DATABASE TRANSACTIONS
TYPE A: Very Short Answer Questions

1	Define a transaction.
Ans.	Transaction is a logical unit of work that must succeed or fail in its entirety.
2	What do you mean by committing a transaction?
Ans.	Committing a transaction means all the steps of a transaction are carried out successfully and all data changes are made permanent in the database.
3	What does transaction ROLLBACK indicates?
Ans.	Transaction ROLLBACK means transaction has not been finished completely and hence all data changes made by the transaction in the database are undone. And the database returns to the same state as it was before this transaction execution started.
4	What are the different properties of transactions maintained by database systems?
Ans.	Following are the properties of transactions maintained by database systems: 1. Atomicity 2. Consistency 3. Isolation 4. Durability.
5	Define the following terms: (i) Atomicity (ii) Consistency (iii) Isolation (iv) Durability.
Ans.	(i) **Atomicity** –This property ensures that either all operations of the transaction are reflected properly in the database, or none are. Atomicity ensures either all-or-none operations of a transaction are carried out. (ii) **Consistency** –This property ensures that the database remains in a consistent state before the start of transaction and after the transaction is over (whether successfully or unsuccessfully). (iii) **Isolation** –Isolation ensures that each executing transaction execution in isolation i.e., is unaware of other transactions executing concurrently in the system. (iv) **Durability** –This property ensures that after the successful completion of a transaction i.e., when a transaction COMMITs, the changes made by it to the database persist i.e., remain in the database irrespective of other system failures.
6	What are the two ways in which multiple transactions can be executed?
Ans.	Multiple transaction can be executed in one of the following two ways: 1. Serially i.e., serial Execution of transactions. 2. Concurrently i.e., Concurrent (simultaneous) execution of transactions.
7	What are the advantages of serial execution?
Ans.	In a serial execution transactions are executed strictly serially. Thus, Transaction T1 completes and writes its results to the database then only the next transaction T2 is scheduled for execution. This means at one time there is only one transaction is being executed in the system. 1. Correct execution, i.e., if the input is correct then output will be correct. 2. Fast execution since all the resources are available to the active.
8	Name ACID properties.
Ans.	1. **A**tomicity 2. **C**onsistency 3. **I**solation 4. **D**urability.
9	What is savepoint?
Ans.	Savepoint is a point transaction, uptil which all changes have been saved permanently.
10	What do following commands do? (i) COMMIT (ii) ROLLBACK (iii) SAVEPOINT

Ans.	**(i) COMMIT –** It explicitly ends the transaction and stores the changes (in memory) permanently in the database.
	(ii) ROLLBACK – It explicitly ends the running transaction and undoes the changes in data (in memory) i.e., roll back.
	(iii) SAVEPOINT – Point to a transaction uptil which all changes have been saved permanently.

TYPE B: Short Answer Questions

1	**What is the concept of database transaction?**
Ans.	A **Database transaction** is a logical unit of work (LUW) that must succeed or fail in its entirety. This statements means that a transaction may involve many sub-steps, which should either all be carried out successfully or all is ignored if some failure occurs.
	Each transaction, generally, involves one or more data manipulation language (DML) statements (such as INSERT, DELETE or UPDATE) and ends with either a COMMIT to make the changes permanent or ROLLBACK to undo the changes. The database transaction must be handled in a way so that their integrity is maintained.
2	**Briefly explain the system view of the transaction.**
Ans.	The system (data base software) views a transaction as a logical sequence of read and write operations, where
	1. **Read (X) –** operation brings data item X from database to an area in the main memory where the transaction is executing.
	2. **Write (X) –** operation transfers data item X from transaction's main memory area to the database file.
3	**What is the function of redo and undo logs?**
Ans.	DBMS maintains special logs to perform redo and undo operations if required.
	A storage area that holds copies of data modified by active transactions. By default, this area is physically part of the system tablespace. Redo log stores correct data written by incomplete transactions. And used for redo operations.
4	**Discuss briefly the ACID properties and their importance in transaction handling?**
Ans.	**Atomicity –**This property ensures that either all operations of the transaction are reflected properly in the database, or none are. Atomicity ensures either all-or-none operations of a transaction are carried out.
	Consistency –This property ensures that the database remains in a consistent state before the start of transaction and after the transaction is over (whether successfully or unsuccessfully).
	Isolation –Isolation ensures that each executing transaction execution in isolation i.e., is unaware of other transactions executing concurrently in the system.
	Durability –This property ensures that after the successful completion of a transaction i.e., when a transaction COMMITs, the changes made by it to the database persist i.e., remain in the database irrespective of other system failures.
5	**What are the various advantages and limitations of serial execution of transactions?**
Ans.	**Advantages**
	1. Correct execution, i.e., if the input is correct then output will be correct.
	2. Fast execution since all the resources are available to the active.
	Limitation
	1. The worst thing about serial execution is very inefficient resource utilization.
	2. Waiting time time is high as only one transaction executes at a time.
6	**Why do we use ROLLBACK statement? Explain in brief with the help of an example.**
Ans.	The ROLLBACK statement is used to undo transactions that have not already been saved to the database.
	The ROLLBACK command can only be used to undo transactions since the last COMMIT or ROLLBACK command was issued.
	The syntax for ROLLBACK command is as follows:
	ROLLBACK;
	Example:
	Consider the CUSTOMERS table having the following records:

```
+----+------------+-------+----------------+-------------+
| ID | NAME    | AGE | ADDRESS     | SALARY  |
+----+------------+-------+-----------------+-------------+
|  1 | Ramesh  | 32  | Ahmedabad | 2000.00  |
```

2	Khilan	25	Delhi	1500.00
3	kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
6	Komal	22	MP	4500.00
7	Muffy	24	Indore	10000.00
+----+----------+-----+----------+----------+

Following is the example, which would delete records from the table having age = 25 and then ROLLBACK the changes in the database.

