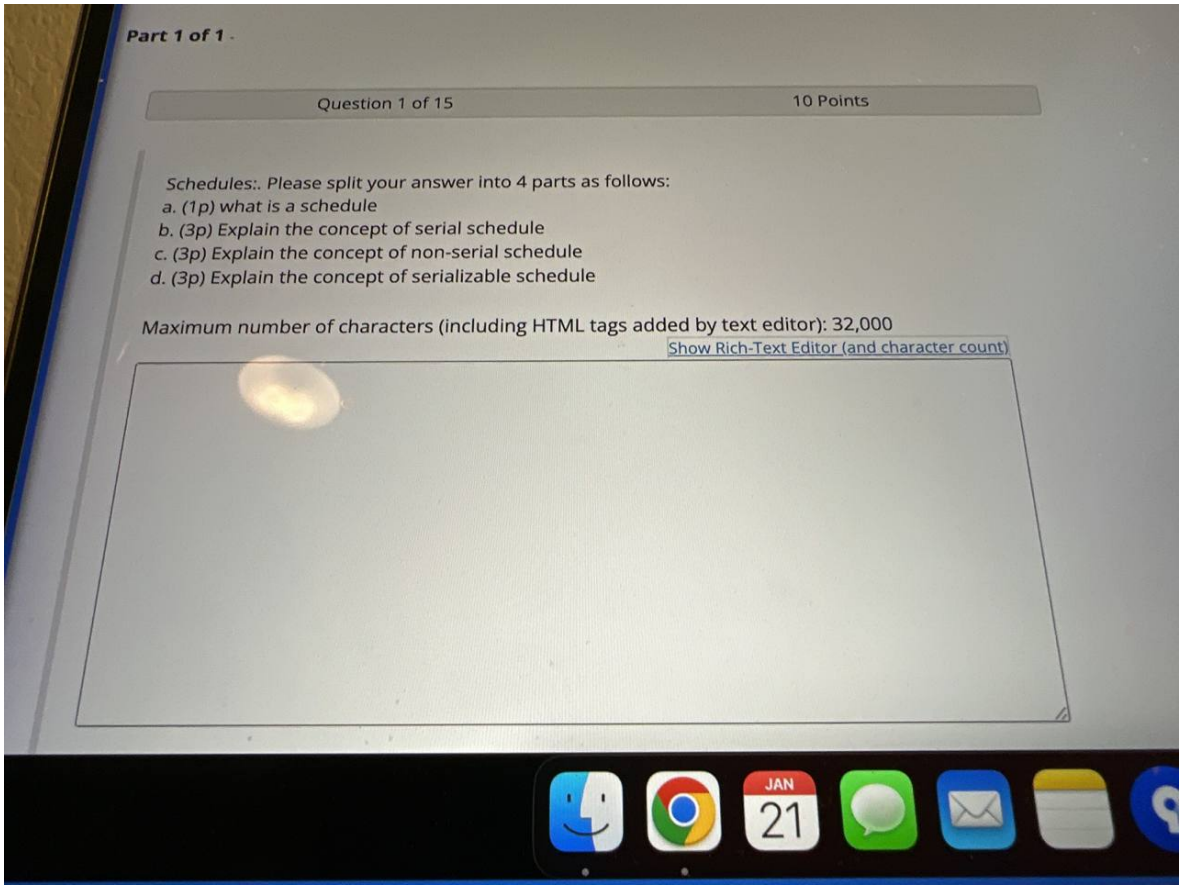
DBMS Finals

Please place the answers on the respective boxes:



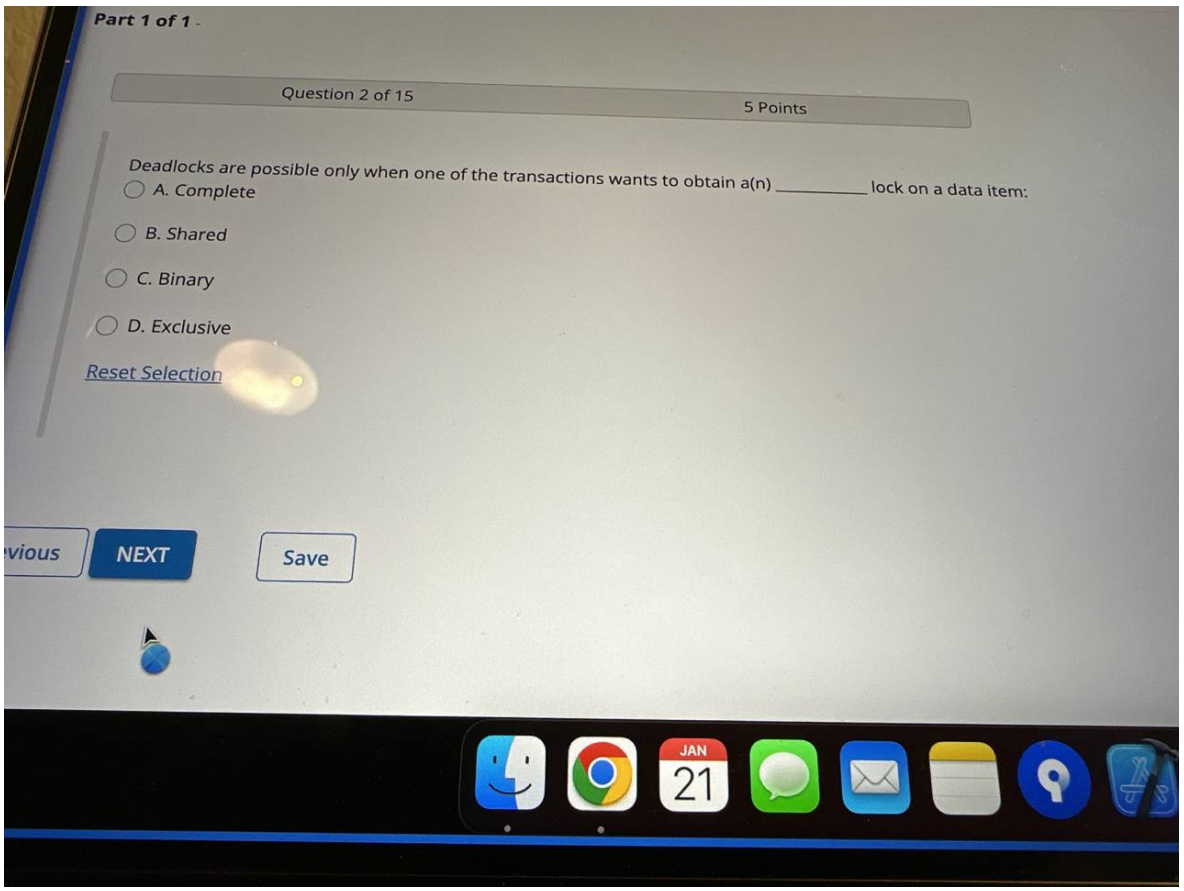
a) A schedule in a DBMS is the order in which operations are executed by the system. It is important to ensure that schedules are correct to maintain the consistency and integrity of the database through the use of concurrency control mechanisms.

b) A serial schedule is a schedule in which all transactions are executed one after the other, in a specific order, with no concurrent execution. It is considered to be the simplest and most straightforward type of schedule but not always desirable as it can cause poor performance in systems that need to handle many concurrent transactions.

