



## QUESTION BANK

**Year: II- II Branch: CSE Subject: DBMS Name of the Faculty: T.S.S.L.N. Sarma**

### UNIT-I

**Data base System Applications:** data base System VS file System – View of Data – Data Abstraction – Instances and Schemas – data Models – the ER Model – Relational Model – Other Models – Database Languages – DDL – DML – database Access for applications Programs – data base Users and Administrator – Transaction Management – data base System Structure – Storage Manager – the Query Processor

- 1) Diff b/w data base System VS file System
- 2) What are the database Applications
- 3) What are the Advantages of a DBMS
- 4) What is the Data Abstracations? Explain each of level
- 5) What are the Data models?
- 6) What are the Database languages
- 7) Explain Procedure the database Access for applications Programs
- 8) Describe the Structure of DBMS
- 9) Explain the Query Processor

### UNIT-II

**History of Data base Systems.** Data base design and ER diagrams – Beyond ER Design Entities, Attributes and Entity sets – Relationships and Relationship sets – Additional features of ER Model – Concept Design with the ER Model – Conceptual Design for Large enterprises

- 1) Explain the History of Data base Systems?
- 2) Explain the Data base design?
- 3) Discuss about the ER-Diagrams?
- 4) What the features of the ER-Models?
- 5) What are the Class Hierarchies?
- 6) Diff b/w Relationships and Relationship sets?
- 7) Discuss about the Concept Design with the ER Model?
- 8) Explain the Conceptual Design for Large enterprises?

### UNIT III

**Introduction to the Relational Model** – Integrity Constraint Over relations – Enforcing Integrity constraints – Querying relational data – Logical data base Design – Introduction to Views – Destroying /altering Tables and Views.

**Relational Algebra** – Selection and projection set operations – renaming – Joins – Division – Examples of Algebra overviews – Relational calculus – Tuple relational Calculus – Domain relational calculus – Expressive Power of Algebra and calculus

- 1) How distinct tuples are in a relation instance with cardinality?
- 2) Does the relational model, as seen by SQL query writer provide physical and logical data independence?
- 3) What is a relation? Differentiate between relation schema and relational instance. Define the terms arity and degree of a relation? What are domain constraints?
- 4) What SQL construct enables the definition of a relational? What constraints allow modifications of relation instances?
- 5) What are the integrity constraints? Define the terms primary key constraints and foreign key constraints. How are these constraints expressed in SQL?
- 6) What are other kind of constraints can we express in SQL?
- 7) What is relational database query?
- 8) What is the input to a relational query? What is the result of evaluating a query?
- 9) Explain why algebra is suitable for this purpose?
- 10) Describe the selection operator
- 11) describe the set operations of relational algebra,
- 12) What is the difference between tuple relational calculus and domain relational calculus?

#### UNIT IV

**Form of Basic SQL Query** – Examples of Basic SQL Queries – Introduction to Nested Queries – Correlated Nested Queries Set – Comparison Operators – Aggregative Operators – NULL values – Comparison using Null values – Logical connectivity's – AND, OR and NOT – Impact on SQL Constructs – Outer Joins – Disallowing NULL values – Complex Integrity Constraints in SQL Triggers and Active Data bases.

- 1) What are the parts of basic SQL query? Are input result tables of SQL query sets or multisets? How can obtain a set of tuples as the result of a query?
- 2) What are the variables in SQL? How can you give names to output columns in a query that are defined by arithmetic or string expressions? what support does SQL offer for string pattern matching
- 3) What Are operations does SQL provide over (multi)sets of tuples ,and how would you use the in writing queries ?
- 4) What are nested queries? What is correlation in nested queries?
- 5) What are aggregate operators does SQL support?
- 6) What is grouping ?what are null values
- 7) What types of SQL constraints can be specified using the query language
- 8) How would you use these operators IN, EXISTS, UNIQUE, ANY, and ALL in writing nested queries?