SQL> DELETE FROM CUSTOMERS
     WHERE AGE = 25;
SQL> ROLLBACK;

As a result, delete operation would not impact the table and SELECT statement would produce the following result:

+----+----------+-----+----------+----------+
| ID | NAME | AGE | ADDRESS | SALARY |
+----+----------+-----+----------+----------+
1	Ramesh	32	Ahmedabad	2000.00
2	Khilan	25	Delhi	1500.00
3	kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
6	Komal	22	MP	4500.00
7	Muffy	24	Indore	10000.00
+----+----------+-----+----------+----------+

7	**What is transaction? What TCL commands are supported by SQL?**
Ans.	**Transaction** is a logical unit of work that must succeed or fail in its entirety. Following are the TCL commands are supported by SQL: 1. BEGIN | START TRANSACTION 2. COMMIT 3. ROLLBACK 4. SAVEPOINT 5. SET AUTOCOMMIT

Type C: Long Answer Questions

1	**Discuss the user view and system view of a transaction.**

Ans.	1. **User view of the transaction –** A user view transaction in terms of work carried out by it i.e., the function(s) performed by the transaction. For example, the user views above mentioned transaction T1 as follows:
	Begin transaction
	    Get balance from account X
	    calculate new balance as X – 1000
	    store new balance into database file
	    get balance from account Y
	    calculate new balance as Y + 1000
	    store new balance into database file
	End transaction
	2. **System View of the transaction –** The system (data base software) views a transaction as a logical sequence of read and write operations, where
	   1. Read (X) – operation brings data item X from database to an area in the main memory where the transaction is executing.

	2. Write (X) – operation transfers data item X from transaction's main memory area to the database file.
2	**Discuss various properties of the transactions. Why are these considered very important?**

Ans.	**(v)** **Atomicity –**This property ensures that either all operations of the transaction are reflected properly in the database, or none are. Atomicity ensures either all-or-none operations of a transaction are carried out.
	
	 (vi) **Consistency –**This property ensures that the database remains in a consistent state before the start of transaction and after the transaction is over (whether successfully or unsuccessfully).
	
	 (vii) **Isolation –**Isolation ensures that each executing transaction execution in isolation i.e., is unaware of other transactions executing concurrently in the system.
	
	 (viii) **Durability –**This property ensures that after the successful completion of a transaction i.e., when a transaction COMMITs, the changes made by it to the database persist i.e., remain in the database irrespective of other system failures.
	
	 Showing importance of transaction by example - Transaction to transfer ₹50 from account A to account B:
	     1. read(A)
	     2. A := A – 50
	     3. write(A)
	     4. read(B)
	     5. B := B + 50
	     6. write(B)
	
	 Atomicity requirement — if the transaction fails after step 3 and before step 6, the system should ensure that its updates are not reflected in the database, else an inconsistency will result.
	 Consistency requirement – the sum of A and B is unchanged by the execution of the transaction.
	 Isolation requirement — if between steps 3 and 6, another transaction is allowed to access the partially updated database, it will see an inconsistent database (the sum A + B will be less than it should be).
	 • Isolation can be ensured trivially by running transactions serially, that is one after the other.
	 • However, executing multiple transactions concurrently has significant benefits, as we will see later.
	 Durability requirement — once the user has been notified that the transaction has completed (i.e., the transfer of the ₹50 has taken place), the updates to the database by the transaction must persist despite failures.

3	Given a table t3 (code, grade, value). <table><tr><td>1</td><td>G</td><td>300</td></tr><tr><td>2</td><td>K</td><td>600</td></tr><tr><td>3</td><td>B</td><td>200</td></tr></table> Considering table t3, for the following series of statements, determine which changes will become permanent, and what will be the content of the table t3 after statement 12. Justify your answer. 1. START TRANSACTION; 2. INSERT INTO t3 VALUES(4, 'A',100); 3. DELETE FROM t3 WHERE code=2; 4. ROLLBACK WORK 5. BEGIN; 6. UPDATE t3 SET grade ='C' WHERE code = 1; 7. ROLLBACK WORK 8. INSERT INTO t3 VALUES (5, 'A', 200); 9. DELETE FROM t3 WHERE code = 1; 10. ROLLBACK WORK 11. UPDATE t3 SET value = 400 WHERE code = 1; 12. COMMIT WORK
Ans.	Statement 11 will become permanent because statement 2,3 ,6,8 and 9 are rollbacked. Table t3 will finally contain following records <table><tr><td>1</td><td>G</td><td>400</td></tr><tr><td>2</td><td>K</td><td>600</td></tr></table>

		3	B	200	
4	For the following series of statements, determine which will become permanent: 1. SELECT… 2. SAVEPOINT S1 3. INSERT… 4. COMMIT WORK 5. INSERT … 6. SAVEPOINT S2 7. DELETE … 8. ROLLBACK WORK TO SAVEPOINT S1 9. DELETE … 10. SAVEPOINT S3 11. DELETE … 12. ROLLBACK WORK TO SAVEPOINT S3 13. COMMIT WORK
Ans.	Statement 3 will be permanently saved because commit is following immediate after the statement. Statement 11 will be permanently deleted because rollback operation is done upto savepoint S3 and this statement is after S3.