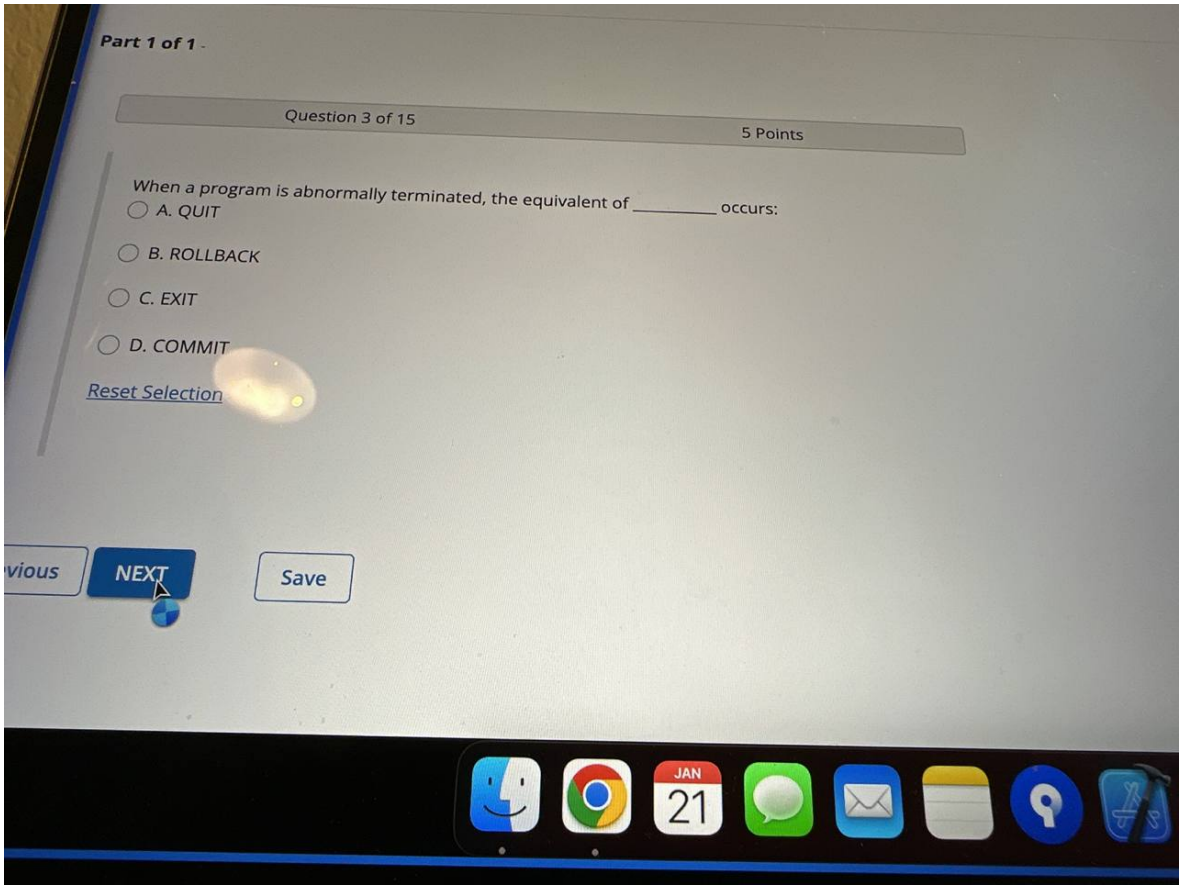
c) A non-serial schedule is a schedule in which multiple transactions are executed concurrently, potentially leading to conflicts and inconsistencies. Additional mechanisms, known as concurrency control, are required to ensure the consistency and integrity of the database. Non-serial schedules can increase performance, but also increase the complexity of the system.

d) A serializable schedule is a schedule in which the concurrent execution of

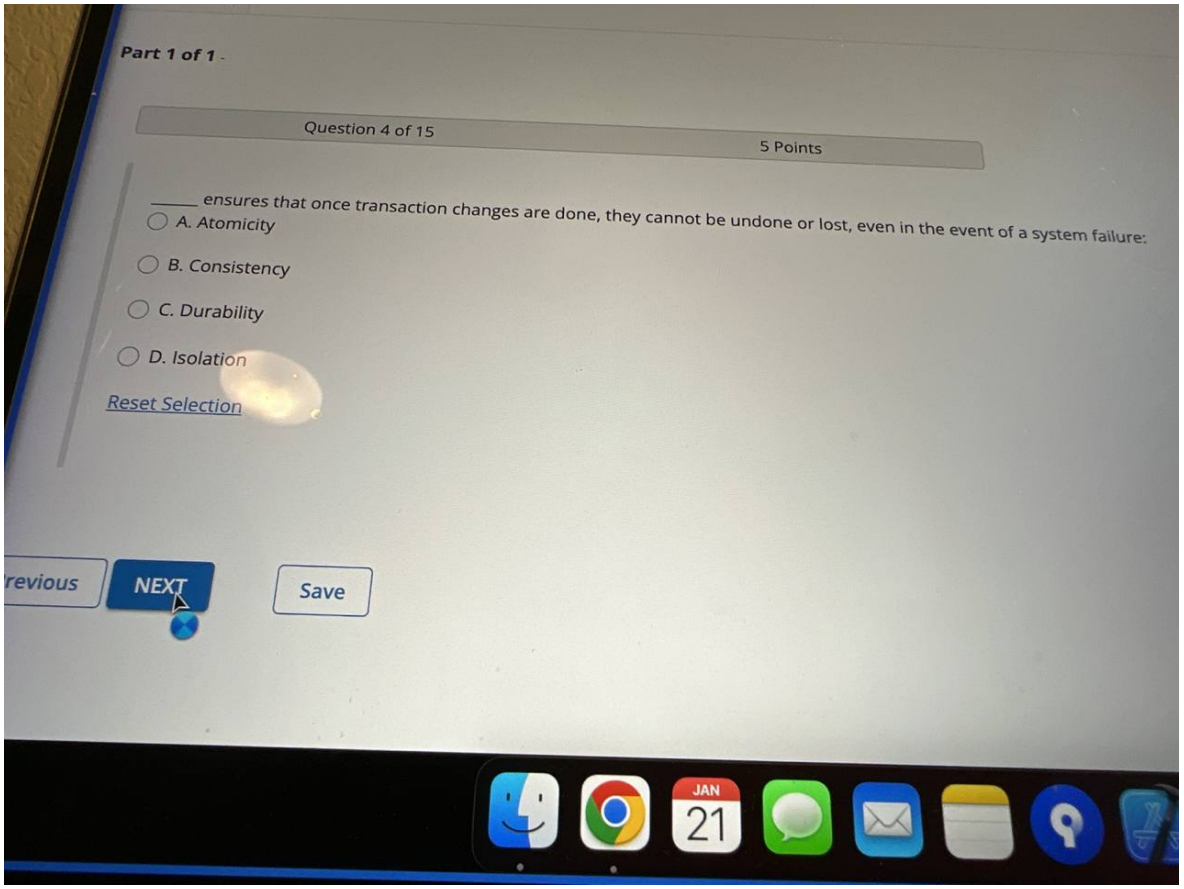
multiple transactions preserves the same result as if they were executed serially. It guarantees that the database remains in a consistent state and prevents conflicting updates made by different transactions. Concurrency control techniques such as locking, timestamp ordering and multi-version control are used to achieve serializability.



Ans: Exclusive


Part 1 of 1
Question 3 of 15
5 Points
When a program is abnormally terminated, the equivalent of _____ occurs:
A. QUIT
B. ROLLBACK
C. EXIT
D. COMMIT
Reset Selection
NEXT
Save
JAN
21

A. Rollback



C

Q. _____ ensures that once transaction changes are done, they cannot be undone or lost, even in the event of a system failure.