#### UNIT V

**Schema refinement** – Problems Caused by redundancy – Decompositions – Problem related to decomposition – reasoning about FDS – FIRST, SECOND, THIRD Normal forms – BCNF – Lossless join Decomposition – Dependency preserving Decomposition – Schema refinement in Data base Design – Multi valued Dependencies – FOURTH Normal Form.

- 1) Illustrate redundancy and the problem that it can cause. Give examples of insert, delete, and update anomalies? Are they a complete solution?
- 2) What is decomposition and how does it address redundancy? What problem may be caused by the use of decompositions?
- 3) Define functional dependencies. how are primary keys related to FD's?
- 4) Define 1NF, 2NF, 3NF, and 4NF and BCNF .what is the motivation for putting a relation in BCNF? What is the motivation for 3NF?
- 5) When is a decomposition said to be dependency-preserving? Why is this property?
- 6) Discuss how schema refinement through dependency analysis and normalization can improve schemas obtained through ER design?

- 7) Why are some functional dependencies called trivial ?

## UNIT VI

**Transaction Concept-** Transaction State- Implementation of Atomicity and Durability – Concurrent – Executions – Serializability- Recoverability – Implementation of Isolation – Testing for serializability- Lock –Based Protocols – Timestamp Based Protocols- Validation- Based Protocols – Multiple Granularity.

- 1) What are the ACID properties? Define atomicity, Consistency, isolation, and durability and illustrate them through examples?
- 2) Define terms transactions, schedule, complete schedule and serial schedule?
- 3) Why does a DBMS interleave concurrent transactions?
- 4) What is a serializable schedule? What is a recoverable schedule?
- 5) What is a locking protocol ?Describe the Strict Two-Phase locking
- 6) How are transactions created and terminated in SQL?
- 7) What transaction characteristics can programmer control in SQL?
- 8) What are save points? What are chained Transactions?
- 9) Discuss the different access modes and isolation levels in particular?
- 10) Explain why save points and chained transactions are useful?
- 11) What recovery-related steps are taken during normal execution?
- 12) What can a DBA control to reduce the time to recover from a crash?
- 13) How is the log used in transaction rollback and crash recovery?

## UNIT VII

**Recovery and Atomicity** – Log – Based Recovery – Recovery with Concurrent Transactions – Buffer Management – Failure with loss of nonvolatile storage-Advance Recovery systems- Remote Backup systems.

- 1) Compare the deferred and immediate modification versions of the log-based recovery scheme in terms of ease of implementation and overhead cost?
- 2) When the system recovers from a crash, it constructs an undo-list and a redo-list?
- 3) Explain why log records for transactions on the undo-list must be processed in reverse order, while those log records for transactions on the redo-list are processed in a forward direction?
- 4) Explain the reasons why recovery of interactive transactions is more difficult to deal with than is recovery of batch transactions?
- 5) Explain the difference between the three storage types-volatile, nonvolatile, and stable in terms of I/O cost?
- 6) Stable storage cannot be implemented. Explain why it cannot be?
- 7) Explain the purpose of the checkpoint mechanism? How often should checkpoints be performed?
- 8) Explain the difference between a system crash and a “Disaster”?
- 9) Explain the benefits of logical logging? Give an example?
- 10) Explain how the buffer manager may cause the database to become inconsistent if some log records pertaining to a block are not output to stable storage before the block is output to disk?

## UNIT VIII

**Data on External Storage** – File Organization and Indexing – Cluster Indexes, Primary and Secondary Indexes – Index data Structures – Hash Based Indexing – Tree based Indexing – Comparison of File Organizations – Indexes and Performance Tuning- Intuitions for tree Indexes – Indexed Sequential Access Methods (ISAM) – B+ Trees: A Dynamic Index Structure.

- 1) What is a File organization? What is an Index?
- 2) What is the relationship between files and indexes?
- 3) Can we have several indexes on a single file of records? Can an index itself store data records?
- 4) What is a Clustered index? What is a primary index?
- 5) How many clustered indexes can you build on a file?
- 6) How is data organized in a hash-based index? When would you use a hash-based index?
- 7) How is data organized in a tree-based index? When would you use a Tree-based index?