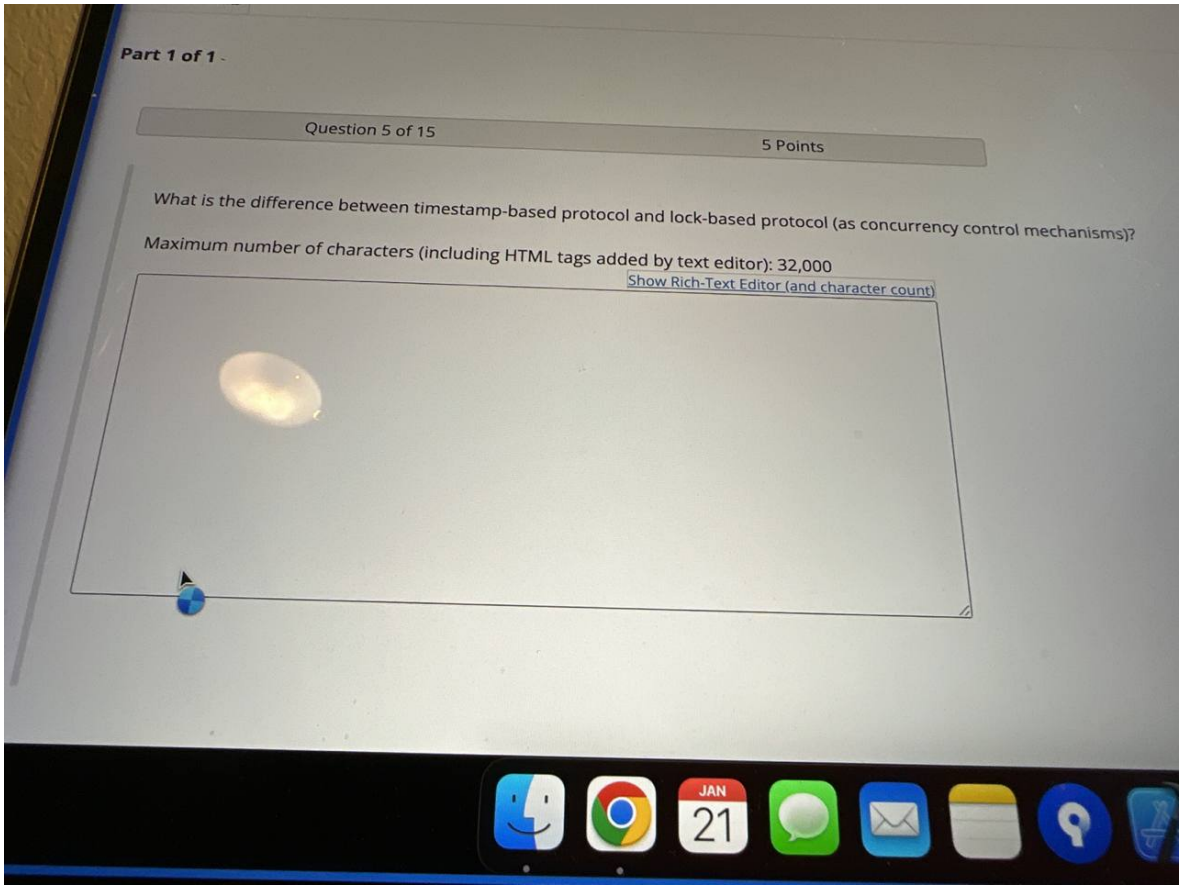
A. Atomicity

B. Consistency

C. Durability

D. Isolation

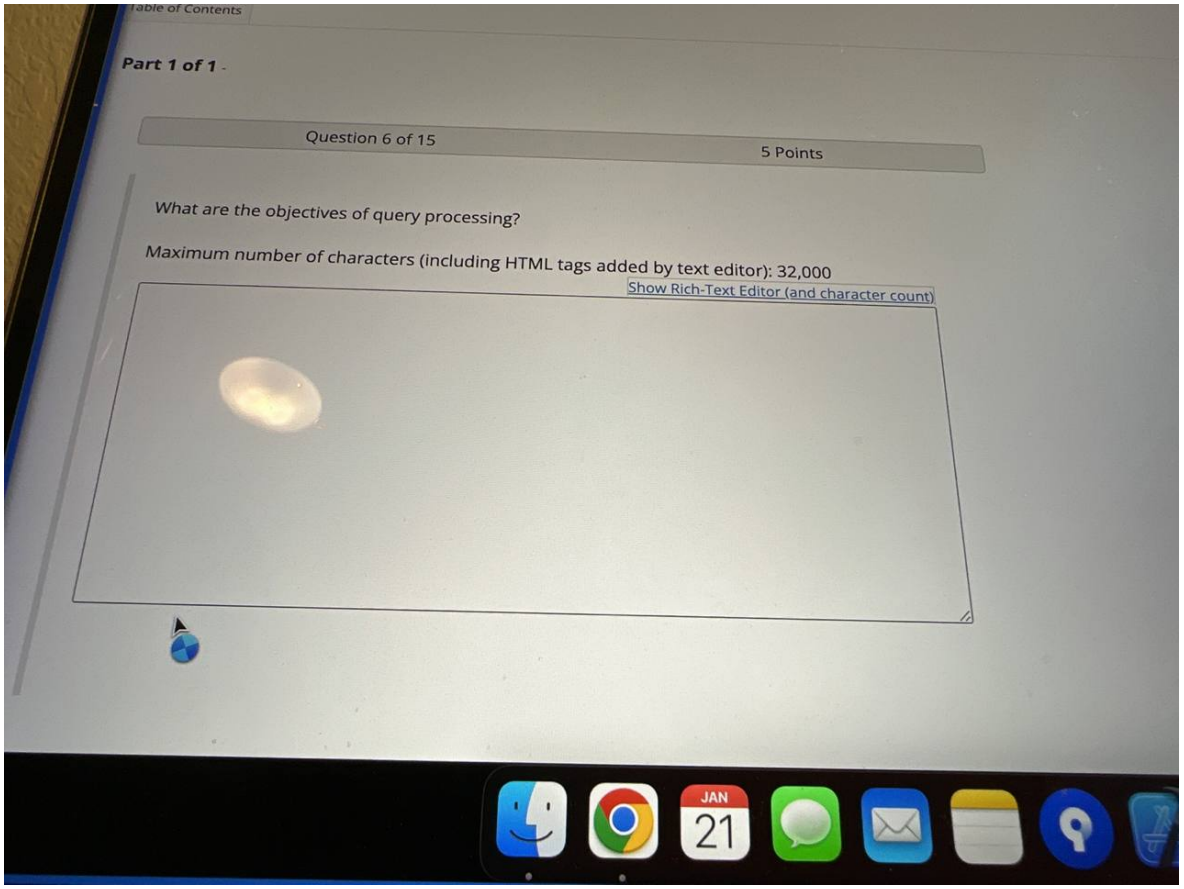
Answer» C. Durability



Timestamp based Protocol: Timestamp-based protocols utilize timestamps to maintain consistency in a distributed system by assigning a timestamp to each transaction and executing them in the order of their timestamps. This method is commonly applied in distributed databases.

Lock based protocol: Lock-based protocols use locks to ensure consistency in a distributed system by granting exclusive access to a shared resource to a single transaction at a time. Other transactions are not allowed to access the resource until it is unlocked. This method is commonly used in systems that need to prevent conflicts and data inconsistencies resulting from concurrent access to shared resources.

Alternative:
Timestamp-based protocols differ from locking-based protocols in that they do not use locks to control access to the data. Instead, they rely on the timestamps to determine the order in which transactions should be executed. This makes timestamp-based protocols more efficient than locking-based protocols, as they do not require the overhead of acquiring and releasing locks.



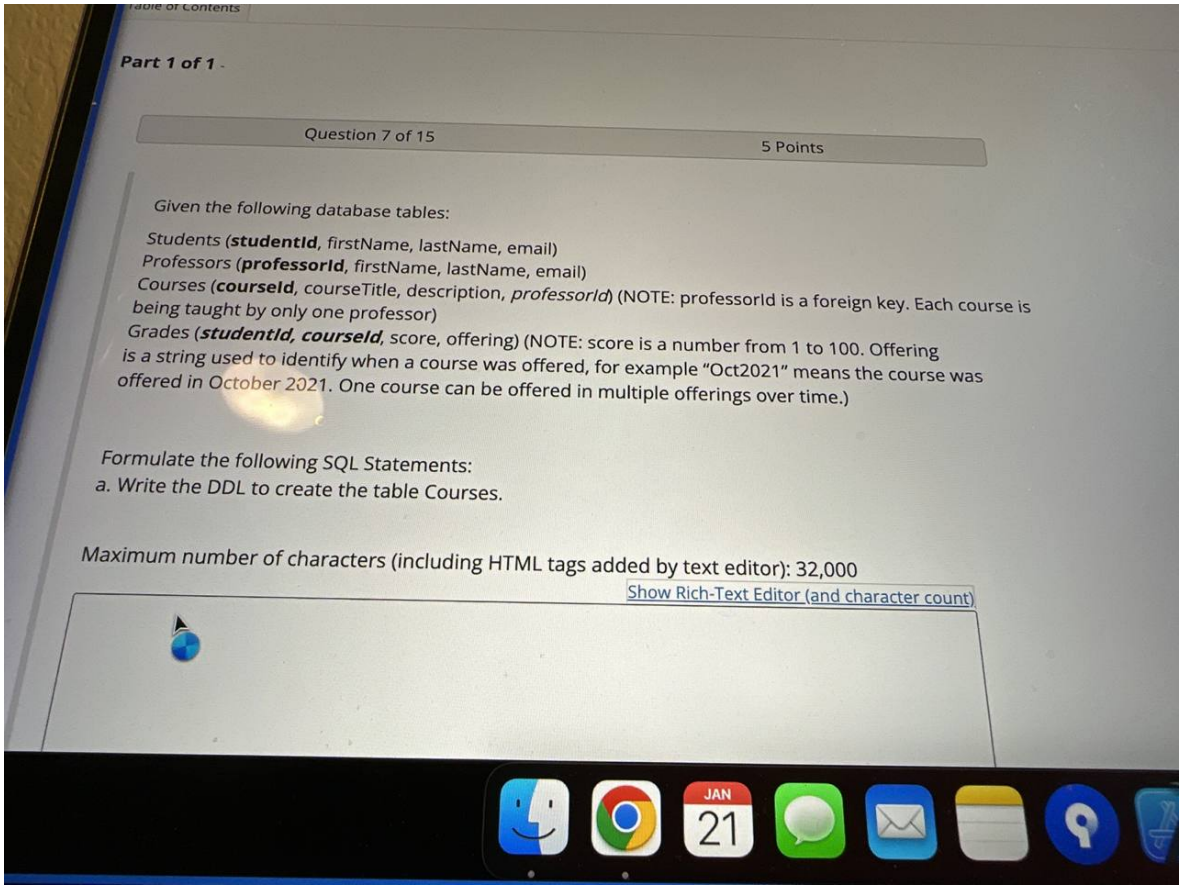
The main objectives of query processing in a DBMS are to:

1. Optimize the performance of the query by selecting the most efficient execution plan.
2. Ensure that the correct output is returned to the user by correctly executing the query according to the syntax and semantics of the query language.
3. Provide security and data privacy by enforcing the access control policies and ensuring that the user is only able to access the data they are authorized to see.
4. Handle errors and exceptions in a graceful manner and provide appropriate feedback to the user.
5. Provide support for concurrent access to the data by multiple users by using concurrency control mechanisms.
6. Scalability to handle large amount of data and many concurrent queries.
7. Optimize the use of resources such as memory, disk space and network bandwidth.
8. Provide support for different types of data and query languages.

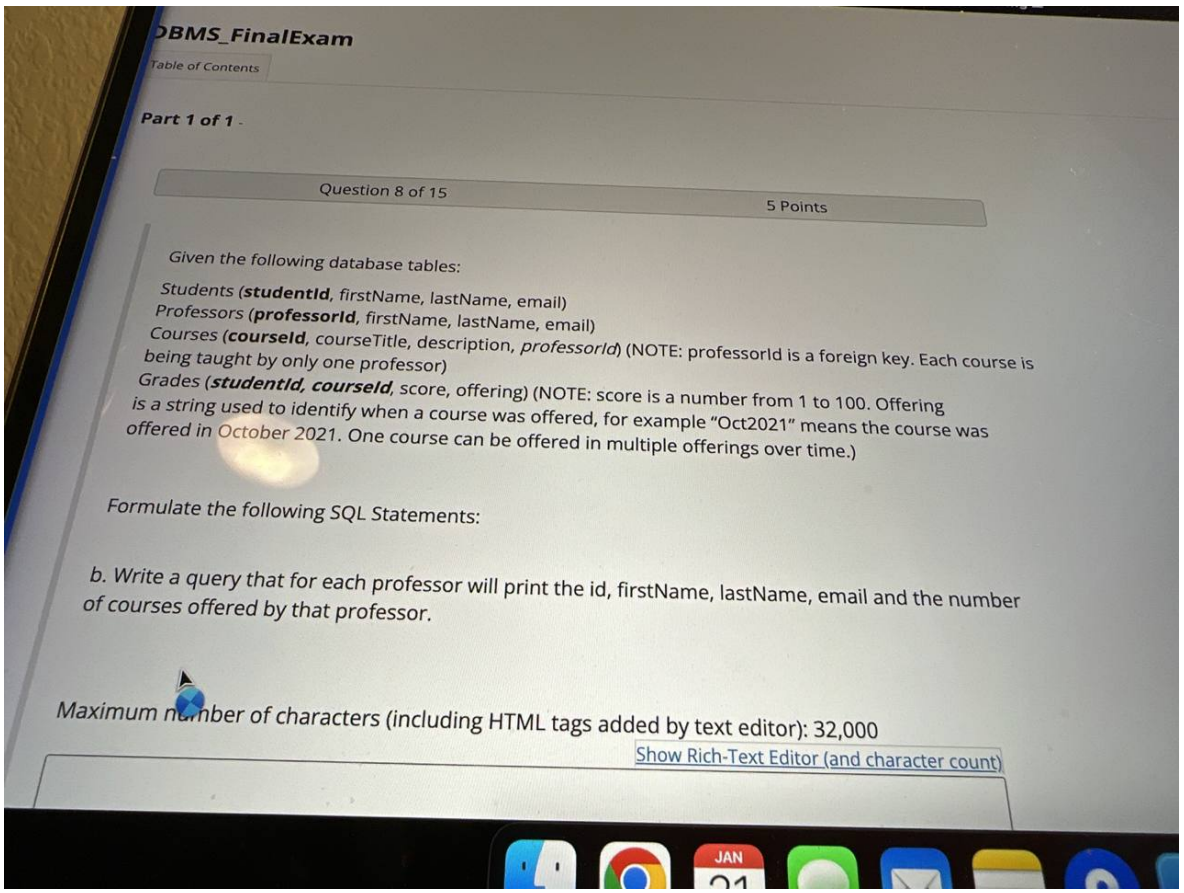
Alternative:

The main aim of query processing is to transform a query written in a high-level language, typically SQL, into a correct and efficient execution strategy expressed in a low-level language like the relational algebra, and to execute the strategy to retrieve the required

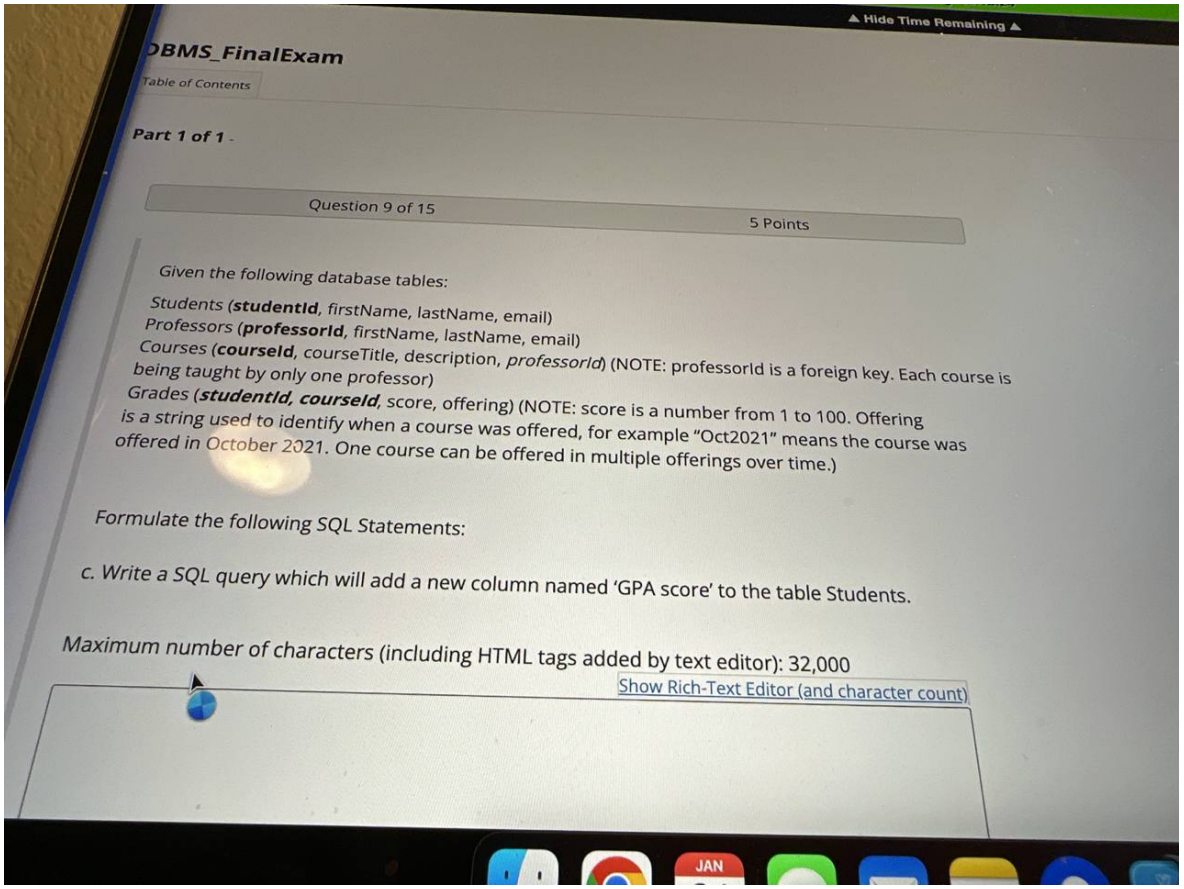
data. The other objectives of query processing are as follows:
a. To execute the queries efficiently: The techniques such as indexing, optimization, and parallel processing can be used by DBMS to execute queries quickly and efficiently.
b. To generate accurate results: DBMS should be able to give the correct result for any given query.
c. To ensure data integrity: Every database system should ensure the database is accurate, consistent, and complete.
d. To support concurrent access: DBMS must support multiple users accessing the same data simultaneously ensuring the data is consistent and accurate.
e. To provide security: DBMS should ensure that only authorized users can access the database.



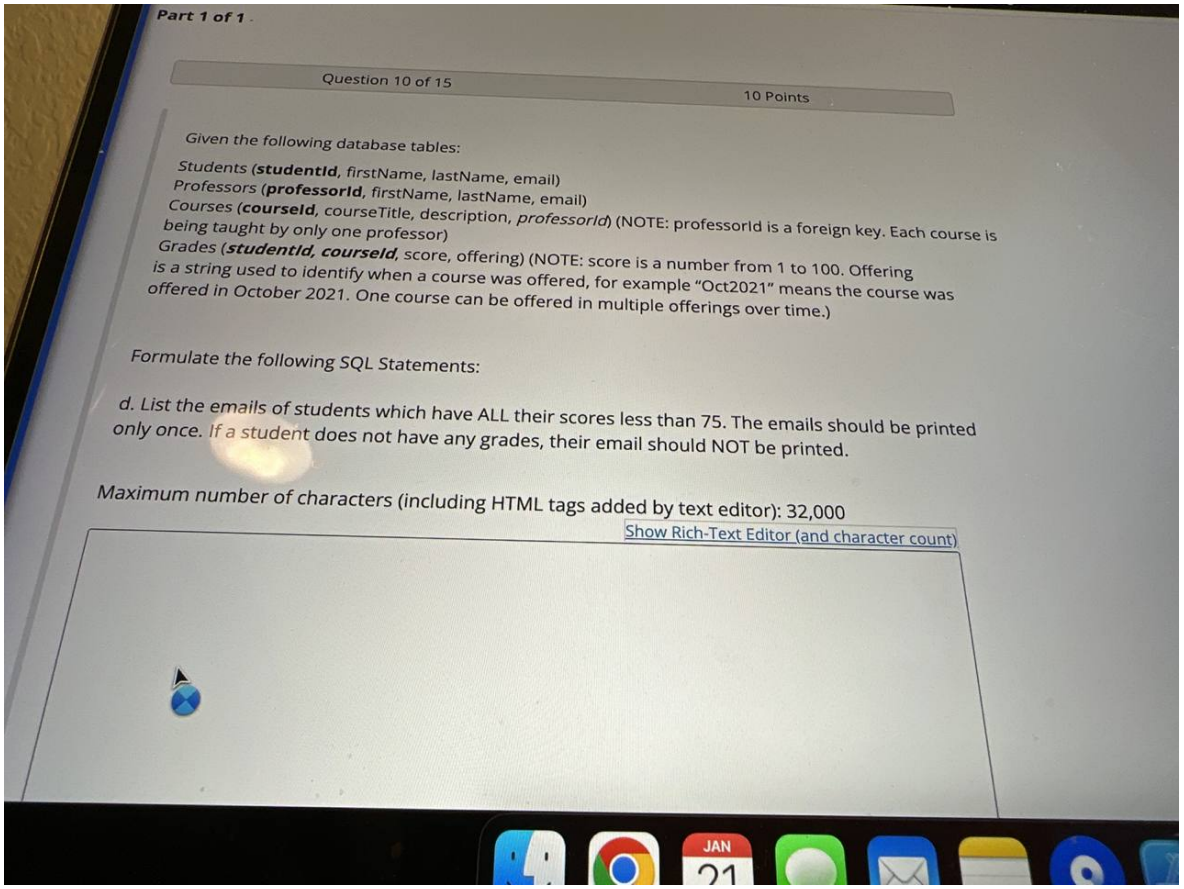

```
CREATE TABLE Courses (
courseId INT PRIMARY KEY,
courseTitle VARCHAR(255),
description TEXT,
professorId INT,
FOREIGN KEY (professorId) REFERENCES Professors(professorId)
);
```



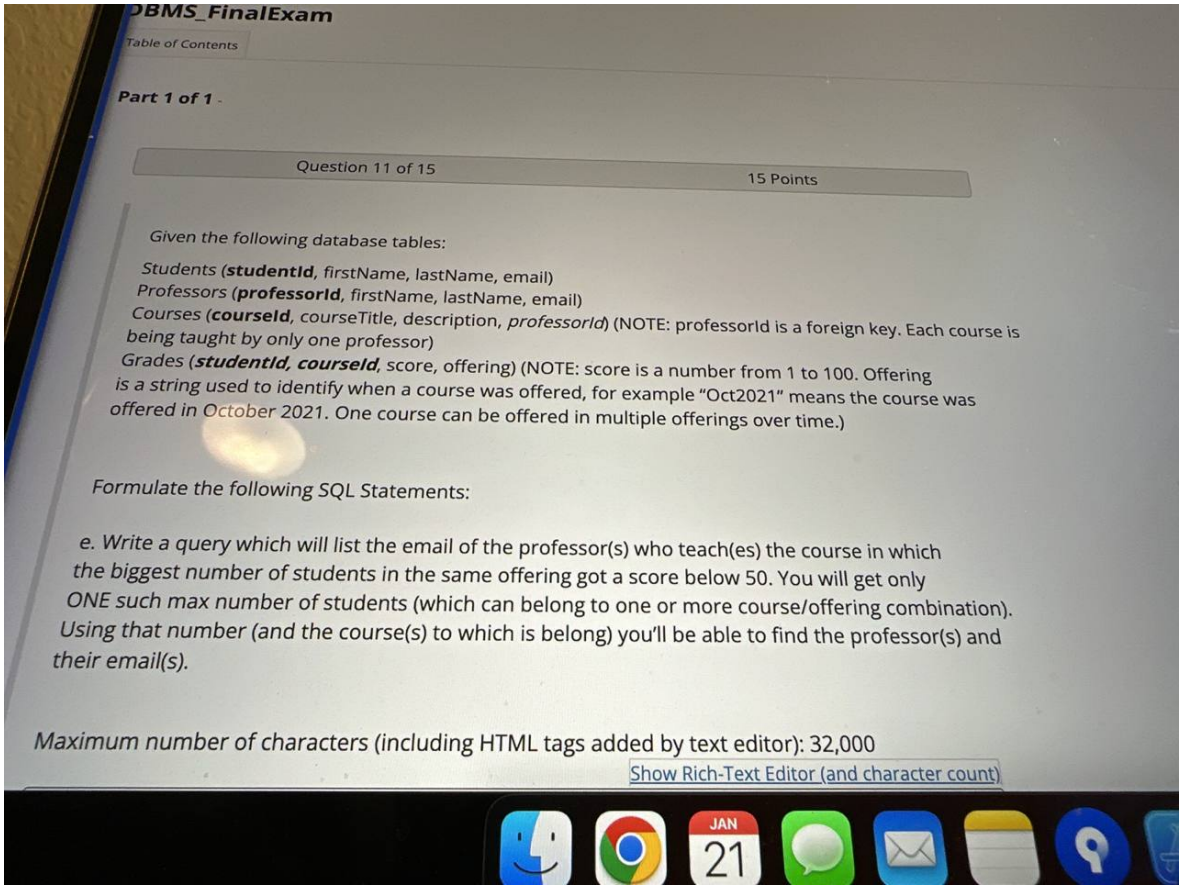

```
SELECT P.professorId, P.firstName, P.lastName, P.email, COUNT(C.courseId) as num_courses
FROM Professors P
JOIN Courses C ON P.professorId = C.professorId
GROUP BY P.professorId, P.firstName, P.lastName, P.email;
```



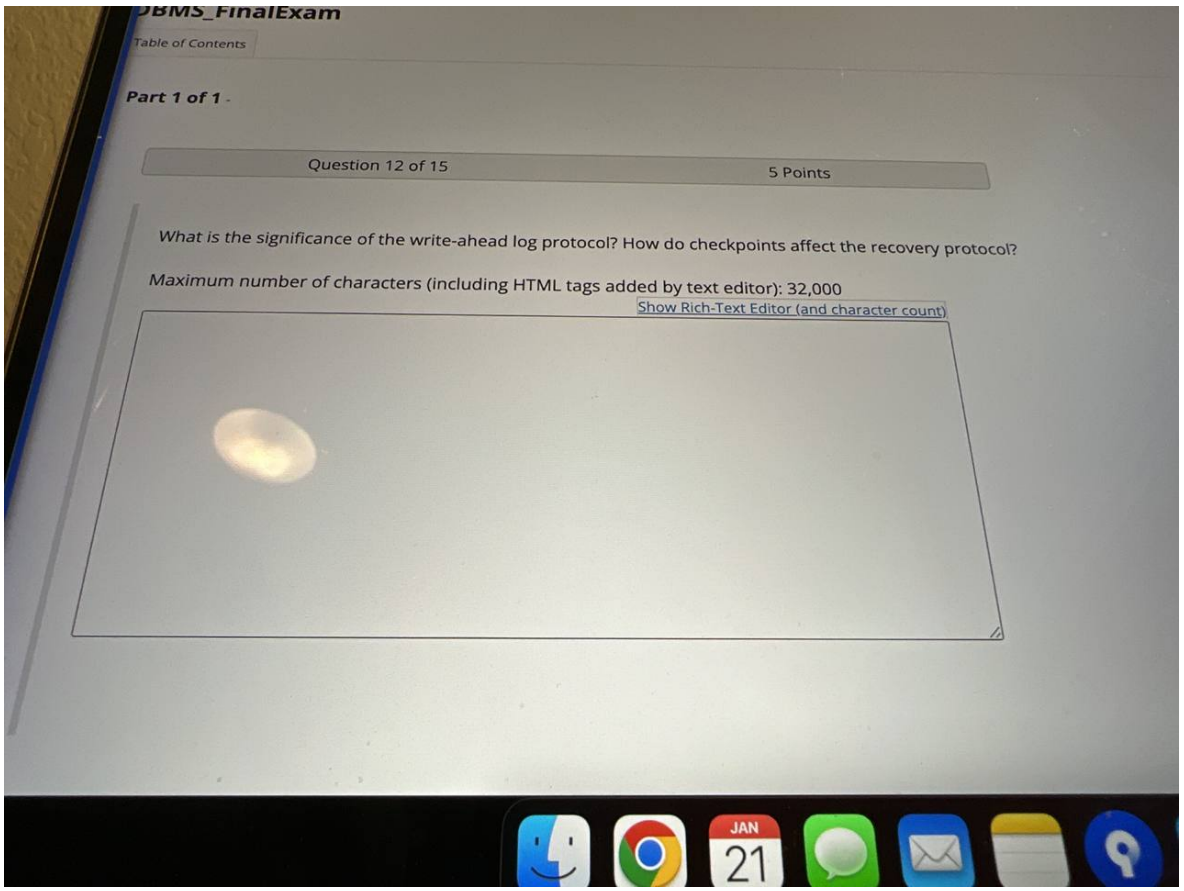
ALTER TABLE Students
ADD COLUMN GPAScore DECIMAL(4,2);




```
SELECT DISTINCT S.email
FROM Students S
JOIN Grades G ON S.studentId = G.studentId
WHERE G.score < 75
GROUP BY S.email
HAVING COUNT(DISTINCT G.courseId) = (SELECT COUNT(*) FROM Courses);
```




```
SELECT P.email
FROM Professors P
JOIN Courses C ON P.professorid = C.professorid
JOIN (
SELECT G.offering, COUNT(*) as num_below_50
FROM Grades G
JOIN Courses C ON G.courseId = C.courseId
WHERE G.score < 50
GROUP BY G.offering
ORDER BY num_below_50 DESC
LIMIT 1
) as max_below_50 ON C.courseId = max_below_50.courseId
;
```

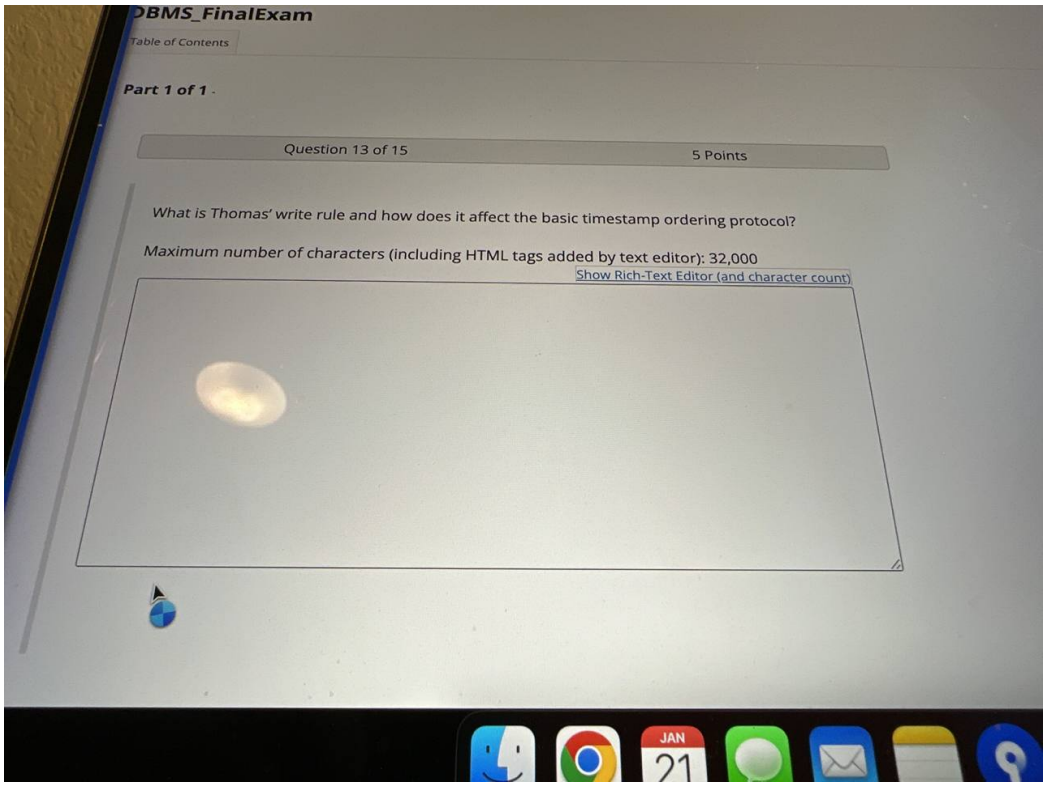


The write-ahead log (WAL) protocol is a technique used in database systems to ensure data durability and consistency. It works by writing all changes to the database to a separate log file before they are made to the actual data files. This log file is used to track all changes made to the database and can be used to recover the database in case of a crash or other failure. The WAL protocol ensures that a database can be recovered to a consistent state in the event of a crash or other failure. It does this by providing a record of all changes made to the database, so that the database can be reconstructed to the state it was in at the time of the failure. Checkpoints are a mechanism used in database systems to periodically save the state of the database to disk. This is done to minimize the amount of work that needs to be done to recover the database in case of a crash or other failure. Checkpoints are typically done at regular intervals, such as every hour or every day. When a checkpoint is taken, the database system writes the current state of the database to disk, along with a record of the position in the WAL up to which all changes have been applied. This allows the database system to quickly recover to the state it was in at the time of the checkpoint, without having to process the entire WAL. Checkpoints also affect the recovery protocol by reducing the amount of work needed to recover the database after a crash or other failure. Because the database system can recover to the state it was in at the time of the last checkpoint, it can avoid having to process the entire WAL. This can significantly speed up the recovery process, especially if the WAL is very large or if the database system has been running for a long period of time.

Alternative:
Write ahead log protocol is a mechanism for data recovery ensuring the consistency of a database. In this protocol, all updates to the database must be written to the log file before they are written to the database itself. This makes sure that the log file has a complete and accurate records of all updates made to the database. So, such log file can be used to restore the database to a consistent state after a failure.

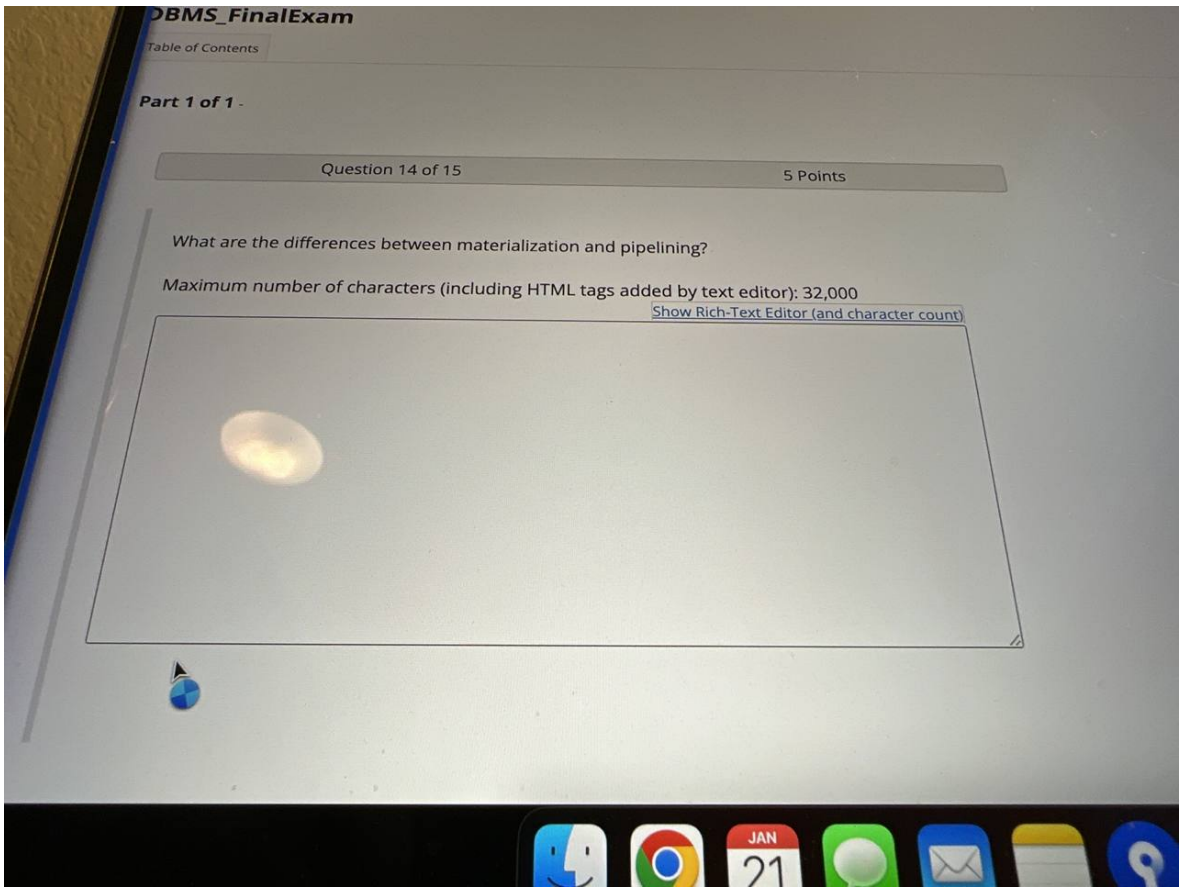
Checkpoint is a consistent state of the database that is saved to disk at a particular point in time. These are a prime key for write ahead protocol. Indeed, checkpoints help us reducing the amount of work that is needed to recover the database after a failure as it contains a complete and accurate records of all updates made to the database. It makes the recovery process more efficient.

- The write-ahead log (WAL) protocol is a technique used to ensure the durability and consistency of data by writing information about each transaction to a log file before the actual changes are made to the data files.
- Checkpoints are used to periodically save the state of the database and the WAL protocol.
- Checkpoints affect the recovery protocol by allowing the database to be recovered to a more recent and consistent state by rolling forward all the changes in the WAL after the last checkpoint, reducing the amount of work needed to recover the database.



Thomas' write rule is a concurrency control protocol that assigns a unique

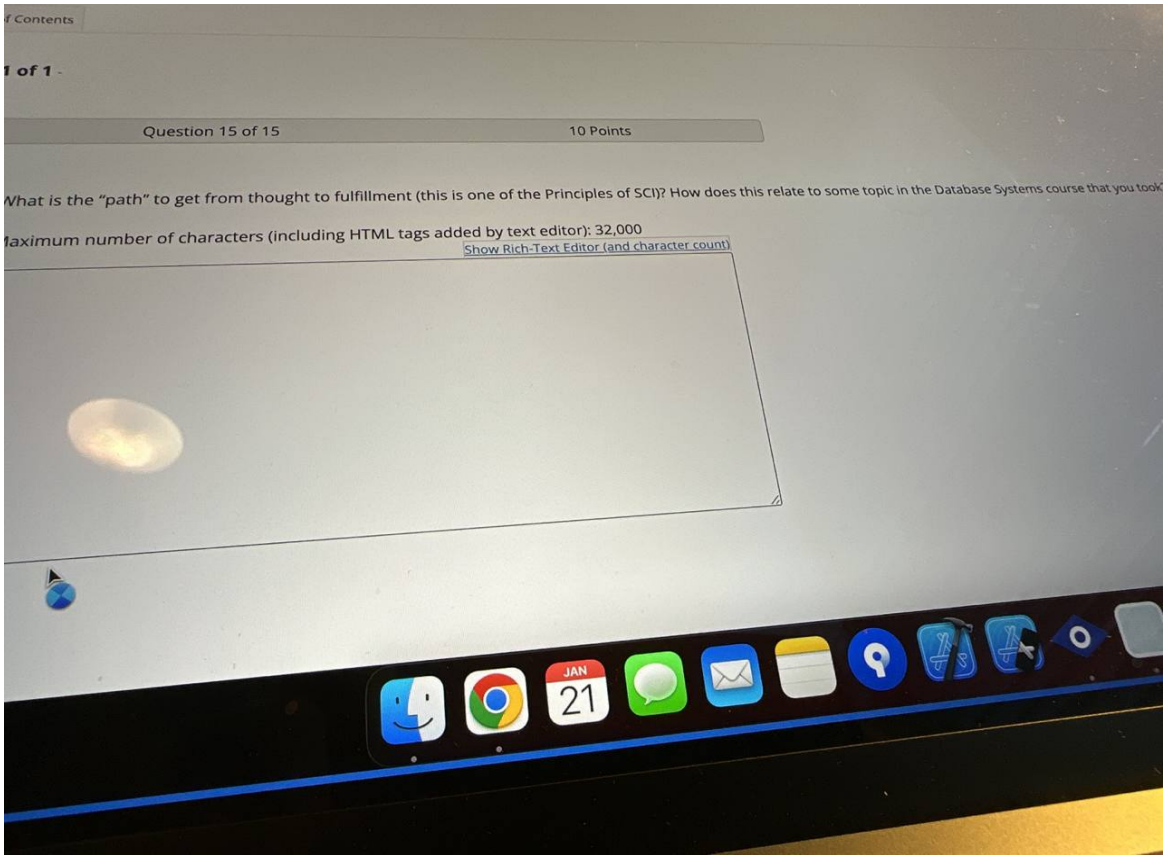
timestamp to each transaction and ensures that a transaction can only write to a data item if its timestamp is greater than the timestamp of the last transaction that wrote to that data item. This protocol is used to ensure the serializability of transactions and preserve the consistency of the data by preventing conflicting writes to shared resources. It is also known as "write-timestamp ordering" or "timestamp ordering" and it affects the basic timestamp ordering protocol by ensuring that the writes of a transaction to a shared resource are done in timestamp order.



Pipelining	Materialization
It is a modern approach to evaluate multiple operations.	It is a traditional approach to evaluate multiple operations.
It does not use any temporary relations for storing the results of the evaluated operations.	It uses temporary relations for storing the results of the evaluated operations. So, it needs more temporary files and I/O.
It is a more efficient way of query evaluation as it quickly generates the results.	It is less efficient as it takes time to generate the query results.
It requires memory buffers at a high rate for generating outputs. Insufficient memory buffers will cause thrashing.	It does not have any higher requirements for memory buffers for query evaluation.
Poor performance if trashing occurs.	No trashing occurs in materialization. Thus, in such cases, materialization is having better performance.
It optimizes the cost of query evaluation. As it does not include the cost of reading and writing the temporary storages.	The overall cost includes the cost of operations plus the cost of reading and writing results on the temporary storage.

Alternative:
Materialization:
• This is the process of creating and storing an intermediate result in disk temporarily.
• It may be slower than pipelining.
• It requires additional memory to store the intermediate result.
• Intermediate result can be used by other queries.
• It is simpler to implement as it does not need the operators to process the data in a stream.

Pipelining:
• This is the process of passing the intermediate result directly from one operation to the
next, without storing it in a temporary relation. It is sometimes known as stream-based
processing or on-the-fly processing.
• It is faster than materialization.
• It does not require additional memory as it does not store intermediate result.
• There is no intermediate result with pipelining, so is only available to the operators in
the pipeline.
• It is comparatively complex as it needs the operators to process the data in a pipeline.


f Contents
1 of 1 -
Question 15 of 15
10 Points
What is the "path" to get from thought to fulfillment (this is one of the Principles of SCI)? How does this relate to some topic in the Database Systems course that you took
Maximum number of characters (including HTML tags added by text editor): 32,000
Show Rich-Text Editor (and character count)
JAN
21

Systems course that you took?

10. **(5 points)**. What is the "path" to get from thought to fulfillment (this is one of the Principles of SCI)? How does this relate to some topic in the Database Systems course that you took?

ANS:

1. "path" to get from thought to fulfillment:

Knowledge is for Action. Action is for Achievement. Achievement is for **Fulfillment**.

(from http://a-dc.org/raam/bitraam/sci/16Principles/index.htm)

2. How this relate to some topic in the Database Systems course that I took:

A distributed database is a collection of multiple, logically interrelated collections of shared data (and a description of this data), physically distributed over a computer network yet appearing as an undifferentiated wholeness to the user.

The DDBMS should appear like a centralized DBMS by providing distribution transparency, transaction transparency, performance transparency and DBMS transparency. The idea here is that the DDBMS, not the user, is doing all the work of organizing the parts into a whole. In a state of enlightenment, one's desires are carried out by Nature and **fulfillment** of one's desires is effortless.

(from: Main Points 12.htm)